



HAL
open science

Gestion des informations dans les premières étapes de la vision par ordinateur

Jean-Marc Salotti

► **To cite this version:**

Jean-Marc Salotti. Gestion des informations dans les premières étapes de la vision par ordinateur. Interface homme-machine [cs.HC]. Institut National Polytechnique de Grenoble - INPG, 1994. Français. NNT: . tel-00005111

HAL Id: tel-00005111

<https://theses.hal.science/tel-00005111>

Submitted on 25 Feb 2004

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

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

THESE

présentée par

Salotti Jean Marc

pour obtenir le grade de DOCTEUR

de l'Institut National Polytechnique de Grenoble

(arrêté ministériel du 30 mars 1992)

Spécialité : Informatique

GESTION DES INFORMATIONS
DANS LES PREMIERES ETAPES
DE LA VISION PAR ORDINATEUR

Date de soutenance : 25 janvier 1994

Composition du jury :

Jean Marc Chassery (Examineur)
Jean Pierre Cocquerez (Rapporteur)
Catherine Garbay (Directrice)
Augustin Lux (Examineur)
Piero Zamperoni (Rapporteur)

Thèse préparée au sein du laboratoire TIMC / IMAG

Remerciements

Je voudrais remercier d'abord Catherine Garbay qui m'a accueilli dans son équipe et qui a bien voulu encadrer cette thèse. Elle a su orienter mes recherches dans le bon sens avec un remarquable esprit critique et une dévotion sans pareil. Grâce à un dialogue constant, j'ai appris énormément tant au niveau de la recherche qu'au niveau de la méthode de travail, ce dont j'avais le plus grand besoin. Elle a su également canaliser ce que j'appellerai mon enthousiasme de jeune chercheur, pour que je puisse tirer parti du meilleur de moi-même. Je ne peux que la remercier chaleureusement pour son aide, sa disponibilité et sa gentillesse.

Je voudrais remercier Jean Pierre Cocquerez pour avoir accepté d'être rapporteur de ma thèse. Ses remarques pertinentes m'ont permis d'approfondir un peu plus mes travaux et d'adopter une démarche plus rigoureuse. C'est pour moi un honneur qu'il soit présent à ce jury.

Je voudrais remercier Piero Zamperoni à la fois pour avoir accepté d'être rapporteur, mais aussi pour toutes les discussions intéressantes par courrier électronique interposé dans le cadre du Groupe de Réflexion en Vision par Ordinateur. Je suis également très honoré de sa présence à ce jury.

Je voudrais remercier Jean Marc Chassery et Augustin Lux pour avoir accepté d'être membres de mon jury. J'ai pu discuter avec eux à de nombreuses reprises, lors de séminaires ou plus personnellement, ce qui s'est toujours avéré très intéressant.

Enfin, je voudrais remercier tous les membres du groupe SIC et toutes les personnes travaillant dans notre laboratoire, qui ont tous contribué à créer une ambiance agréable et conviviale propice à des travaux de thèse.

Sommaire

Remerciements	2
Introduction générale	6
1. Cadre de la thèse	6
2. Objectifs	6
<u>Chapitre 1 : Les problèmes de gestion des informations</u>	8
1. Introduction	9
2. La détection d'indices visuels	11
2.1 La vision humaine	11
2.2 La vision artificielle	14
2.2.1 Organisation d'un système de vision	14
2.2.2 Les indices visuels	17
3. La gestion des informations	20
3.1 L'abondance des informations	20
3.2 La complémentarité des informations	23
3.2.1 Les bases du problème	23
3.2.2 La technique incrémentale	24
3.3 L'émergence de l'information	26
3.3.1 La notion d'émergence	26
3.3.2 La reconnaissance de visages	27
3.3.3 L'émergence des informations 3D	28
3.3.4 Conclusion sur l'émergence des informations	30
3.4 La liberté de l'information	30
4. La conception des algorithmes	32
4.1 Etude théorique de l'indice visuel	32
4.2 Conception expérimentale d'une stratégie heuristique	34
<u>Chapitre 2 : La détection des contours</u>	37
1. Introduction	38
2. Etat de l'art	39
2.1 Les techniques de filtrage	39
2.1.1 Filtrage linéaire	39
2.1.2 Filtrage non linéaire	40
2.1.3 Discussion	41
2.2 Les différentes techniques de détection des contours	42
2.2.1 Gradients et laplaciens	42
2.2.2 L'affinage et le seuillage	44
2.2.3 Le prolongements, la correction et le chaînage	47
2.2.4 Le modèle de Canny	48
2.2.5 Le modèle de Perona et Malik	50
2.2.6 D'autres détecteurs de contours	52
2.3 L'évaluation des contours	54
2.3.1 Evaluation visuelle	54
2.3.2 Evaluation statistique	56
2.3.3 Evaluation pragmatique	58

2.4 Discussion	59
2.4.1 Les petites phrases	59
2.4.2 La critique du lissage	62
2.4.3 La critique du modèle "step edge"	63
2.4.4 L'enchaînement des étapes	64
2.4.5 Conclusion	65
3. Choix algorithmiques pour notre détecteur	66
3.1 Protocole de construction de l'algorithme	66
3.2 Spécifications générales	68
3.3 Calcul du gradient	70
3.3.1 Les différents choix possibles	70
3.3.2 Choix retenu	73
3.4 Choix des germes	74
3.4.1 Les différents choix possibles	74
3.4.2 Choix retenu	77
3.5 Choix du pixel suivant	79
3.5.1 Les différents choix possibles	79
3.5.2 Choix retenu	81
3.6 Recherche d'informations sur les régions voisines	83
3.6.1 Les différents choix possibles	83
3.6.2 Choix retenu	86
3.7 Décisions heuristiques	88
4. Ajustement des heuristiques	90
4.1 Décomposition de notre expertise	90
4.2 Formes transversales des contours	91
4.2.1 Critère de visibilité	91
4.2.2 Recherche du critère de visibilité	91
4.2.3 Construction de l'heuristique	94
4.3 Formes longitudinales des contours	97
4.3.1 Critère de visibilité	97
4.3.2 Proposition d'heuristique	99
4.4 Ajustement manuel	100
5. Evaluation	102
5.1 Protocole expérimental	102
5.2 Tests locaux	104
5.2.1 Gradients faibles (type 1)	104
5.2.2 Gradients complexes (type 2)	105
5.3 Tests globaux	107
5.3.1 Proximité d'autres contours (type 3)	107
5.3.2 Les petits contours (type 4)	108
5.3.3 Dynamique complexe (type 5)	109
5.4 Tests négatifs	112
5.4.1 Dégradés (type 6)	112
5.4.2 Textures (type 7)	113
5.5 Evaluation de notre détecteur	114
5.5.1 Evaluation des coefficients du gradient relatif	114
5.5.2 Comparaison avec le détecteur de Deriche	115
5.5.3 Evaluation globale	117
5.6 Les temps de calculs	119
6. Conclusion sur la détection des contours	120
6.1 Conclusion sur l'évaluation	120
6.2 Perspectives	120
6.3 Exploitation des contours	122

Chapitre 3 : La segmentation descriptive	124
1. Introduction	125
2. Etat de l'art	127
2.1 Les méthodes simples de segmentation	127
2.1.1 Exploitation de l'histogramme	127
2.1.2 Croissance de régions	128
2.2 La coopération régions-contours	131
2.2.1 Les approches classiques	131
2.2.2 Le système expert de Nazif et Levine	132
2.2.3 L'approche multi-agents	134
2.3 Discussion	135
2.3.1 Les petites phrases	135
2.3.2 Les informations sémantiques	137
2.3.3 Le but de la segmentation	138
3. Une coopération région-contours intégrale	142
3.1 Agrégation de pixels adaptative	142
3.1.1 Spécifications générales	142
3.1.2 Description de l'algorithme	143
3.1.3 Définition du critère de similarité	147
3.1.4 Spécifications des critères	148
3.1.5 Formulation de l'heuristique	152
3.1.6 Evaluation de notre détecteur de régions	154
3.1.7 Discussion	159
3.2 Gestion de la coopération	160
3.2.1 Contrôle incrémental	160
3.2.2 Enchaînement des méthodes	162
3.2.3 Création des premiers germes	162
3.2.4 Aide des contours vers les régions	165
3.2.5 Aide des régions vers les contours	167
4. Evaluation	171
4.1 Analyse de quelques résultats	171
4.1.1 L'image du muscle	171
4.1.2 L'image de la femme	172
4.1.3 L'image de la souris	172
4.1.4 L'image IRM	173
4.2 Evaluation globale	173
4.2.1 Utilité de nos principes	173
4.2.2 Evaluation qualitative des résultats	175
5. Conclusion sur la coopération	176
5.1 Aspects positifs	176
5.2 Aspects négatifs	177
5.3 Perspectives	177
5.3.1 Les critères d'agrégation	177
5.3.2 Une aide plus efficace entre les processus	178
Conclusion générale	180
Références	182
Annexe 1 : Pixels-contours de visibilité ambiguë	192
Annexe 2 : Résultats du détecteur de contours	195
Annexe 3 : Résultats de la coopération région-contour	200

INTRODUCTION GENERALE

1. Cadre de la thèse

Cette thèse a été effectuée dans le laboratoire Traitement d'Images Modélisation et Cognition de l'institut d'Informatiques et Mathématiques Appliquées de Grenoble. Notre groupe travaille en étroite collaboration avec des spécialistes de cytologie et de biologie moléculaire. Les biologistes ont besoin d'automatiser les traitements qu'ils effectuent sur des images de cellules vues au microscope. Nos applications de vision par ordinateur sont donc essentiellement tournées vers l'identification, le dénombrement ou l'examen de cellules biologiques. En ce qui concerne mes travaux, j'ai développé un algorithme de détection de contours particulièrement adapté à ce genre d'images.

Pendant ma thèse, j'ai fait partie du groupe Systèmes Intégrés Cognitifs. Outre les problèmes relatifs à la vision, notre groupe est intéressé par d'autres domaines de l'intelligence artificielle, notamment les systèmes multi-agents, la modélisation de l'expertise et les fondements même de l'intelligence, du raisonnement à l'acquisition des connaissances. L'orientation de mes travaux a été fortement influencée par l'aspect "analyse du traitement de l'information" qui caractérise sans doute la plupart des travaux du groupe. Comme nous allons essayer de le montrer par la suite, cette particularité nous a poussé à aborder les problèmes de la vision sous un angle différent, et à concevoir des algorithmes nouveaux pour la détection d'indices visuels.

2. Objectifs

L'objectif de cette thèse est d'aborder le problème de la détection des indices visuels simples sous l'angle de la gestion des informations. Par gestion des informations, nous entendons la manière dont est décomposé le problème, comment s'enchaînent les étapes, comment se fait le flux des informations, quelles sont les relations entre les différents types d'informations et quelle est la structure de contrôle de l'algorithme. Il ne s'agit pas ici uniquement de génie logiciel. Les problèmes de gestion des informations nous amènent à étudier les problèmes de conception, de réalisation et d'évaluation liés à notre application. La nature de l'application est très importante car nous verrons que le but est difficile à définir, ce qui entraîne des répercussions d'ordre méthodologique.

Nous proposons dans une première partie l'étude théorique ou générale des problèmes de gestion des informations, en illustrant nos propos par des exemples. Nous

discutons brièvement du fonctionnement de la vision humaine en essayant de situer l'étape de détection des premiers indices visuels ("indices visuels primaires"), qui nous intéresse plus particulièrement. Nous effectuons ensuite une analyse des problèmes liés à la prise de décision, à l'émergence des informations et au flux de celles-ci. Nous discutons également du problème de conception des algorithmes, où nous opposons les approches qui tentent de modéliser les indices visuels aux approches heuristiques qui privilégient l'aspect expérimental. La méthodologie que nous proposons est sans doute un point essentiel de notre contribution. Ce chapitre nous permet donc de dégager les principes de base pour la construction d'un détecteur d'indices visuels.

Nous proposons dans une deuxième partie l'étude de la détection des contours. Nous commençons par une analyse bibliographique et une discussion générale des différentes approches. Nous remarquons qu'il existe un débat de fond dans la façon d'aborder le problème de la détection des contours et que nous sommes peut-être à un tournant dans les orientations de recherche dans ce domaine. Nous présentons ensuite un nouveau détecteur de contours adaptatif, qui respecte les grands principes présentés dans la première partie. Nous effectuons une évaluation pragmatique détaillée de ce détecteur, ce qui constitue à notre avis un premier pas vers une évaluation réellement efficace. Notre détecteur et notre façon de l'évaluer sont en fait les deux grandes parties de la même approche que nous préconisons pour la détection des discontinuités.

Dans une troisième partie, nous nous intéressons à la détection des régions, éléments qui constituent un autre type d'indice visuel. Nous effectuons une analyse bibliographique, puis nous nous focalisons sur la coopération régions-contours. La coopération comporte implicitement des problèmes de gestion des informations, ce qui va nous permettre d'illustrer les grands principes présentés au premier chapitre. Nous proposons un nouvel algorithme de coopération régions-contours fondé sur le principe d'incrémentalité et une structure de contrôle symétrique. Nous évaluons notre algorithme en présentant des résultats et en commentant les erreurs et les imperfections constatées.

Nous concluons enfin en résumant notre contribution et en proposant les perspectives de nos travaux.

CHAPITRE 1

Les problèmes de gestion des informations

1. Introduction

Le but des premières étapes de la vision par ordinateur est de **décrire** l'image à l'aide d'**indices visuels simples**. Ces indices doivent être pertinents afin de faciliter les étapes suivantes dans une perspective de reconnaissance rapide et robuste des objets de la scène. Cette description consiste souvent en une détection des discontinuités ou une segmentation de l'image en régions.

Dans la première partie de ce chapitre, nous essayons de définir la notion "d'indice visuel primaire" en étudiant le mécanisme global de la vision. D'abord, nous nous penchons brièvement sur la vision humaine, puis nous essayons de dégager les grandes lignes d'un système de vision artificiel. Ceci nous conduit à cerner les difficultés majeures de la détection d'indices visuels.

Dans la deuxième partie de ce chapitre, nous abordons les problèmes liés à la gestion des informations dans les algorithmes de détection d'indices visuels. Ces problèmes sont difficiles à aborder car il n'est pas possible d'en faire une modélisation simple et le but lui-même n'est pas clairement défini. Nous nous attachons plus particulièrement à identifier et à comprendre les difficultés qui précèdent la décision qui détermine la présence d'un indice visuel. Nous présentons ainsi des idées originales concernant l'abondance et la complémentarité des informations, puis nous abordons le problème délicat de leur qualité objective qui conditionne l'émergence des indices visuels et nous terminons par la notion intuitive de "liberté de l'information".

Dans la dernière partie de ce chapitre, nous examinons les problèmes liés à la conception d'un détecteur d'indices visuels en présentant deux grandes catégories d'approches, l'une consistant à modéliser l'indice et l'autre cherchant à approcher la solution par des heuristiques.

L'étude des problèmes de gestion des informations nous semble finalement d'un grand intérêt pour deux raisons essentielles :

- Elle nous a permis d'abord de mieux comprendre les défauts et les limites des approches classiques. Nous formulons ainsi dans ce chapitre des critiques nouvelles et nous n'hésitons pas à remettre en cause des principes fondamentaux, qui sont à la base même de nombreuses approches en vision.
- Elle nous a conduit ensuite à proposer une méthodologie nouvelle de conception et d'évaluation d'algorithmes, illustrée dans les chapitres suivants par un nouveau détecteur

de contours, un nouvel algorithme de croissance de régions et une coopération régions contours fondée sur le principe d'incrémentalité.

2. La détection d'indices visuels

Nous définissons un indice visuel comme étant une information descriptive de l'image, ne faisant intervenir aucune connaissance sur les objets. Nous essayons ici de situer l'étape de détection des indices visuels dans un système de vision et de définir les "indices visuels primaires" qui nous intéressent plus particulièrement dans le cadre de cette thèse.

2.1 La vision humaine

La qualité de la vision humaine est exceptionnelle. Elle s'avère être malheureusement une mécanique extrêmement complexe dont nous n'avons qu'une idée générale du fonctionnement. Des expériences ont été réalisées et des théories ont été proposées mais elles ne donnent qu'une vue d'ensemble de la vision naturelle [HuWi 79], [Gros 80], [Gros 87], [Bied 85], [Free 91].

Notre but est ici de rappeler quelques éléments connus de la vision naturelle (cités dans la plupart des ouvrages) et de situer l'étape de détection d'indices visuels. Cette étude reste d'un niveau très superficiel car d'une part seuls les principes généraux nous intéressent et d'autre part la neurobiologie n'est pas de notre compétence.

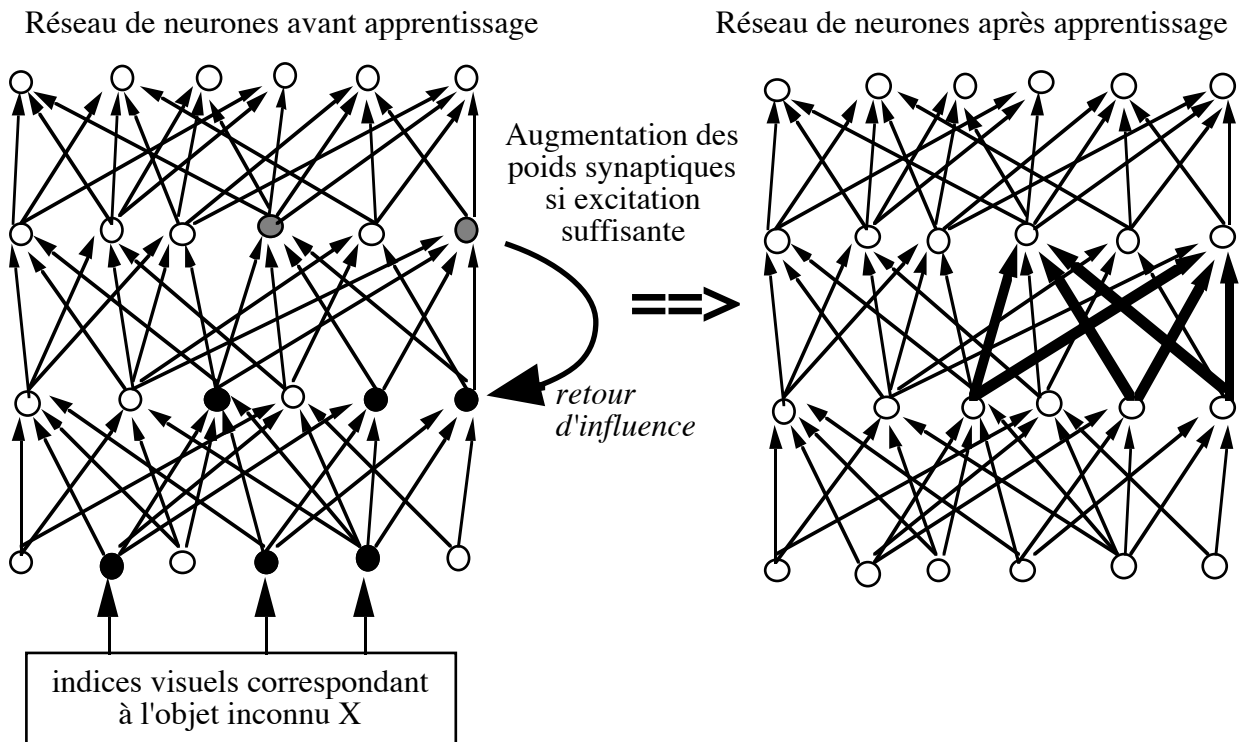
- Des tests sur la rapidité de la reconnaissance visuelle ont montré que celle-ci est très rapide. En tenant compte de la vitesse relativement lente de transmission du potentiel au niveau de la synapse qui lie deux neurones, les chercheurs ont émis l'hypothèse que le traitement des informations visuelles est direct, des récepteurs situés sur la rétine jusqu'aux neurones exprimant la reconnaissance d'un objet [Free 91]. En d'autres termes, il n'y a pas de temps d'attente ou de cycle de retour pour une remise en cause de l'information ou une vision à deux vitesses. Tout se passe comme si la vision était un enchaînement d'étapes simples sans remise en cause des résultats intermédiaires.

- L'étude du cerveau montre qu'il existe des régions où les neurones sont différemment organisés. Les neurones sont souvent regroupés en couche avec des spécificités propres, tant au niveau des connexions avec les autres couches qu'au niveau des propriétés internes comme le seuil d'excitabilité. Cette particularité permet de supposer qu'il existe des groupes de neurones dédiés à une tâche particulière. La vision pourrait ainsi être décomposée en plusieurs étapes distinctes. Des expériences ont par exemple montré que certaines zones sont sensibles à la couleur alors que d'autres sont sensibles à la forme, ce qui confirme cette hypothèse [Gros 87].

- Un des phénomènes liés étroitement à la vision est notre capacité d'apprentissage. Par exemple, quelques secondes d'attention suffisent pour que nous mémorisions un nouveau visage et que nous soyons capables de le reconnaître quelques instants plus tard. Nous sommes donc capables d'apprendre en un temps très court les signes caractéristiques d'un visage. Des expériences ont montré qu'après apprentissage, certains poids synaptiques (la force des liaisons entre les neurones) changent. Lorsque le même événement se reproduit, des neurones qui n'étaient pas excités avant apprentissage le sont ensuite et caractérisent celui-ci. Cela suppose que le changement des poids synaptiques est un des mécanismes essentiels à l'apprentissage et à la mémorisation [Klop 82].

A travers ces grands principes, essayons de dégager un schéma approximatif de la vision naturelle (voir figure 1). La détection d'indices visuels serait donc la première grande étape de la vision. Ces indices seraient des informations sur la présence et la forme des discontinuités, sur la disposition de points caractéristiques ainsi que des informations sur la couleur, sur l'homogénéité, la texture ou le dégradé des régions. Finalement, de façon grossière, nous pourrions considérer le fonctionnement suivant : si un indice visuel primaire est présent, nous pouvons supposer que des neurones spécifiques sont excités. Ensuite, à l'étape suivante, d'autres neurones spécifiques sont excités en fonction de la présence simultanée de plusieurs indices visuels primaires. Le schéma de la figure 1 est bien sûr très simplifié car nous pouvons imaginer qu'il existe ainsi plusieurs niveaux de complexité croissante correspondant à la présence d'indices visuels de plus en plus sophistiqués. Il devrait y avoir de plus des zones traitant un problème spécifique. Si les indices primaires sont essentiellement locaux comme par exemple les discontinuités, il est probable que les indices visuels de niveau supérieur correspondent à des associations d'indices recouvrant une partie plus grande de l'image comme des formes polyédriques ou des symétries globales. Ensuite, nous pouvons sans doute continuer l'association d'indices visuels pour former des indices de niveau encore supérieur en trois dimensions [Gros 87]. A ce stade, il n'y a pas encore reconnaissance, mais uniquement génération d'hypothèses de structure sur ce qui est vu. Les contraintes globales doivent pouvoir inhiber les neurones correspondant à des hypothèses non vérifiées, ce qui permettrait de faire émerger finalement une description de la scène avec des indices visuels de très haut niveau. Ensuite, deux cas se présentent. Si les neurones excités à ce dernier niveau associent leurs réponses pour exciter un groupe de neurones qui identifient un objet, la reconnaissance est validée. Sinon, il est probable que c'est la première fois que les indices visuels présents ont été découverts simultanément. Ils excitent un groupe de neurones (un seul sur la figure 1) et par modification des poids synaptiques, ceux-ci correspondront à l'identification de l'objet inconnu.

a) Apprentissage des caractéristiques d'un objet



b) Reconnaissance d'un objet

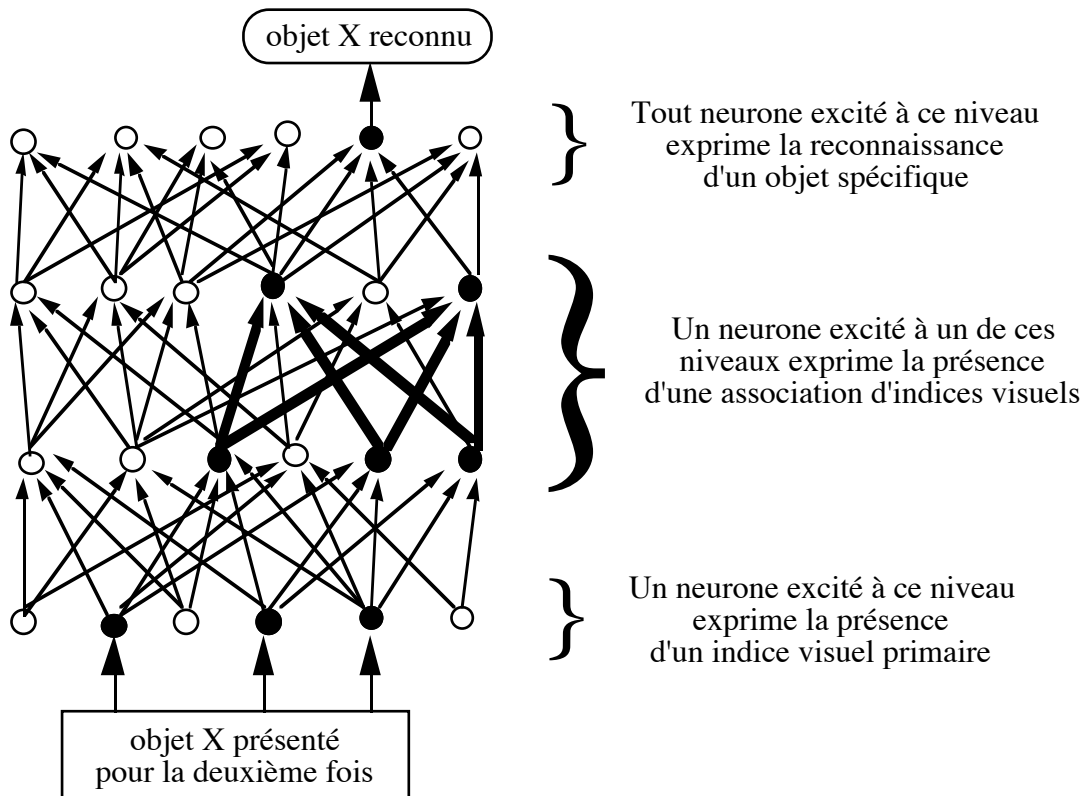


Figure 1 : Une version possible de l'apprentissage et de la reconnaissance d'objets.

Lorsque l'objet est de nouveau présent dans la scène, les indices visuels sont les mêmes. Grâce à l'apprentissage, le groupe de neurones identifiant l'objet va être excité ce qui caractérise alors la reconnaissance. Nous ne prétendons pas ici donner une description exacte de la vision naturelle, nous donnons juste une idée de ce que pourrait être son fonctionnement.

A travers l'étude de la vision humaine, nous nous apercevons que la notion d'indice visuel est complexe et qu'il est bien difficile d'en donner une définition précise. En particulier, est-il raisonnable d'appeler indices visuels une description en trois dimensions de la scène (Marr ou Grossberg parlent par exemple de "perception 3D" [Marr 82], [Gros 87]) ? Quoiqu'il en soit, nous n'étudions dans le cadre de cette thèse que les indices visuels primaires, donc nécessairement de dimension inférieure.

2.2 La vision artificielle

2.2.1 Organisation d'un système de vision

Nous proposons ici une étude très simple des systèmes de vision artificielle. Notre but est de comprendre quels sont les résultats attendus de la détection des indices visuels et dans quel contexte cette étape se situe. De nombreuses tentatives ont été faites pour réaliser un système complet de vision. Nous pouvons classer les approches en deux catégories, celles qui tentent de copier la vision humaine (approche neuromimétique) et celles qui considèrent la vision indépendamment de l'exemple humain ou animal.

- En ce qui concerne les approches neuromimétiques, nous pouvons citer les travaux de Grossberg et Mingolla dont le but est d'exploiter des réseaux de neurones informatiques pour simuler les différentes étapes de la vision humaine [GrMi 87], ainsi que les travaux de Gaussier et Cocquerez pour la reconnaissance de scènes selon des principes similaires avec simulation des "saccades oculaires" [GaCo 91]. Les approches fondées sur les réseaux de neurones ont montré de grandes qualités pour diverses applications et la recherche est actuellement très active dans ce domaine.

- Citons l'approche de David Marr qui a proposé une théorie de la vision avec un paradigme de description de l'image en trois étapes : croquis élémentaire, croquis 2D1/2 et description 3D [Marr 82]. L'originalité de cette théorie est de vouloir construire une description 3D très riche sans incorporer d'information sémantique. Cette approche s'inspire fortement des principes de la vision humaine, avec une tentative d'adaptation à

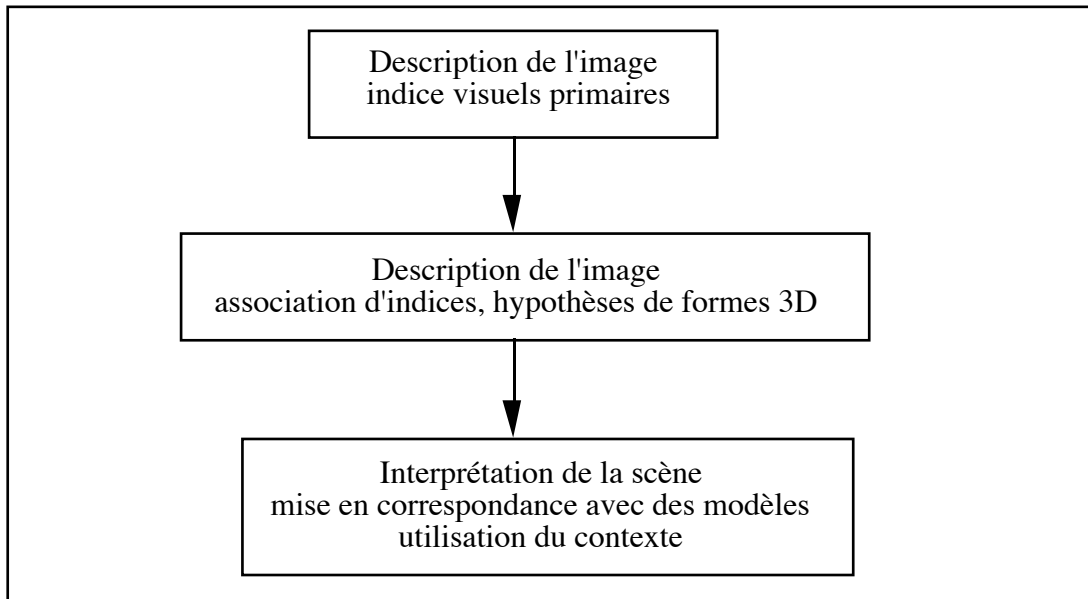
nos moyens informatiques. Le paradigme de Marr représente un courant de pensée important dans la communauté et de nombreux travaux actuels s'inspirent de cette théorie.

- En ce qui concerne les systèmes de vision qui n'ont pas été inspirés de la vision humaine, un des plus connus est celui de Hanson et Riseman, VISIONS [HaRi 78]. Ce système est capable de reconnaître des routes, des arbres et des maisons dans un contexte relativement limité. Une des caractéristiques de ce système est l'exploitation du "tableau noir" qui permet de relier tous les processus entre eux et d'échanger ainsi des informations de différente nature. Le système ACRONYM [Broo 81] est également un projet ambitieux qui a donné lieu à une application en imagerie aérienne, de même que le système MESSIE [GaGM 89] ou le système SPAM [McHM 85]. Chacun de ces systèmes a été implanté avec des algorithmes *ad hoc* développés pour la circonstance. Citons également les travaux de Clement et Thonnat avec le système expert OCAPAPI [Clem 90], ainsi que l'approche de Baujard et Garbay [BaGa 93] qui ont exploité une architecture "multi-experte" (le système MAPS) pour construire un système de vision. Dans les deux cas, les meilleurs outils du traitement d'images sont combinés entre eux pour résoudre une application concrète, comme par exemple la classification des galaxies ou la reconnaissance de cellules spécifiques. Notons qu'une étude des systèmes experts pour la vision a été réalisée par Matsuyama [Mats 89], qui propose lui-même le système SIGMA [Mats 85], pour intégrer simultanément les stratégies "dirigées par les données" et "dirigées par les buts". Une des caractéristiques de ces systèmes est la possibilité d'exploiter des connaissances de haut niveau pour guider la détection des indices visuels. L'interaction entre tous les niveaux d'un système de vision est à opposer à la décomposition hiérarchique de Marr et aux approches neuromimétiques. Il existe enfin des approches mixtes combinant par exemple réseaux de neurones et systèmes experts [CoGP 92], où le neuromimétisme est centré sur les premières étapes du système de vision.

Que ce soit avec les approches neuromimétiques ou les autres, il existe en fait deux manières différentes de considérer l'étape de détection des indices visuels qui conduisent à deux grandes catégories de système de vision (voir figure 2). La première implique une détection d'indices visuels de plus en plus riches sans l'aide d'information sémantique, c'est l'approche guidée par les données, et la deuxième exploite au mieux le contexte pour aider la détection de ces indices, c'est l'approche dirigée par le but. Il nous semble que ces deux approches sont complémentaires avec des domaines d'applications spécifiques. Dans le deuxième cas en effet, il faut que le contexte soit simple et bien déterminé pour que la détection d'indices puisse être correctement guidée, ce qui implique une application relativement simple. Dans le premier cas en revanche, il s'agit d'un cadre plus général où

les situations ne sont pas toutes connues avec précision. Il existe bien sûr de nombreuses variantes avec des différences dans la structure de contrôle et dans les niveaux de représentations (voir notamment les travaux de Demazeau [Dema 87]). En ce qui nous concerne, nous n'avons pas choisi d'application particulière, nous étudierons donc la détection des indices visuels primaires sans l'aide du contexte et d'informations *a priori* sur les objets, en tant que première étape d'un système de vision.

SYSTEME DE VISION NUMERO 1



SYSTEME DE VISION NUMERO 2

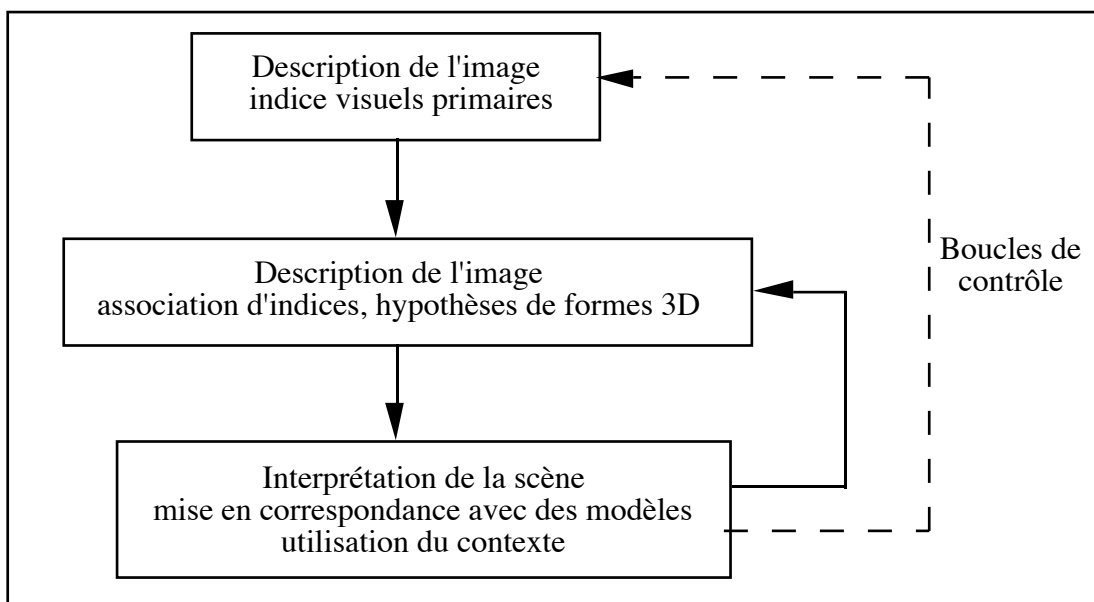


Figure 2 : Deux approches différentes pour décomposer le problème de la vision.

2.2.2 Les indices visuels

Après avoir étudié rapidement les systèmes de vision, il est nécessaire de savoir ce que nous entendons exactement par indices visuels et indices visuels primaires. De façon intuitive, un indice visuel est une information "visuelle", qui apparaît donc directement à partir de la visualisation de l'image. La visibilité d'un indice est en fait un des problèmes clés qu'il nous faudra résoudre. En ce qui concerne la distinction d'indices visuels primaires ou complexes, nous proposons la décomposition suivante :

Les indices visuels primaires sont des informations sur l'image obtenues directement à partir de calculs sur les pixels et formant les premières structures de données importantes de niveau juste supérieur à celui des pixels. Ce sont par exemple les discontinuités (souvent appelées contours) qui correspondent à des ruptures de propriété locales, les points caractéristiques et les régions qui sont au contraire des zones où une propriété locale ou globale est conservée. Les indices visuels étant descriptifs, ces propriétés doivent être relativement simples et faciles d'exploitation.

Une zone de texture uniforme (texture simple avec variations locales irrégulières) fait donc partie de ces indices visuels primaires, mais une zone de texture complexe comprenant des éléments répétitifs de structure est du niveau supérieur. Cet aspect nous paraît très important. Pour illustrer nos propos, nous présentons en annexe 2 de nombreuses images tests. Pour l'image du muscle, la texture des cellules grises est simple, aucun contour ne doit donc être détecté dans celles-ci (visuellement, aucun indice visuel de type contour n'est détectable). En revanche, pour l'image de la souris, la texture est composée de petits fragments de bois dont quelques uns sont visibles à l'œil nu. Pour que la visibilité de ces contours ne soit pas dépendante de la résolution de l'image, **nous supposons que toutes les images ont le même facteur d'échelle, quelle que soit l'application.** Cette hypothèse de travail est fondamentale pour lever une partie des ambiguïtés concernant la présence d'un indice visuel. En effet, si un indice visuel est défini à partir de la rupture ou de la conservation d'une propriété locale, il est nécessaire de fixer la taille de cette "localité" pour ne pas le confondre avec un indice visuel plus global. En particulier, pour un facteur d'échelle donné, une zone présentant un dégradé régulier doit pouvoir être représentée par un indice visuel, sans être confondue avec une discontinuité détectée à basse résolution. Finalement, tout groupe de pixels présentant une propriété intéressante simple constitue un indice visuel primaire qui mérite d'être détecté. La rupture d'une telle propriété ("rupture de modèle") est également un indice visuel intéressant. Nous proposons intuitivement discontinuités et régions mais notre imagination est limitée et il doit exister un nombre considérable d'indices visuels primaires intéressants.

Tous les autres indices visuels sont le résultat d'une modélisation et d'une association d'indices plus simples, ils ne sont donc plus "primaires". Par exemple, il est possible de modéliser les discontinuités par des courbes paramétriques ou des segments et jonctions, puis de les associer en fonction de leurs propriétés comme la proximité, la forme, la similarité du voisinage, la continuité etc... (théorie Gestaltiste, voir par exemple [SpLe 87] ou [RoPa 91]). De même, les régions peuvent être regroupées en fonction de leurs formes ou de leurs caractéristiques de niveaux de gris. Il peut y avoir également des associations plus complexes entre régions et discontinuités (pour détecter par exemple des textures complexes).

Ces deux niveaux d'indices visuels, présents figure 2, sont très grossiers. Si notre décomposition est simpliste, elle n'en revêt pas moins des difficultés majeures. En effet, il existe une propriété locale que nous avons volontairement exclue (ou non abordée) dans les indices visuels primaires, c'est la ou plutôt les propriétés de forme. En effet, une région a une forme dont la description peut être directement obtenue à partir de calculs sur la position relative des pixels. Nous pouvons d'ailleurs contraindre l'association des pixels pour arriver à une région de forme simple spécifique, ou même une forme seulement non complexe. Il en est de même pour les discontinuités. Par exemple un alignement de pixels plus clairs est visible dans une zone de texture alors que ces mêmes pixels placés autrement ne constituent aucun ensemble intéressant.

Le problème est que les informations relatives à la position des pixels s'ajoutent aux informations sur les niveaux de gris. L'une n'allant pas sans l'autre, la détection des indices visuels primaires doit donc prendre en compte simultanément ces deux types d'informations. Cependant, dans la pratique, c'est rarement le cas. Par exemple, les pixels sont souvent regroupés en fonction d'un modèle simple de région, ou les pixel-contours sont regroupés en fonction d'une forme à obtenir (en utilisant par exemple la transformée de Hough [BaBr 91]). Notons d'ailleurs que les regroupements en fonction de la position relative des pixels s'effectuent souvent après la détection des indices visuels primaires. S'il en est ainsi, c'est que les problèmes de gestion des informations sont d'une grande complexité et qu'il est difficile d'exploiter en même temps ces deux types d'informations. Pour illustrer nos propos, remarquons que l'être humain est capable de décrire pertinemment des images de cellules vues au microscope sans savoir de quoi il s'agit (chaque cellule est perçue comme une entité unique), alors que les algorithmes les plus performants exploitent des informations contextuelles sur l'image et l'application pour des résultats pas toujours probants. Or, les cellules ont souvent des forme simples ce qui permet de les identifier malgré les textures et les discontinuités pas toujours bien

marquées. Bien que cela soit difficile, il est donc important d'exploiter simultanément les informations sur la forme et les informations sur les niveaux de gris.

Pour conclure cette partie, nous retenons que les indices visuels primaires s'obtiennent directement à partir de calculs sur les niveaux de gris et la position des pixels. Leur détection pose de nombreux problèmes au niveau de la gestion des informations, problèmes que nous abordons dans la suite de ce chapitre.

3. La gestion des informations

La détection des indices visuels primaires s'avère être d'une grande complexité. Nous essayons dans cette partie de développer les grandes difficultés liées à la gestion des informations. Nous discutons d'abord de l'abondance des informations, tant au niveau de la quantité de données à traiter qu'au niveau du nombre de paramètres intervenant dans la justification de la présence d'un indice visuel. Nous évoquons ensuite les problèmes délicats de fusion des informations et d'exploitation de données d'origines diverses. Nous mettons également l'accent sur les problèmes liés à l'émergence des indices visuels et des informations en général, et nous introduisons la notion de "liberté de l'information" où nous abordons brièvement les problèmes de contrôle [Salo 92].

3.1 L'abondance des informations

Les premières étapes de la vision sont confrontées au problème de l'abondance des informations présentes dans une image. Les pixels représentant un ensemble de données de taille non négligeable, outre les problèmes d'espace mémoire, les temps de calculs deviennent vite prohibitifs. Ainsi, pratiquement tous les algorithmes proposés sont de complexité linéaire. Cela était encore plus vrai il y a dix ou vingt ans, car les ordinateurs étaient beaucoup moins rapides qu'aujourd'hui. En détection de contours, par exemple, des algorithmes très simples ont été proposés avec un calcul de gradient à l'aide de masques 2×2 et un seuillage simple ensuite. Actuellement, grâce au gain de puissance des ordinateurs et le développement des machines parallèles, la tendance est à la sophistication des algorithmes, mais tout en restant essentiellement dans le domaine de la complexité linéaire.

Le problème de l'abondance des informations est toutefois moins bien perçu au niveau de la prise de décision. Par exemple, lorsqu'il faut décider de la fusion de deux régions ou de l'existence d'une discontinuité, il existe un grand nombre d'informations théoriquement indispensables pour que la décision soit correcte. Nous allons détailler ces deux exemples pour mieux comprendre l'importance du problème.

- **PROBLEME DE FUSION DE DEUX REGIONS**

Lors de la fusion de deux régions, les informations utilisées sont généralement l'écart-type et la moyenne de gris des deux régions, la norme du gradient à la frontière, et le nombre de pixels qui composent cette frontière (voir par exemples les "survey" de Zucker [Zuck 76] ou de Haralick et Shapiro [HaSh 85]). Ce nombre d'informations est relativement conséquent et les critères amenant la meilleure décision sont difficiles à

déterminer. Toutefois, le nombre d'informations prises en compte est-il suffisant ? Au chapitre 3, nous présentons une croissance de régions qui prend en compte le gradient moyen et sa direction à l'intérieur de la région pour décider de l'annexion d'un nouveau pixel. Selon notre définition de l'indice visuel primaire, ne faudrait-il pas utiliser également des informations sur la forme des deux régions pour décider de leur fusion (par exemple si la région résultant de la fusion est de forme circulaire, les autres critères doivent être relâchés) ? Ne faudrait-il pas aussi prendre en compte les configurations existantes à d'autres endroits de l'image ? Il est bien évident qu'il n'est pas possible de tout prendre en compte, mais la sélection de ces informations conditionne sans aucun doute la qualité des résultats.

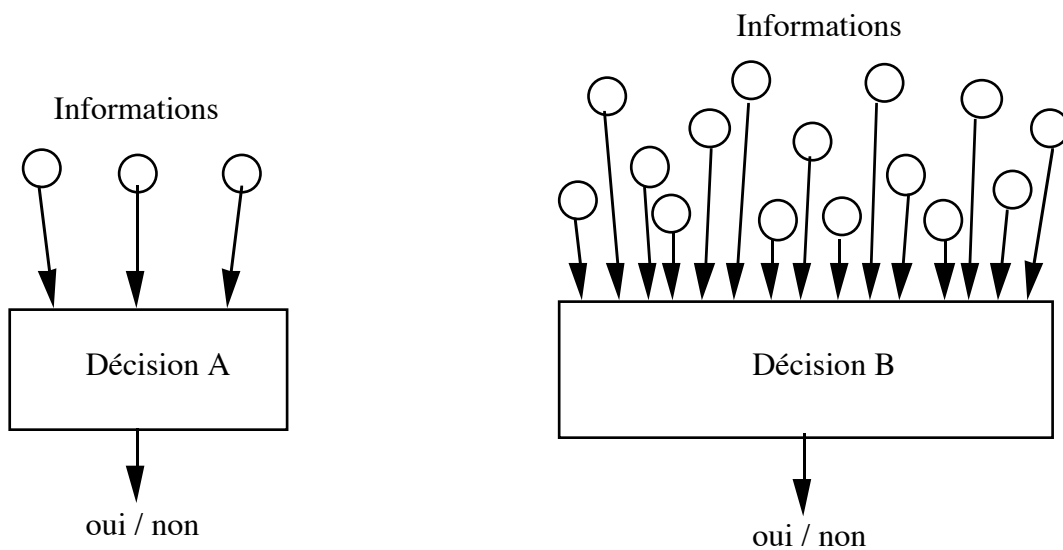


Figure 3 : La qualité du résultat est potentiellement meilleure pour la décision B.

Nous voyons ici que l'abondance des informations qui jouent théoriquement un rôle dans la prise de décision est un des problèmes clés qu'il ne faut surtout pas négliger. Nous pouvons d'ailleurs proposer un nouveau critère d'évaluation qualitative d'un algorithme, relatif aux résultats potentiels qu'il peut obtenir. En considérant que toutes les informations ont à peu près la même importance, un algorithme B a ainsi un meilleur potentiel qu'un algorithme A s'il prend en compte plus d'informations pour prendre la décision fondamentale inhérente au problème (voir figure 3). Bien qu'évident, ce principe est en fait rarement pris en considération lors de la conception de l'algorithme.

• VALIDATION D'UN CONTOUR

Lors de la prise de décision qui détermine la présence d'une discontinuité, les informations prises en compte sont souvent peu nombreuses. Dans la plupart des cas, le problème est ramené au calcul d'une fonction des valeurs de gris dont la réponse est maximale en présence d'un modèle de discontinuité. Cette fonction est souvent un calcul

de gradient avec lissage dans une fenêtre d'observation réduite. Il est difficile d'évaluer le nombre d'informations prises en compte pour décider de la présence d'une discontinuité. Toutefois, il est intéressant de noter que pour éviter la détection de discontinuités dans certaines textures, il paraît logique d'avoir des informations provenant des régions voisines. Ces informations ne sont pourtant pas disponibles dans la grande majorité des détecteurs actuels. De manière générale, la détection et la localisation d'une discontinuité dépendent largement de la présence des discontinuités et des propriétés des régions du voisinage. Or, la plupart des détecteurs effectuent une opération de seuillage global sur le gradient sans tenir compte de ces informations. Il est vrai que la détection des contours est souvent considérée comme une détection de toutes les variations présentes dans l'image et que le problème se ramène alors à trouver le meilleur opérateur de différentiation, après régularisation. Cependant, d'un point de vue gestion des informations, cette définition du contour est critiquable car elle ne correspond pas vraiment au besoin des étapes suivantes de la vision. Cela est d'autant plus vrai que des opérations telles qu'un seuillage par hystérésis, un prolongement des contours interrompus ou une suppression des petites chaînes de pixel-contours peuvent être interprétées comme une remise en cause du modèle choisi et donc de la définition même du contour en tant que forte variation. Selon le même critère d'évaluation qualitative présenté avec l'exemple précédent, l'algorithme de détection de contours que nous présentons au chapitre 2 a en revanche un meilleur potentiel par rapport aux détecteurs classiques. En effet, la quantité d'informations prises en compte pour valider la présence d'une discontinuité est plus grande (voir chapitre 2).

La recherche de toutes les informations théoriquement nécessaires pour prendre la meilleure décision est donc un problème majeur qui mérite une grande attention lors de la conception d'un algorithme. Les décisions étant complexes, il s'agit d'opérer la meilleure sélection des informations qui vont être prises en compte, en évitant de trop simplifier et tout en restant bien sûr dans des temps de calcul raisonnables. Ceci nous amène au premier principe pour la conception d'algorithmes de détection d'indices visuels.

PRINCIPE N°1 : La recherche de toutes les informations théoriquement nécessaires pour décider de la présence d'un indice visuel primaire est un problème majeur qui mérite une grande attention lors de la conception d'un algorithme.

3.2 La complémentarité des informations

3.2.1 Les bases du problèmes

Un des problèmes les plus difficiles lié à la décision est la prise en compte d'informations d'origines diverses. Ainsi, lors de la décision d'agrégation dans un processus de croissance de régions, il faut tenir compte du gradient, de l'écart-type de la région, de la forme qui va être générée etc.... Outre les problèmes inhérents à la différence de nature de ces informations, le problème fondamental qui est sous-jacent est la coopération entre les méthodes qui fournissent ces informations. Nous illustrons nos propos par deux exemples :

- Il n'est pas très intéressant de disposer de l'information du gradient sur un pixel donné ainsi que de l'écart-type d'une région 10x10 centré sur ce même pixel pour valider la présence d'une discontinuité. Ces informations sont de nature différente mais elles ne sont pas vraiment complémentaires. Nous remarquons au passage que la prise en compte d'un grand nombre d'informations ne s'avère efficace que si celles-ci sont pertinentes pour résoudre le problème posé. Il est donc nécessaire de faire coopérer les méthodes pour que soient recherchées des informations spécifiques en fonction des besoins de la situation (voir figure 4).

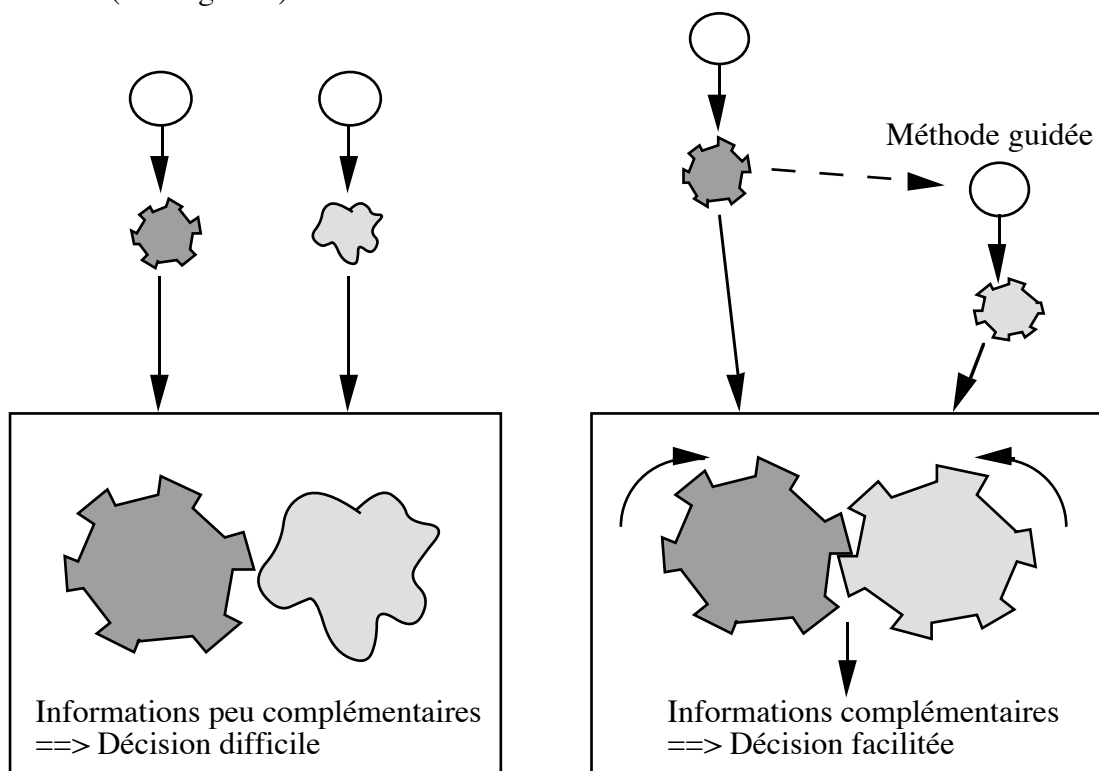


Figure 4 : Pour obtenir des informations complémentaires, il faut une coopération entre les méthodes.

Par exemple, pour valider un pixel-contour, il est plus intéressant de savoir quel est l'écart-type des niveaux de gris, localement, juste à côté du contour hypothétique (en particulier, si l'écart-type est très faible, la présence d'un gradient d'amplitude moyenne valide la disoccontinuité). Pour avoir des informations sur cette région locale, il est possible d'utiliser une technique de croissance de régions, paramétrée et donc contrôlée en fonction des besoins. Cette coopération, qui a été effectivement implantée (présentée au chapitre 3), permet ainsi la fusion d'informations plus complémentaires.

- De même, lors d'un processus de croissance de régions, la fusion entre deux régions dépend souvent des informations sur la moyenne et l'écart-type global des valeurs de gris de celles-ci. Or, lorsque nous effectuons une segmentation "à la main", nous prenons plutôt en compte l'écart-type et la moyenne des valeurs de gris localement, près de la limite entre les deux régions. Ceci nous paraît justifié car la décision de fusion dépend des variations essentiellement locales qui définissent la zone de transition. Pour obtenir ces informations locales, certainement plus complémentaires que les informations globales, il est nécessaire de tenir compte, pour le moins, de la forme et de la localisation de la frontière entre les deux régions, et donc de guider la méthode qui a pour but de fournir ces informations.

Pour accumuler des informations complémentaires, chaque méthode doit préciser ses besoins aux autres méthodes, en déterminant à la fois la manière de les acquérir, le type d'information demandé et la précision requise. Pour cela, une **technique incrémentale** nous semble adaptée. Celle-ci est présentée ci-dessous.

3.2.2 La technique incrémentale

Lorsqu'il s'agit de guider une méthode en fonction des informations (et donc des résultats) qui viennent d'être obtenues par d'autres, c'est le principe incrémental qui est implicitement utilisé. Nous proposons figure 5 une détection de contours classique et une détection incrémentale. La différence majeure réside dans l'exploitation immédiate des résultats intermédiaires et la boucle de retour pour revenir traiter un autre endroit de l'image. L'exploitation immédiate des résultats intermédiaires permet en fait un guidage dynamique des méthodes et une focalisation implicite sur le problème courant. Pour ces raisons, les algorithmes que nous avons développés sont fortement incrémentaux. Il existe bien sûr de très nombreuses variantes exploitant plus ou moins le principe d'incrémentalité. Nous venons de montrer qu'une technique incrémentale est adaptée à nos problèmes, mais il est tout à fait possible de stocker les résultats intermédiaires, de se

focaliser sur un autre endroit de l'image et de ne revenir que bien plus tard exploiter ces résultats pour guider une méthode.

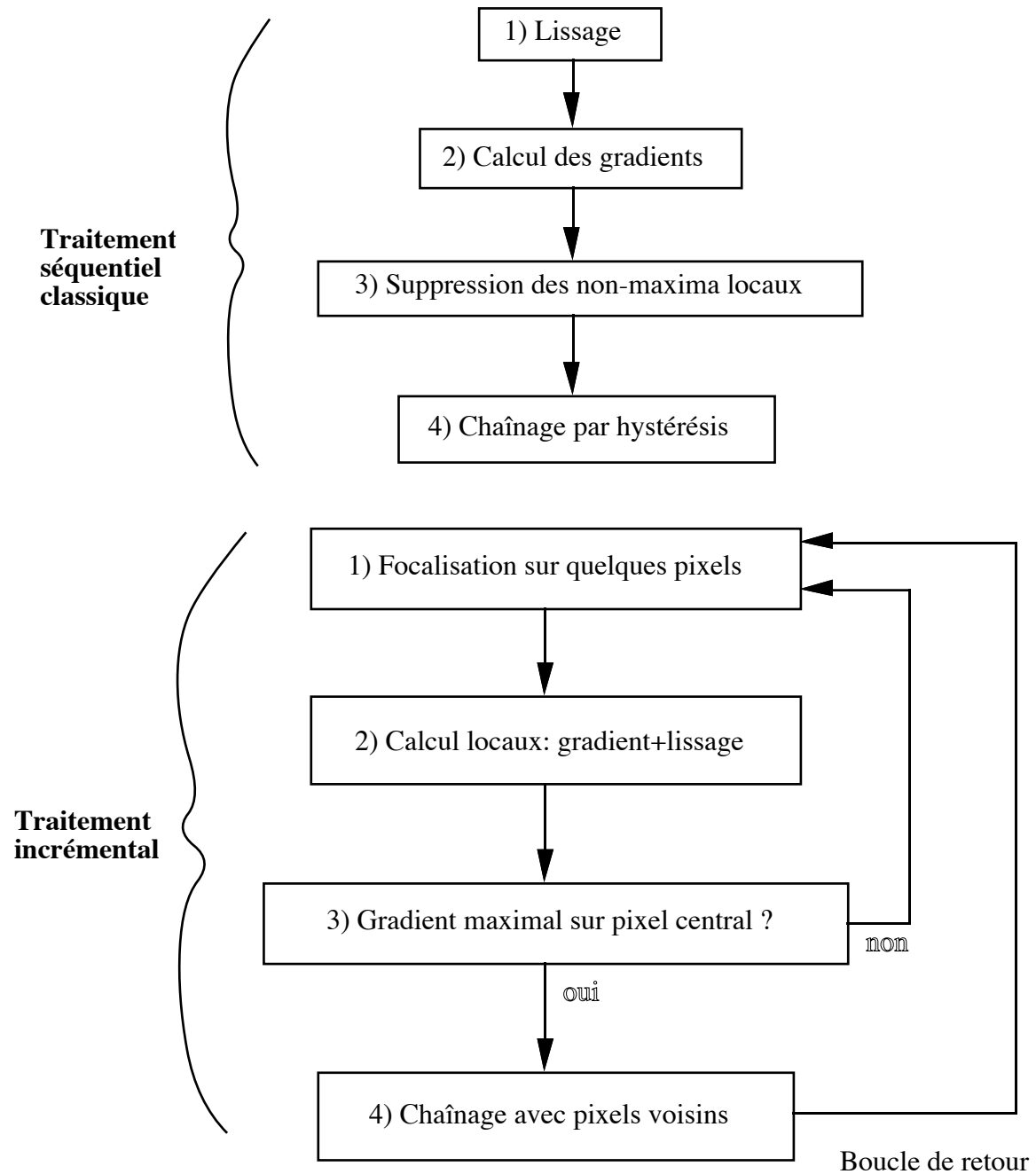


Figure 5 : Exemple d'algorithme classique et d'algorithme incrémental pour la détection des contours.

On peut remarquer que les algorithmes massivement parallèles permettent une exploitation immédiate des résultats intermédiaires tant que chaque processeur reste focalisé sur un problème local. Même si ces algorithmes ne comportent pas de boucle de retour, ils possèdent donc néanmoins les qualités essentielles des algorithmes

incrémentaux. Cette remarque est intéressante si on étudie la vision humaine. En effet, il est actuellement reconnu qu'il n'y a pratiquement pas de boucle de retour vers les neurones correspondant aux premiers traitements visuels. Les traitements étant effectués en parallèle, la boucle de retour qui caractérise l'algorithme incrémental séquentiel n'est effectivement plus nécessaire.

Outre les qualités des algorithmes incrémentaux, retenons donc ici le principe suivant:

PRINCIPE N°2 : Il faut guider les méthodes pour qu'une coopération soit fructueuse et fournisse des informations complémentaires.

3.3 L'émergence de l'information

Le problème de l'émergence des informations est selon nous fondamental, mais rarement apprécié à sa juste valeur, ou mal identifié. Nous commençons donc par définir cette notion, puis nous la discutons et l'illustrons sur plusieurs exemples.

3.3.1 La notion d'émergence

La notion d'émergence issue des travaux d'intelligence artificielle exprime le fait qu'il est nécessaire d'avoir certaines connaissances locales pour arriver à une autre connaissance plus globale. On parle par exemple de "l'émergence d'un comportement global d'un système grâce au comportement local de ses composantes". Il existe deux grandes catégories d'émergence de la connaissance.

- La première est l'émergence dans un système réactif. Un comportement global résulte de plusieurs comportements locaux sans qu'il y ait eu ni intention, ni connaissance explicite du problème global. C'est par exemple le cas des neurones qui interagissent entre eux et qui permettent l'émergence de la pensée et de la vision en ce qui concerne l'être humain, ou plus simplement de l'apprentissage et de la classification de données en intelligence artificielle.

- La deuxième notion exprime l'émergence de l'information attendue par le système, dans notre cas l'indice visuel ou une donnée nécessaire à son identification. Cette deuxième définition diffère de la première au niveau de la connaissance globale du système et de l'intention.

Le problème relatif à l'émergence est essentiellement de collecter des informations "objectives". Nous reprenons ici la définition de "l'objectivité" de Lux [Lux 85].

<< Une information est objective dans la mesure où sa connaissance permet de résoudre des problèmes >>

Si les informations ne sont pas objectives, alors les connaissances ne vont pas émerger correctement et le problème ne sera pas résolu. Cette notion est en fait très subjective et ne peut être étudiée que de façon qualitative. Néanmoins, elle est très importante car c'est la facilité d'exploitation des indices visuels qui conditionne avant tout les étapes suivantes de la vision. Par exemple, les informations de formes peuvent être codées à l'aide de chaînes de points-contours, de segments, de courbes B-splines, de combinaison de différents types de courbes, de squelettes, etc... . Or, si nous voulons tenter une mise en correspondance entre la forme d'un objet connu et une forme présente dans l'image, le choix que nous ferons pour modéliser cette forme va être déterminant pour la qualité du résultat. Le problème est en fait de savoir si telle ou telle représentation est objective et facilite l'émergence des informations de forme ou au contraire si elle les dissimule. A l'extrême, la forme peut également être représentée par la valeur de gris des pixels. Cependant, autant il est aisé de se rendre compte que cette représentation ne fait pas émerger correctement l'information, autant il est très difficile de comparer l'émergence d'une représentation par B-splines et l'émergence d'une représentation par un ensemble de courbes modèles.

Les problèmes d'émergence des informations se retrouvent à tous les niveaux d'un système de vision. En effet, s'il n'existe encore aucun système relativement général de vision, c'est sans doute en grande partie parce qu'il manque des informations indispensables en sortie des processus de description d'images. Autrement dit, il existe des informations qu'il est difficile de faire émerger et qui sont comme dissimulées dans les indices visuels fournis. Nous étudions ce problème sur deux exemples, le premier relatif à la reconnaissance de visages et le deuxième lié aux informations 3D.

3.3.2 La reconnaissance de visages

La vision humaine est très performante pour la reconnaissance de visages. Ainsi, en apercevant une personne pour la deuxième fois, nous sommes capables d'affirmer dans la plupart des cas que nous l'avons déjà vue. Cette reconnaissance implique l'existence d'indices visuels invariants. Cependant, ces indices semblent être très complexes, puisque les systèmes actuels de reconnaissance de visages n'obtiennent pas encore des résultats du même ordre (voir par exemple [NaMM 91]). Le problème est donc de déterminer quelles sont les informations qui correspondent à ces indices visuels invariants et comment parvenir à l'émergence de celles-ci. Sans résoudre le problème de façon explicite, nous pouvons tout de même aller à la recherche de ces indices pour comprendre

ce qui est sans doute la nature de ces informations. En effet, il est remarquable que la vision humaine puisse reconnaître des visages uniquement à partir d'une ébauche des traits les plus marquants du visage. Par exemple, les portraits robots utilisés par la police ou les caricatures de personnages célèbres ne sont que des traits noirs sur du papier blanc, mais cela suffit néanmoins pour que l'identification soit correcte. L'émergence des indices visuels invariants peut donc être ramenée en simplifiant à une association d'informations caractérisant ces traits. Le mot association nous semble ici très important. Il faut noter que de nombreuses approches ont été tentées en décomposant les traits en courbes ou segments, puis en les associant selon les symétries détectées (voir par exemple les travaux de Harmon [HKLR 81] pour la reconnaissance de profils), mais les résultats ne sont pas toujours probants et les applications sont simplifiées. Pourtant, on peut affirmer que certaines associations d'informations relatives à ces courbes ou segments devraient être suffisantes pour résoudre le problème. Ce sont ces informations associatives qui sont dissimulées derrière une description simple des contours et qui n'émergent pas dans nos algorithmes. La solution devrait donc passer par l'étude de relations moins strictes que la symétrie prenant par exemple en compte des informations sur l'éloignement relatif des traits, et conduisant sans doute à des indices visuels d'un niveau plus complexe pouvant à leur tour être associés selon d'autres principes relationnels. Il est d'ailleurs intéressant de constater que nous n'étudions en général que des relations binaires, alors que les neurones de notre cerveau sont souvent connectés à des centaines d'autres neurones. Cette solution théorique montre clairement l'ampleur du problème lié à ce que nous appelons l'émergence de l'information. Tant que les informations indispensables n'émergeront pas correctement de nos processus descriptifs, il est probable que nous n'obtiendrons que des résultats sous-optimaux.

3.3.3 L'émergence des informations 3D

Un autre exemple concerne l'émergence des informations 3D. En effet, même à partir d'un dessin sur du papier, donc sans information sur la position en profondeur des objets, nous (notre perception visuelle) sommes capables de nous représenter la scène en trois dimensions. Cette représentation est nécessairement approximative, et même ambiguë mais elle nous permet d'avoir des indices visuels 3D de grande qualité. Si ensuite nous devons résoudre un problème de reconnaissance, la mise en correspondance est grandement facilitée car les contraintes 3D sont telles que le résultat est pratiquement immédiat et d'une fiabilité exceptionnelle malgré le degré d'incertitude des informations. Comme l'a montré Biederman avec sa théorie "Recognition by Components" [Bied 85], la seule connaissance des contours détectés dans l'image, même entachés d'erreurs,

permet à l'œil humain de reconnaître et de localiser correctement les objets de la scène dans la plupart des cas. Une remarque pertinente pourrait alors être faite : les informations 3D descriptives (même seulement à l'état d'hypothèses) qui peuvent émerger à partir de l'étude d'une seule image sont-elles indispensables pour obtenir un processus de reconnaissance performant ? Citons les propos de Bergevin et Levine qui posent le problème de façon similaire [BeLe 93] : *"This suggests that, at least for generic object recognition, building coarse descriptions from single view edge maps or line drawings might be an appropriate, or even required, alternative to surface and volume reconstruction approaches"*. Il est sans doute utile de rappeler les approches des graphes d'aspects, développés notamment par Ikeuchi [Ikeu 87] où le problème est ramené à la recherche de sous-graphes isomorphes. Les contraintes étant peu nombreuses, cette recherche mène à des calculs difficiles et des résultats pas toujours probants [FMAD 92]. Que faudrait-il donc faire pour remédier à ce problème ? Nous voudrions donner ici quelques idées pour mieux comprendre à quel niveau se situe, selon nous, le problème de l'émergence des informations 3D. Nous proposons trois exemples d'heuristiques utiles pour reconstruire la troisième dimension, dans le cas où la scène se compose d'objets posés sur une table. Notons que des heuristiques du même ordre ont déjà été proposées par Guzman [Guzm 68], et que toutes les approches qui tentent de construire des modèles 3D à partir d'une seule image exploitent nécessairement ce genre d'heuristiques.

1) Les segments à peu près verticaux de l'image sont généralement verticaux dans la scène, si la caméra est posée sur un plan horizontal.

2) Un segment relié à deux segments verticaux est généralement horizontal, en particulier si ce segment est un des premiers situés dans le bas de l'image (car posé sur un plan horizontal). Son orientation 2D permet alors de déterminer son orientation 3D.

3) Un segment à peu près parallèle à un autre possède en général la même orientation.

Il existe ainsi de très nombreuses heuristiques qui peuvent être appliquées pour générer toutes sortes d'hypothèses. Celles-ci génèrent en fait un grand nombre de contraintes qui valident ou invalident les autres hypothèses et il se dégage finalement en pratique une seule représentation 3D satisfaisante. Il faut noter que cette solution ne peut pas être sûre, car aucune information véritablement 3D n'a été utilisée. Toutefois, les contraintes sur le monde réel sont tellement fortes qu'en pratique, cette description 3D de la scène est tout à fait fiable et suffisante pour qu'un processus de reconnaissance puisse être appliqué avec efficacité. Sans exploiter le mouvement, la vision humaine monoculaire confirme sans aucun doute nos propos, car en fermant un œil ou en regardant une photo, nous sommes toujours capables de décrire la scène en trois dimensions avec une qualité de reconnaissance exceptionnelle.

Ce qu'il faut donc retenir ici, c'est qu'il existe des informations qui sont dissimulées derrière notre description trop simpliste de la scène, qui sont difficiles à faire émerger, et qui pourtant si elles étaient prises en compte, permettraient d'obtenir une description fiable en trois dimensions de la scène, facilitant ainsi le processus de reconnaissance.

3.3.4 Conclusion sur l'émergence des informations

A travers le problème de l'émergence des informations, c'est donc l'approche générale des problèmes de vision qui est en question. En ce qui concerne les indices visuels primaires, des problèmes similaires existent. En particulier, les régions fournies par la plupart des algorithmes de segmentation ne sont pas toujours très pertinentes. La difficulté est en fait de combiner à la fois les informations de formes et les informations de niveaux de gris pour faire émerger des régions aux caractéristiques intéressantes. La solution passe sans doute par une explicitation de la connaissance et par un choix "objectif" de structure de données. Bien que la notion d'émergence ait été abordée de façon informelle, nous retiendrons donc qu'il s'agit d'un problème épineux qui conditionne fortement le potentiel de nos algorithmes. Nous en tirons notre troisième principe :

PRINCIPE N°3 : Un modèle de description des indices visuels est suffisamment "objectif", si il permet une exploitation efficace et une émergence d'indices visuels de plus haut niveau.

3.4 La liberté de l'information

Nous définissons la "liberté de l'information" de façon assez intuitive comme la facilité de création, d'accessibilité et de manipulation de l'information. Nous illustrons nos propos par un exemple.

Si nous procédons à un lissage de l'image selon la technique de Nagao ([NaMa 79]) avant d'appliquer un détecteur de contours comme celui de Sobel ([Sobe 78]), il existe de nombreuses informations locales qui sont exploitées dans le premier traitement et qui ne servent plus ensuite. Nous pouvons dire en quelques sortes que ces informations sont "prisonnières" d'une procédure (absence de liberté) et que seule l'image lissée est disponible. Le problème est que ces informations sont du même ordre que celles exploitées par le détecteur et qu'elles auraient donc pu servir à nouveau. La critique que nous venons de formuler est en fait très importante car c'est une des raisons pour laquelle

la coopération entre plusieurs méthodes n'est pas toujours fructueuse. Si la coopération présente de nombreux avantages qui résident dans le cumul et la complémentarité des informations, le risque est de garder également les défauts de chaque méthode si celles-ci sont appliquées indépendamment les unes des autres.

Ceci nous amène directement aux problèmes de la structure de contrôle. En effet, pour faciliter la création et la manipulation d'informations d'origines diverses (liberté de mouvement de l'information), il est nécessaire d'avoir une gestion souple qui autorise une focalisation adaptative, un traitement partiel et un retour sur un problème non résolu. Nous rejoignons ici l'idée d'un traitement incrémental qui permet une exploitation adaptative des informations. Si le principe d'incrémentalité est intéressant, il s'agit en fait de l'utiliser à bon escient selon les besoins. Ainsi, si la gestion des informations est suffisamment souple, il doit être possible de traiter les informations d'une façon séquentielle classique tout en préservant l'opportunité d'un traitement local plus riche.

Sans l'avoir abordé de façon explicite, le problème central est ici le contrôle des décisions à tous les niveaux de l'algorithme. Nous proposons d'ailleurs dans notre rapport de recherche ([Salo 92]) des directives simples liées à ce contrôle, nous les rappelons ici :

- 1) **Retarder la décision** problématique le plus longtemps possible, tant que les informations disponibles ne permettent pas de trancher de façon sûre.
- 2) **Aider la décision** en allant chercher toutes les informations qui sont nécessaires.
- 3) **Adapter la décision.**
- 4) **Centraliser les décisions problématiques.**
- 5) **Décentraliser les décisions simples.**
- 6) **Faire savoir le cas échéant qu'une décision peu sûre a été prise.**

Tous ces principes se rattachent plus ou moins à la liberté de mouvement de l'information et à l'émergence de celle-ci.

Nous concluons en proposant un dernier principe lié à la liberté de l'information :

PRINCIPE N°4 : Pour une gestion efficace des informations, il faut préserver la liberté de mouvement de celles-ci.

4. La conception des algorithmes

Il existe de multiples façons d'aborder un problème et de proposer une solution algorithmique. En ce qui concerne la construction d'un détecteur d'indices visuels, nous distinguons deux grandes catégories d'approches [SaGa 93b].

- La première est l'étude théorique. Elle consiste à effectuer une modélisation du problème et à découvrir la meilleure solution en fonction de critères définis à l'avance.

- La deuxième est expérimentale. Elle consiste à élaborer une stratégie heuristique qui tente d'approcher la meilleure détection possible, en fonction de critères expérimentaux.

Ces deux approches sont fondamentalement différentes. Alors que dans le premier cas, il s'agit de trouver la solution optimale pour un problème et un but donné, dans le deuxième cas, la solution est approximative mais le problème et le but restent imprécis et dépendent de l'évaluation des résultats expérimentaux.

Il existe bien sûr tous les stades intermédiaires, mais c'est en général l'une ou l'autre des deux méthodologies qui est utilisée.

4.1 Etude théorique de l'indice visuel

Pour concevoir un algorithme de détection d'indices visuels, l'approche la plus répandue est de proposer un modèle de l'indice ainsi que des critères à respecter, puis de trouver la meilleure solution théorique au problème posé. La solution algorithmique est alors une traduction simple de la solution mathématique. Par exemple, pour la détection de contours, le modèle de Canny [Cann 86] est une "marche d'escalier" bruitée, tandis que ses critères sont :

- 1) Une bonne détection.
- 2) Une bonne localisation.
- 3) Une réponse unique pour une discontinuité unique.

Geman et *al* modélise le problème de la détection des contours à l'aide d'une fonction d'énergie qui doit être minimisée par une technique de relaxation [GGGD 90]. Pour la détection des régions, Zucker propose une modélisation de la segmentation, qui a été reprise par de nombreux chercheurs. Celle-ci s'énonce de la façon suivante [Zuck 76] :

Etant donné un prédicat P pouvant s'appliquer aux attributs des régions, une segmentation SEG est un partitionnement de l'image en N régions tel que :

- 1) $SEG = \{ R_1, R_2, \dots, R_n \}$
- 2) R_i est une région connexe quel que soit i .
- 3) Chaque pixel appartient à une région unique.
- 4) $P(R_i) = \text{vrai}$ quel que soit R_i

5) $P (R_i \cup R_j) = \text{faux}$ quelles que soient les régions connexes R_i et R_j

- L'intérêt de la modélisation d'un problème est évident. Il s'agit de décrire exactement le cadre de travail, de fixer les limites du problème et d'avancer des hypothèses pour finalement effectuer une analyse mathématique rigoureuse et essayer de dégager une solution. C'est en fait la démarche de base de tout chercheur qui essaie de résoudre un problème. Si aucune modélisation n'est faite, le problème reste mal défini et aucune étude théorique sérieuse ne peut être envisagée. La modélisation permet en fait de donner la spécification complète d'un programme, et d'aborder le problème méthodiquement, au contraire des approches empiriques où le chemin menant à la solution est souvent chaotique.

De manière générale, toute étude théorique des indices visuels implique des choix au niveau du modèle de l'indice et au niveau de la définition formelle du but, ce qui place donc les problèmes d'objectivité de l'information au premier plan. Si cette façon d'aborder les problèmes est couramment utilisée et a fait ses preuves dans de nombreux domaines, elle comporte toutefois certains risques que nous rappelons ici : (voir également notre discussion dans [SaGa 93b]).

- Le risque majeur de toute étude théorique est de travailler sur un modèle trop simple ou insuffisamment objectif de la réalité. Il existe de très nombreuses configurations de pixels et il est souvent difficile de choisir un modèle sans simplifier à la fois les problèmes et les buts. Par exemple, le modèle de contour de Canny est local (il ne dépend pas des régions voisines ni de l'alignement avec des pixel-contours voisins) et l'analyse du modèle est effectuée dans un espace unidimensionnel [Cann 86]. En ce qui concerne la modélisation de Zucker, le non recouvrement des régions est une contrainte très forte dont les implications mériteraient une étude approfondie. Dans toute tentative de modélisation théorique de la détection d'indices visuels, il est donc très important de donner toutes les limites explicites ou implicites que celle-ci implique par rapport au problème général. Si ces limites ne sont pas claires, alors le risque est de dénaturer le problème, en assimilant le but de la détection des indices visuels à la détection d'un modèle de l'indice.

- Une autre difficulté liée à la modélisation de l'indice visuel concerne l'évaluation des résultats et l'amélioration de l'algorithme. En effet, lorsque la solution algorithmique est proche de la détection optimale du modèle, l'évaluation des résultats doit permettre de se rendre compte de la pertinence de celui-ci. Or, le but de la détection des indices visuels tels que les contours ou les régions est mal défini. Comment pouvons-nous d'ailleurs le définir ? Pour une application donnée, il est intéressant d'exploiter un certain type

d'indices et pour une autre application, d'autres indices sont préférables. De plus, pour de nombreuses images, nous ne sommes pas capables de déterminer quelle serait une détection optimale, tout en respectant la prise en compte d'informations strictement locales, donc sans aide contextuelle. Nous pouvons éventuellement comparer deux résultats et préférer celui qui facilite le plus la reconnaissance, mais nous n'avons qu'une vague idée du résultat théorique optimal. La présence de ce flou au niveau de l'évaluation rend difficile le jugement concernant la pertinence de la modélisation. Le modèle du "step edge" en détection de contours a ainsi été adopté par de nombreux chercheurs, tandis que d'autres pensent qu'il n'est pas approprié [Pavl 91], [DIAL 91], sans que personne ne soit parvenu à montrer clairement les avantages ou les défauts de ce modèle. Enfin, si la modification d'une théorie est tout à fait possible dans le cas général, en ce qui concerne les indices visuels, la complexité est telle que nos outils mathématiques sont peut-être insuffisants pour envisager des améliorations notables dans la formulation théorique des problèmes. Comment envisager par exemple une amélioration de la modélisation de Canny ? Bien que cela soit possible, la marge de manœuvre est peut-être limitée.

Pour conclure, nous tenons à souligner que nous ne critiquons pas les études théoriques et les modélisations qui ont été faites. Notre propos est de rappeler les limites et les faiblesses de telles approches, dans le cadre de la détection d'indices visuels.

4.2 Conception expérimentale d'une stratégie heuristique

Lorsqu'un problème est trop complexe pour que toutes les solutions soient examinées, ou que l'étude théorique s'avère trop difficile, une approche classique en intelligence artificielle consiste à approcher la solution idéale à l'aide d'heuristiques (voir par exemple [Pear 87]). La solution trouvée n'est pas toujours la meilleure, mais elle garantit toutefois une qualité minimale. En ce qui concerne la détection d'indices visuels, que signifie une approche heuristique ? Nous l'expliquons de la façon suivante. Le problème est de déterminer la meilleure fonction de décision qui réponde correctement si l'indice est présent ou pas à une position donnée.

Une stratégie heuristique peut très bien s'intégrer dans la modélisation d'un problème. Il s'agit alors d'une fonction de décision qui permet de résoudre une étape précise dont la solution algorithmique exacte est trop complexe ou demande trop de temps de calcul. Par exemple, Gagalowitz et Monga proposent une stratégie heuristique pour fusionner les régions [GaMo 85].

La stratégie heuristique peut cependant être plus globale. Cela suppose que le problème n'est pas clairement défini, avec donc une modélisation restreinte, et que l'algorithme entier est une solution approchée d'une solution théorique trop complexe.

La création ou l'ajustement d'une heuristique se fait généralement en fonction de l'expertise que nous avons du problème. La difficulté majeure de cette approche est la délimitation et la transcription de l'expertise. Pour cela, la démarche classique est la construction d'un système expert qui intègre les différentes règles logiques de notre expertise. Toutefois, en ce qui concerne la détection d'indices visuels, le problème est plus complexe, car d'une part nous n'avons qu'une vague idée de ce qui caractérise un indice visuel, et d'autre part, les relations entre les informations manipulées, telles que niveaux de gris et positions des pixels, sont difficiles à exprimer avec une base de règles. Par conséquent, pour construire un détecteur d'indices visuels à l'aide d'une stratégie heuristique, il nous semble important de tenir compte de deux principes :

1) Il faut pouvoir évaluer les heuristiques, les comparer et choisir la meilleure stratégie. Mais si notre expertise de la détection des indices visuels est approximative, comment parvenir à une évaluation satisfaisante ? Il y a en fait deux aspects à considérer.

- Le premier concerne l'évaluation quantitative des résultats. Il est nécessaire d'adopter une démarche expérimentale rigoureuse pour évaluer et comparer les différentes approches. Le problème est de définir les critères qui vont caractériser la qualité d'une détection. Ces critères dépendent de la qualité de notre expertise, qui est le deuxième aspect à considérer.

- Puisque notre expertise est approximative, le problème majeur est en fait l'enrichissement de celle-ci. Pour cela, nous proposons de privilégier la démarche expérimentale afin d'améliorer l'évaluation qualitative des résultats. En effet, n'ayant qu'une vague idée de ce qui caractérise un indice visuel pertinent (les premières étapes de notre vision se font inconsciemment), il est important de "prendre la place du détecteur" afin de comprendre quelles sont les relations qui existent entre les différentes informations locales et comment les combiner pour discriminer un indice visuel donné. En privilégiant l'aspect expérimental, notre expertise grandit ce qui permet de proposer d'autres heuristiques ou même d'envisager une autre manière de traiter le problème.

2) Si nous adoptons une démarche expérimentale et que notre expertise grandit, il reste encore à expliciter et à transcrire cette expertise dans un algorithme. Or, le problème se prêtant difficilement à une décomposition en règles simples, il est nécessaire d'une part de ramener notre expertise à un traitement méthodique, et d'autre part de concevoir un algorithme performant, capable de traiter des cas complexes avec une structure de

contrôle efficace. Les principes que nous avons dégagés dans ce chapitre concernant la gestion des informations méritent ainsi une attention particulière.

La recherche du meilleur compromis entre notre expertise et sa transcription algorithmique est une démarche empirique à plusieurs niveaux. Après évaluation des résultats d'une stratégie de détection, toutes les étapes de l'algorithme peuvent en effet être remises en cause pour adopter une autre gestion des informations. Cette démarche est à opposer à l'approche classique de modélisation du problème avec étude théorique de la meilleure solution. En effet, il n'est pas possible de donner clairement la spécification de l'algorithme car il faudrait expliciter notre expertise, ce qui est justement le problème à résoudre. Si l'approche empirique a le défaut de ne pas proposer de modélisation du problème, l'avantage se situe au niveau expérimental. Alors qu'une étude théorique peut être biaisée par une évaluation insuffisante, due à une expertise incomplète ou incertaine, l'approche que nous venons de décrire tend à réduire le fossé existant entre ce que nous croyons être la détection des indices visuels, et ce qu'il est réellement possible de faire avec un algorithme. Notons que cet avantage est spécifique de la détection des indices visuels car le problème fondamental est certainement celui de la recherche d'une référence solide pour évaluer les résultats. Cette référence, c'est l'expertise humaine, que nous proposons donc d'approcher par une démarche empirique.

Pour conclure, il est difficile de dégager une démarche précise de la construction d'un détecteur d'indices visuel selon ces principes sans s'appuyer sur un exemple. Nous proposons pour cela deux algorithmes présentés l'un au chapitre 2 et l'autre au chapitre 3.

CHAPITRE 2

La détection des contours

1. Introduction

La détection des contours peut être perçue différemment selon le but recherché. Ainsi, selon Pavlidis [Pavl 92], elle consiste en une détection des discontinuités présentes dans une image. La détection des contours est alors considérée comme une des premières étapes de la vision par ordinateur et aucune information *a priori* sur les objets présents dans l'image n'est utilisée. Toutefois, dans certaines applications, il est utile d'exploiter des informations sur la forme et l'emplacement approximatif de l'objet pour en détecter le contour.

Etant donnée la définition de l'indice visuel primaire que nous avons donnée au premier chapitre, nous définissons un contour comme une rupture ou une discontinuité locale d'une propriété intéressante émergeant d'un groupe de pixels. Il faut noter qu'il n'y a pas isomorphisme entre discontinuité et contour d'un objet et c'est donc par abus de langage que nous parlons de détection des contours.

La détection des contours préparant la reconnaissance, il ne s'agit pas en fait de détecter toutes les transitions de niveau de gris, mais plutôt celles qui permettent une description pertinente de l'image. Ainsi, un des problèmes fondamentaux s'avère être la définition même des discontinuités recherchées et la modélisation qui en résulte.

Nous effectuons dans la première partie de ce chapitre un bref état de l'art, en nous préoccupant des problèmes de gestion des informations que nous avons soulignés au premier chapitre.

Dans une deuxième partie, nous présentons notre détecteur de contours. Nous essayons de justifier chaque étape de l'algorithme, puis nous proposons une étude du critère de visibilité des contours pour déterminer l'heuristique qui décide de leur validation.

Nous procédons enfin à une évaluation de notre détecteur et nous concluons ce chapitre en donnant les perspectives de notre approche.

2. Etat de l'art

Nous commençons cet état de l'art par une étude des techniques de filtrage, suivi d'une discussion. Nous nous focalisons ensuite sur la détection des contours en effectuant une description simple des techniques les plus connues et en présentant différentes méthodes d'évaluation des résultats. Nous terminons cet état de l'art par une discussion générale des problèmes méthodologiques.

2.1 Les techniques de filtrage

Un des problèmes fondamentaux de la détection des contours est la résistance au bruit et aux petites variations locales dues à la texture ou à des irrégularités de surface. Pour remédier à ce problème, il est possible d'effectuer un prétraitement sur l'image afin de lisser les petites irrégularités locales. Ce prétraitement peut être réalisé par des filtres de différentes natures. L'opération de filtrage prend donc en entrée une image et rend en sortie une autre image dite filtrée. Nous en présentons ici quelques exemples, puis nous discutons de leur utilisation pour aider la détection de contours.

2.1.1 Les filtres linéaires

Un filtre est dit linéaire si la nouvelle valeur du pixel de l'image résultat est une combinaison linéaire des valeurs des pixels situés dans une fenêtre de l'image initiale.

Par exemple, un filtre linéaire 3x3 est défini de la façon suivante :

$$P'(x,y) = (1/9) \cdot \sum_{i=-1, j=-1}^{i=1, j=1} A_{i,j} * P(x+i, y+j)$$

où $P'(x,y)$ définit la nouvelle valeur de gris en x .

Cette somme de produits est en fait une convolution de matrices. Il suffit donc de représenter la matrice des coefficients $A_{i,j}$ pour définir un filtre linéaire quelconque.

Si tous les coefficients $A_{i,j}$ sont égaux à 1, on trouve le filtre "moyenneur". Toutes les variations locales sont amoindries et une impression de flou apparaît sur les discontinuités. Ce phénomène est expliqué sur les figures 1 et 2 avec un signal unidimensionnel. Ce filtre entraînant un lissage des discontinuités, il y a une perte d'information sur leur localisation.

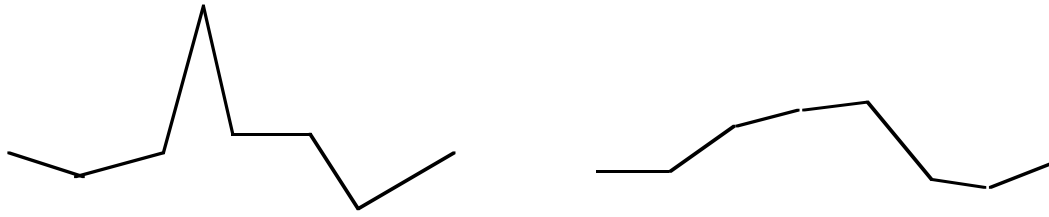


Figure 1: A gauche, signal original, à droite, signal après filtrage par la moyenne.

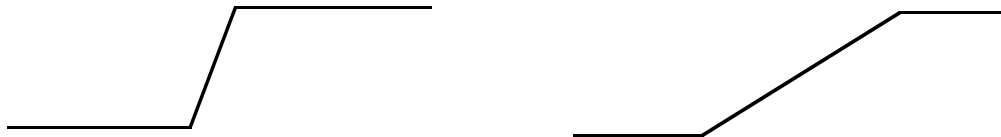


Figure 2: A gauche, signal original, à droite, le résultat après filtrage par la moyenne.

2.1.2 Les filtres non linéaires

Il existe un nombre considérable de techniques. Parmi celles-ci, citons les filtrages d'ordre, homomorphiques, morphologiques et adaptatifs. Une bonne description de celles-ci peut être trouvée dans le rapport du GDR 134 sur les prétraitements et l'approche frontière de la segmentation [GREC 91].

Parmi les filtres d'ordre, le plus connu est certainement le filtre médian. Il consiste à classer par ordre croissant les valeurs de gris dans une fenêtre $(2N+1) \times (2M+1)$ et à choisir celle de rang $((2N+1) \times (2M+1) + 1) / 2$, c'est à dire la valeur médiane de la suite ainsi formée. Les caractéristiques de ce filtre sont très intéressantes. En effet, les irrégularités sont assez bien éliminées et les discontinuités sont préservées. Toutefois, les traits trop étroits disparaissent et les coins sont un peu arrondis. Nous laissons le soin au lecteur de s'en persuader en effectuant à la main les calculs sur de telles configurations. D'autres filtres ont été proposés afin d'améliorer les résultats. Citons par exemple les travaux de Nagao et Matsuyama [NaMa 79]. Un voisinage 5×5 est divisé en 9 domaines connexes qui se recoupent. La moyenne et la variance des niveaux de gris de chaque domaine est calculée. La nouvelle valeur de gris est la moyenne du domaine qui a la plus faible variance.

De manière générale, les techniques de filtrage consistent à modifier les valeurs de gris en fonction de leur distribution locale. La difficulté majeure est de lisser correctement les variations dues au bruit en préservant la force des contours et leur localisation.

Citons également l'existence d'opérateurs de rehaussement de contraste dont le but est de réduire la largeur des fortes transitions sans pour autant augmenter les petites variations locales, voire même en les atténuant. Zamperoni utilise par exemple des modèles d'histogrammes locaux pour typer la situation et choisir la valeur de gris dont le rang est le plus approprié [Zamp 89]. Combinées à des opérations de lissage, les

techniques de rehaussement de contraste sont les outils indispensables de la restauration d'images.

2.1.3 Discussion

Dans de nombreuses applications, une étape de filtrage est utilisée comme prétraitement et améliore de façon sensible les résultats. Notamment, lorsque la nature de l'image est connue avec par exemple la présence d'un fort bruit Gaussien, l'étape de filtrage permet une restauration de l'image sans perdre d'information importante et s'avère d'une grande utilité.

Cependant, dans le cas général, l'utilisation des filtres dans les premières étapes d'un système de vision est controversée. Les résultats de certains détecteurs de contours (du type de Canny par exemple) sont sensiblement meilleurs si on effectue un prétraitement sur l'image. Toutefois, cela pourrait être interprété comme un manque de robustesse du détecteur. Le même genre de remarques peut être fait dans le cas de la segmentation en régions. Les reproches qui sont faits aux filtres sont liés à la perte d'informations qui en résulte. En effet, il existe toujours des configurations particulières de niveaux de gris que l'opération de filtrage ne traite pas correctement et si la segmentation est améliorée dans la plupart des cas, le filtrage entraîne parfois des erreurs irréversibles.

Remarquons que les meilleures techniques de filtrage sont souvent adaptatives. Celles-ci consistent à identifier le type de configuration locale et à appliquer ensuite un filtre adapté. Or, la détermination des configurations locales est souvent très proche d'une identification des pixel-contours. Par conséquent, il suffirait sans doute de peu de choses pour transformer un filtrage adaptatif en détection de contours. Du point de vue gestion des informations, une étape de prétraitement ne devrait pas être indépendante de la détection des contours. Il serait préférable d'exploiter les résultats intermédiaires des techniques de filtrage (principe numéro 1, cumul des informations et principe numéro 4, liberté de mouvement de celles-ci) et d'ajuster les paramètres de la détection de contours de façon dynamique. Bien que cela reste encore à prouver et que le débat soit loin d'être clos, nous pensons qu'un algorithme de segmentation qui se donne les moyens de résoudre les cas difficiles doit pouvoir se passer d'une étape de prétraitement indépendante. Les variations locales irrégulières (par exemple le bruit impulsionnel) qui sont les problèmes majeurs de la segmentation ne sont plus alors considérées comme des "perturbations" mais comme des indices visuels supplémentaires à détecter et à prendre en compte, ce que font justement les techniques de filtrage adaptatifs avant la transformation.

2.2 Les différentes techniques de détection des contours

Les premières idées de modélisation d'une discontinuité étaient très simples. Une discontinuité était assimilée à un fort gradient ou à un passage par zéro du laplacien. Nous montrons ici les deux approches, puis nous détaillons ensuite les modèles de Canny et Deriche et le modèle plus général de Perona et Malik. Nous présentons également d'autres détecteurs pour donner une vue d'ensemble des travaux qui ont été effectués dans ce domaine. Notons que cette étude est presque exclusivement descriptive et que nous discutons des problèmes de gestion des informations relatifs aux détecteurs de contours à la section 2.4.

2.2.1 Gradients et laplaciens

Il existe un nombre considérable de techniques pour calculer le gradient discret. Dans la plupart des cas, le gradient est calculé grâce à un produit de convolution. Il suffit donc de présenter le masque (la matrice des coefficients) pour définir complètement le type de gradient calculé.

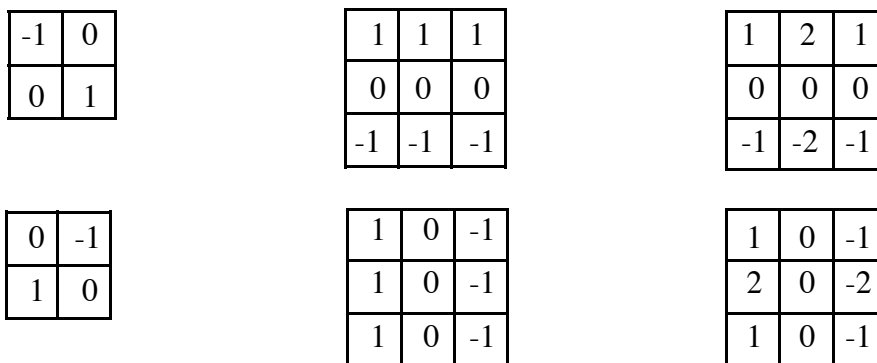


Figure 3 : De gauche à droite, les masques de Roberts, Prewitt et Sobel.

Le gradient discret est un vecteur qui peut être calculé de différentes manières. Chaque composante, calculée à l'aide d'une convolution, donne la valeur du gradient dans une direction donnée. Sur la figure 3, nous présentons les masques 2x2 de Roberts [Robe 65], et les masques 3x3 de Prewitt [Prew 70] et Sobel [Sobe 78]. La norme du vecteur donne la norme du gradient et l'orientation peut être obtenue par une arctangente fonction des deux composantes, selon des principes mathématiques élémentaires. Il faut noter que ce n'est pas un gradient exact qui est calculé, mais un gradient lissé, ce qui entraîne parfois certaines irrégularités. En effet, s'il existe un coefficient du masque non nul situé en dehors de la direction dans laquelle on veut calculer le gradient, alors on peut

montrer que la norme du gradient varie en fonction de l'orientation du contour. Cette particularité est vérifiable sur l'exemple de la figure 4. Alors que le gradient vaut 100 lorsque le contour est horizontal, il vaut 112 dans certaines configurations inclinées, ce qui est paradoxal puisque cela dépasse la différence de moyenne entre les deux régions. Toutefois, la prise en compte de la valeur des pixels voisins est un moyen efficace pour effectuer un lissage implicite et être ainsi moins sensible à toutes les variations hautes fréquences non significatives.

100	100	100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100	100	100
100	100	100	100	100	0	0	0	0	0
100	100	100	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
-	-	-	-	-	-	-	-	-	-
-	0	0	0	0	0	0	0	0	-
-	0	0	0	35	79	100	100	100	-
-	0	35	79	112	112	100	100	100	-
-	100	112	112	56	35	0	0	0	-
-	100	56	35	0	0	0	0	0	-
-	-	-	-	-	-	-	-	-	-

Figure 4 : 1er tableau, valeurs de gris.

2ème tableau, normes des gradients avec les masques de Sobel.

Masques 3x3
inspirés des travaux de Kirsch

1	1	1	1	0	-1	0	1	1	1	1	0
0	0	0	1	0	-1	-1	0	1	1	0	-1
-1	-1	-1	1	0	-1	-1	-1	0	0	-1	-1

Masques 5x5
inspirés des travaux de Kirsch

-1	-1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	0
-1	-1	0	1	1	1	1	1	1	1	-1	0	1	1	1	1	1	1	0	-1
-1	-1	0	1	1	0	0	0	0	0	-1	-1	0	1	1	1	1	0	-1	-1
-1	-1	0	1	1	-1	-1	-1	-1	-1	-1	-1	-1	0	1	1	0	-1	-1	-1
-1	-1	0	1	1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0	-1	-1	-1	-1

Figure 5: Ensemble de masques directionnels.

Une autre façon simple de calculer le gradient consiste à choisir un grand nombre de masques répondant chacun à une orientation différente d'une éventuelle discontinuité. Le masque qui donne la plus forte réponse détermine à la fois la norme et l'orientation approximative du gradient. Nous présentons figure 5 des masques 3x3 et 5x5 inspirés des travaux de Kirsch [Kirs 71].

- Une autre méthode pour détecter les variations locales consiste à repérer les passages par zéro du laplacien ou de la dérivée seconde dans une direction donnée. En effet, à tous les extremums du gradient correspondent des zéros dans la dérivée de celui-ci. Pour ne pas être sensible aux valeurs minimales du gradient, c'est à dire les valeurs nulles correspondant à des zones homogènes, il est nécessaire de détecter les passages par zéro du laplacien et non pas seulement les valeurs nulles de celui-ci. Le problème majeur des techniques exploitant le laplacien est la sensibilité aux petites variations non significatives dûes essentiellement au bruit. Pour résoudre ce problème, il est nécessaire d'opérer un fort lissage, avec par exemple un filtre Gaussien. Marr et Hildreth proposent quant à eux de convoluer directement l'image avec un masque approchant la dérivée d'une Gaussienne [MaHi 85]. Pour plus de sûreté, il est possible d'éviter les passages par zéro dans les zones avec variations locales faibles et irrégulières en acceptant uniquement les pixels dont la valeur de gradient est supérieure à un seuil [WeSk 77].

2.2.2 L'affinage et le seuillage

- Lorsque le gradient a été calculé, un simple seuillage de ces valeurs fournit déjà de bonnes informations sur la présence des discontinuités. Cependant, les contours sont généralement épais, c'est à dire que pour une même discontinuité, plusieurs pixels ont une valeur de gradient supérieure au seuil. Ceci est vérifiable sur la figure 4. Une étape d'affinage des contours est nécessaire. Une des techniques les plus simples consiste à remettre à zéro toutes les valeurs de gradient qui ne sont pas maximales par rapport à celles des pixels voisins situés dans la direction du gradient. En effet, la valeur du gradient doit être maximale localement à la position exacte du contour et décroître de part et d'autre de celui-ci dans la direction du gradient. Bien qu'utilisée pour de nombreux détecteurs, cette technique est à l'origine des trous situés près des jonctions regroupant plusieurs contours [Salo 93]. Nous expliquons ces erreurs à l'aide d'un exemple. Nous présentons figure 6a les niveaux de gris d'un coin d'objet polyédrique et figure 6b la norme des gradients correspondant (calculés avec les masques 5x5 de notre détecteur présenté dans la partie 3 de ce chapitre).

75	70	66	66	63	61	57	56	56	53	51	52	52
81	77	73	68	65	63	62	57	57	53	53	52	51
97	90	83	76	71	67	64	59	56	54	53	51	49
182	157	123	96	85	77	69	61	57	56	53	51	50
244	235	207	172	142	115	84	66	61	54	52	49	51
254	254	248	233	218	193	140	81	59	56	53	51	50
255	254	254	249	242	219	168	96	64	57	53	51	50
255	255	254	254	250	226	164	98	67	55	49	46	47
255	255	254	254	250	219	148	93	69	58	53	50	49
255	255	255	254	248	207	134	89	71	58	53	49	48
255	255	255	254	244	188	119	89	77	60	52	48	47
255	255	254	254	231	163	107	88	75	61	53	47	47
255	255	254	251	215	143	100	86	78	63	52	49	48
255	254	254	247	196	125	97	84	77	66	52	48	43
255	254	253	233	168	113	93	86	76	69	55	44	43
254	254	248	214	144	104	91	82	78	69	55	45	43
254	254	244	193	127	99	84	79	78	66	57	46	43
254	253	235	172	114	96	85	78	77	68	61	48	43
254	250	215	149	108	94	84	80	75	73	65	50	42
254	242	194	128	97	89	83	80	75	70	65	53	43

Figure 6a : Valeurs de gris correspondant à la jonction de trois contours d'un objet polyédrique.

13	12	9	9	7	5	5	4	4	3	0	1	3
24	21	17	13	10	8	8	5	3	4	1	1	3
100	81	58	37	22	15	12	9	4	3	3	3	2
<u>147</u>	<u>144</u>	<u>130</u>	103	76	52	33	17	8	5	4	2	2
76	103	130	<u>140</u>	<u>135</u>	<u>119</u>	89	46	14	6	5	2	2
12	27	55	81	103	119	<u>121</u>	79	27	8	5	2	1
0	3	11	24	43	84	<u>123</u>	<u>96</u>	37	11	7	4	3
0	0	2	7	30	87	<u>125</u>	<u>93</u>	<u>39</u>	15	8	2	1
0	0	0	4	36	101	<u>124</u>	<u>80</u>	<u>36</u>	17	9	4	2
0	0	0	7	50	115	<u>116</u>	<u>61</u>	<u>31</u>	19	9	4	3
0	0	0	14	70	<u>125</u>	<u>100</u>	<u>46</u>	<u>28</u>	<u>22</u>	11	5	2
0	0	1	26	92	<u>125</u>	79	32	<u>26</u>	<u>24</u>	13	5	2
0	0	4	41	110	<u>115</u>	59	24	<u>22</u>	<u>24</u>	15	6	4
0	1	11	64	<u>122</u>	<u>100</u>	42	19	<u>19</u>	<u>24</u>	18	8	4
0	3	25	87	<u>123</u>	79	30	16	16	<u>22</u>	<u>22</u>	11	3
0	6	43	107	<u>113</u>	59	24	13	14	21	<u>23</u>	12	3
1	12	62	<u>118</u>	97	43	20	9	12	20	<u>21</u>	14	4
2	24	85	<u>119</u>	78	32	17	7	9	16	<u>21</u>	18	7
7	42	103	<u>114</u>	60	23	13	8	8	11	20	<u>21</u>	10
14	64	<u>115</u>	97	43	18	10	7	9	10	17	<u>21</u>	14

Figure 6b : La carte des gradients de la région présentée en 6a.

Trois régions peuvent être distinguées : une sur la gauche avec des valeurs de gris souvent égales à 255 (saturation), une sur la droite avec des valeurs de gris décroissant légèrement du haut vers le bas de 55 jusqu'à 43, et une petite région au milieu et en bas avec des valeurs entre 75 et 85. Tous les autres pixels ont des valeurs intermédiaires qui correspondent aux frontières entre les trois régions. Les pixels qui ont des valeurs de gradient maximales dans la direction de celui-ci sont supposés être des pixel-contours, ils ont été soulignés dans la figure 6b. Comme nous pouvons le voir, les pixels marqués en

relief au milieu du tableau ne respectent pas cette propriété, et comme ils n'ont pas été marqués comme pixel-contours, le contour qui correspond à la frontière entre la région de droite et la région du milieu est donc incomplet.

D'autres techniques d'affinage existent. Par exemple, il est possible d'appliquer des filtres morphologiques, érosion et dilatation, ou de calculer le squelette. Ces techniques sont intéressantes mais elles sont parfois moins précises pour la localisation car elles ne tiennent pas compte de la valeur du gradient.

- Un seuillage simple s'avère en général insuffisant. Des faux pixel-contours sont détectés dans les zones bruitées ou les zones de texture (même légèrement) et des pixel-contours importants dont le gradient est faible sont oubliés. D'autres techniques de seuillage existent, nous en détaillons quelques unes.

- Le seuillage par hystérésis est la technique la plus répandue. Proposée à l'origine par Canny en même temps que son détecteur [Cann 86], cette technique consiste à choisir deux seuils s_1 et s_2 (avec $s_1 > s_2$) et à effectuer un chaînage en opérant une sélection des gradients de la façon suivante : l'image est parcourue de haut en bas et de gauche à droite (par exemple). Dès qu'une valeur de gradient dépasse s_1 , elle constitue le départ de la chaîne. On essaie alors de continuer la chaîne dans les deux sens, en fonction de l'orientation du gradient. Tant que le gradient sur le pixel suivant est supérieur à s_2 , on continue la chaîne, sinon une extrémité de la chaîne est atteinte. Lorsqu'une chaîne est terminée, le parcours de l'image reprend là où il s'était arrêté. Tous les pixels chaînés sont bien sûr marqués pour qu'ils ne soient pas traités plusieurs fois.

- Lorsque le problème est l'estimation du bruit pour adapter automatiquement le seuillage, Haddon propose d'évaluer le bruit de façon statistique à partir de deux images prises exactement dans les mêmes conditions [Hadd 88]. Toutes les variations d'une image à l'autre sont essentiellement dues à la digitalisation et au bruit électronique, il suffit donc de calculer la variance entre les deux images pour estimer le bruit et proposer un seuillage adaptatif.

- Zuniga et Haralick propose d'utiliser le "modèle des facettes" pour estimer le bruit local et adapter le seuillage des gradients [ZuHa 88]. L'image est assimilée à une surface échantillonnée dont la troisième dimension est la valeur de gris (voir aussi la section 2.2.6). A partir d'un voisinage local, il est possible d'estimer les paramètres de la surface locale (la facette) ainsi que le bruit par des méthodes statistiques. Le seuil adaptatif du gradient est alors donné à l'aide d'une fonction Bayésienne. Les résultats qu'ils présentent sont intéressants. Toutefois, Zuniga et Haralick se sont restreints à des images synthétiques bruitées avec peu de contours dans l'image.

- Quiguer, Miché et Debrie proposent un seuillage adaptatif à partir de l'analyse de l'histogramme des déclivités [QuMD 91]. Une déclivité est définie comme l'écart entre deux extremums locaux voisins dans une direction donnée, horizontale ou verticale. Ils supposent que l'histogramme doit avoir deux modes, le premier correspondant au bruit, le second aux contours. Le seuil choisi est situé dans la vallée entre les deux modes. Cette technique a été utilisée pour obtenir les contours horizontaux et verticaux de l'image dans un contexte temps réel.

Notons finalement qu'il existe d'autres techniques simples comme un seuillage en fonction de la moyenne des gradients sur l'image ou sur des zones de l'image, un seuillage en fonction de l'écart-type global du gradient ou un seuillage en fonction du pourcentage attendu de pixel-contours par rapport au nombre total de pixels.

2.2.3 Le prolongement, la correction et le chaînage

Un prolongement des contours peut s'avérer nécessaire pour deux raisons. Premièrement, les résultats du détecteur ne sont pas toujours de grande qualité et de nombreux contours tronqués doivent être rallongés et raccordés. Deuxièmement, pour certaines applications, il est intéressant d'avoir des contours fermés pour travailler ensuite sur les formes obtenues. Il existe de très nombreuses techniques pour prolonger les contours. La plupart d'entre elles sont fondées sur la recherche d'un parcours optimal dans un graphe valué. De façon générale, il s'agit d'exploiter les informations sur le gradient, associant intensité et direction, ainsi que l'orientation à l'extrémité pour générer des prolongements jusqu'à ce qu'un critère d'arrêt soit vérifié, comme par exemple la proximité d'un autre contour, un gradient trop faible ou un nombre limite de pixel-contours ajoutés. Des techniques de programmation dynamique, de recherche heuristique, ou de construction d'automates d'états finis ont été développées dans ce but. Il est également possible de raisonner uniquement sur les chaînes de pixel-contours : si deux contours sont suffisamment proches l'un de l'autre et à peu près alignés, ils sont prolongés et raccordés. Une description assez complète des méthodes de prolongement peut être trouvée dans le livre de Ballard et Brown [BaBr 82] ainsi que dans la thèse de Bonnin [Bonn 91].

Le prolongement des contours est souvent considéré comme une étape de correction, puisque les informations utilisées pour rallonger les contours restent essentiellement les mêmes que pour la détection des contours, c'est à dire l'intensité et la direction du gradient. Dans certains cas cependant, les valeurs de gradient ne permettent pas de déterminer la présence ou la fin d'un contour. Un éventuel prolongement ne relève alors

pas de la détection des contours, mais plutôt des techniques d'association inspirées de la théorie Gestaltiste, avec notamment la continuité de la forme.

Parmi les autres moyens de correction, citons également la suppression des petites chaînes de contour, qui bien que très arbitraire, permet d'améliorer les résultats de façon non négligeable.

En ce qui concerne le chaînage, Giraudon [Gira 87] propose un algorithme efficace. Le chaînage est en fait souvent réalisé en même temps que le seuillage par hystérésis, et parfois en même temps que le prolongement. Les seuls problèmes sont d'ordre algorithmique avec le traitement des jonctions et un parcours en une seule passe de la carte des contours.

2.2.4 Le modèle de Canny

Un simple calcul du gradient ou du laplacien s'avère trop simple. De nombreuses recherches ont été faites pour modéliser une transition plus sophistiquée et concevoir un filtre dont la réponse est maximale en présence du modèle. Nous développons ici l'approche de Canny reprise ensuite par Deriche. Cette approche est intéressante car leur méthodologie est caractéristique des techniques de conception d'un détecteur d'indices visuels à partir d'une étude théorique, techniques que nous avons abordées au premier chapitre.

Canny a défini une discontinuité comme une marche d'escalier noyée dans un bruit blanc Gaussien [Cann 86]. Il a d'abord étudié ce modèle dans le cas d'un signal unidimensionnel. Si nous notons l'amplitude de la marche A et la variance du bruit blanc Gaussien V , le signal I peut être décomposé de la façon suivante :

$$I(x) = A \cdot u(x) + V(x) \quad (V(x) \text{ étant la composante du bruit au point } x)$$

Il a ensuite supposé qu'il existait une fonction f antisymétrique dont la convolution avec le signal fournissait une réponse maximale uniquement en présence de la discontinuité. Il a défini trois critères pour évaluer cette fonction :

1) Une bonne détection

Canny a choisi d'exprimer ce critère à l'aide du rapport signal sur bruit. Ce critère est classique dans le domaine du traitement du signal. Sans détailler, retenons ici qu'il obtient une première équation dans laquelle est présente la fonction f . Nous notons SsB la qualité de la détection.

$$SsB = E1(f)$$

2) Une bonne localisation

Ce critère peut être exprimé en fonction de la variance de la position correspondant au passage par zéro du signal. Une deuxième équation faisant intervenir f est ainsi obtenue, dont la particularité est d'être corrélée à la première. Meilleure sera la détection, moins bonne sera la localisation et réciproquement. La qualité de la localisation est notée L .

$$L = E2 (f)$$

3) Une seule réponse pour une seule discontinuité

La distance entre deux pics de la réponse de f doit être la plus grande possible, en fonction de la largeur de l'opérateur. Ce critère donne une troisième équation en fonction de f .

$$X_{\max} = E3 (f)$$

A partir des trois équations $E1$, $E2$, $E3$ et en essayant de maximiser $SsB \times L$, Canny utilise des outils mathématiques complexes pour obtenir finalement une équation différentielle caractérisant approximativement f :

$$2.f(x) - 2.\lambda_1.f''(x) + 2.\lambda_2.f'''(x) + \lambda_3 = 0$$

En appelant W la largeur du filtre, la solution générale de cette équation, définie dans l'intervalle $[-W, 0]$ s'écrit :

$$f(x) = a_1.e^{\alpha.x}.\sin(\omega.x) + a_2.e^{\alpha.x}.\cos(\omega.x) + a_3.e^{-\alpha.x}.\sin(\omega.x) + a_4.e^{-\alpha.x}.\cos(\omega.x) + c$$

avec α , ω et c s'exprimant en fonction de λ_1 , λ_2 et λ_3 , les paramètres a_1 , a_2 , a_3 et a_4 ne pouvant être déterminés que par des conditions supplémentaires sur f . Nous distinguons maintenant deux approches pour exprimer ces conditions supplémentaires : celle de Canny, bien sûr, et celle de Deriche qui a proposé une variante.

- En exploitant la propriété d'antisymétrie ($f(-x) = -f(x)$) et en étendant l'intervalle de définition de f à $[-W, +W]$, Canny utilise les 4 contraintes suivantes pour calculer ces 4 paramètres :

$$f(0) = 0; \quad f(-W) = 0; \quad f'(0) = S; \text{ (S pente à l'origine)} \quad f(W) = 0;$$

Les 3 critères de Canny appliqués à f , maintenant complètement déterminée, peuvent alors être exprimés en fonction de α , ω et β (avec $\beta = S/c$). (Les explications de Canny ne sont pas toujours très claires dans l'article de PAMI 1986, notamment lorsqu'il exprime une de ces contraintes comme $f(1)=0$, mais les calculs étant très complexes, il est légitime qu'il en ait simplifié les explications.) Le meilleur produit $SsB \times L$ est finalement de 1,12. Toutefois, f étant une fonction compliquée, Canny suggère d'approcher ce filtre

par la première dérivée d'une Gaussienne, dont la forme est très voisine. Cette fonction s'écrit :

$$f(x) = - (x / \sigma^2) \cdot e^{-x^2 / (2 \cdot \sigma^2)}$$

En recalculant SsB et L, il obtient un produit SsBxL égal à 0,92. Pour passer à la seconde dimension, il suffit alors de convoluer l'image avec un filtre qui approche la dérivée d'une Gaussienne à deux dimensions et de l'appliquer dans deux directions opposées, comme pour les filtres classiques présentés dans la section 1.2.1.

Le filtre proposé par Canny dépend de σ . Ce paramètre permet de fixer la variance de la Gaussienne et de lisser plus ou moins l'image en fonction du bruit existant.

- Deriche a repris les travaux de Canny [Deri 87]. En supposant que la largeur du filtre est infinie, les conditions qui déterminent a_1 , a_2 , a_3 et a_4 sont modifiées, ce qui conduit, selon lui, à la solution suivante :

$$f(x) = -k \cdot e^{-\alpha \cdot x} \cdot \sin(\omega \cdot x) \quad \text{pour tout } x, \text{ avec } \alpha, \omega \text{ et } c \text{ réels positifs}$$

Toutefois, en détaillant la contrainte $f(+\infty) = 0$, nous obtenons :

$$a_1 \cdot e^{\alpha \cdot x} \cdot \sin(\omega \cdot x) + a_2 \cdot e^{\alpha \cdot x} \cdot \cos(\omega \cdot x) + c = 0$$

ce qui conduit, lorsque x tend vers l'infini à $a_1 = 0$, $a_2 = 0$ et $c = 0$. Or c n'étant pas un paramètre mais une constante, la contrainte imposée par Deriche implique que la fonction f qu'il propose n'est pas une solution générale à l'équation différentielle de Canny. En fait, sa solution vérifie uniquement l'équation générale homogène associée. Cependant, en calculant le nouveau produit SsBxL pour certaines valeurs de α et ω , il arrive à obtenir une valeur proche de 2 ce qui est nettement meilleur que le produit de la fonction idéale de Canny. Il propose ensuite d'implanter son filtre à largeur infinie de façon récursive. La réponse de son filtre à un pixel donné dépend donc de la valeur obtenue sur le pixel précédent. En deux dimensions, la récursivité est appliquée dans deux directions orthogonales (pour plus de détails, voir [Deri 87]).

Notons que nous discutons de la pertinence de cette modélisation au 2.4.

2.2.5 Le modèle de Perona et Malik

La forme du modèle de Canny est relativement simple. Dans la plupart des images complexes, les discontinuités se présentent souvent comme des combinaisons de signaux en forme de "marche d'escalier" de "toit", de "rampes" ou de "pics". Perona et Malik

proposent d'utiliser une combinaison de filtres quadratiques pour détecter n'importe quel type de discontinuité [PeMa 90].

$$W = \sum (I \cdot f_i)^2 \quad (W \text{ réponse, } I \text{ signal et } f_i \text{ filtre linéaire })$$

Ils justifient leur approche en démontrant que quel que soit l'ensemble fini de filtres linéaires utilisé, il existe une discontinuité composite pour laquelle aucun de ces filtres n'obtient une réponse maximale à la position exacte. Pour simplifier les problèmes, ils étudient un signal unidimensionnel composé à la fois de "pics" et de "marches". Ils proposent deux filtres f_e et f_o pairs et impairs (symétriques et antisymétriques) qui permettent d'avoir une réponse maximale correctement localisée quel que soit le contour composite, et sans être sensible aux dégradés réguliers ("shading gradients") comme l'est par exemple le détecteur de Canny. f_e est la dérivée seconde d'une Gaussienne et f_o est sa transformée de Hilbert (leurs transformées discrètes sont notées F_e et F_o). Leur algorithme est le suivant :

1) Détermination de l'orientation θ du contour. Pour cela, l'orientation est d'abord échantillonnée. Ensuite, pour chaque pixel la quantité suivante est calculée :

$$W (x, y, \theta) = (F_{e,\theta} * I)^2 + (F_{o,\theta} * I)^2$$

La plus forte valeur de W détermine l'orientation du contour composite.

2) Suppression des non maxima locaux de W dans la direction de θ .

3) Seuillage

4) Localisation sub-pixel en étudiant la discrétisation du contour composite sur un voisinage 3×3 .

Leurs résultats sont intéressants, notamment en ce qui concerne la localisation du contour. Toutefois, il semble que leur détecteur soit relativement sensible à des contours composites peu pertinents. Il existe en effet des configurations locales particulières où la différence entre deux types de contours est faible. Des pixels dont la forme composite du contour est très différente se retrouvent uniques voisins après l'étape 2, ce qui est très gênant pour une éventuelle étape de chaînage. Plutôt que d'effectuer une suppression irréversible des non maxima locaux, il aurait sans doute été préférable d'utiliser une technique de relaxation ou de suivi de contour, afin que le choix se fasse également en fonction de la nature composite des contours voisins. Notons également le caractère très local de cette modélisation qui ne tient compte que d'un voisinage très restreint.

2.2.6 D'autres détecteurs de contours

Il existe dans la littérature un grand nombre de détecteurs de contours. Nous présentons brièvement quelques techniques.

- En considérant l'image comme une surface en trois dimensions, x , y et le niveau de gris, Haralick exploite la dérivée seconde et l'intensité du gradient pour modéliser la pente de cette surface, appelée facette [Hara 84]. Les contours sont situés sur les plus fortes pentes de cette surface, ce qui correspond aux passages par zéro d'une dérivée seconde en présence d'un gradient non nul. L'avantage de cette technique semble résider dans la modélisation de la facette qui ne présente pas de pente véritable en présence de bruit.

- Bergholm propose une détection de contours pyramidale [Berg 87]. En premier lieu, il s'agit d'opérer un filtrage fort de l'image et de détecter les contours. Ensuite, la carte des contours est raffinée en effectuant une localisation plus précise de ceux-ci à l'aide d'une image un peu moins lissée. Ce processus est réitéré plusieurs fois jusqu'à ce que les contours soient localisés avec précision. Outre la localisation progressive des points-contours, de nouveaux points peuvent être rajoutés à chaque étape en bordure de ceux-ci, afin de combler les trous, notamment près des coins. L'avantage de cette technique est d'obtenir à la fois une bonne localisation et une bonne robustesse au bruit. Cependant, la carte finale des contours dépend beaucoup de la première détection avec fort filtrage. Le risque est de ne pas détecter des contours importants dont le gradient moyen est faible.

- Kundu propose une classification statistique des pixels pour obtenir les contours [Kund 90]. Dans la phase 1, il effectue une série de tests pour déterminer le type du pixel.

Premier test, \max' et \min' étant les secondes valeurs maximales et minimales présentes dans un voisinage $P \times P$ centré sur le pixel, et λ_1 étant un seuil fixé par l'opérateur :

si $(\max' - \min' < \lambda_1)$, alors le pixel n'est pas un point-contour. (zone homogène)

sinon on agrandit la taille de la fenêtre $(P + \Delta P) \times (P + \Delta P)$.

On suppose que deux régions différentes sont présentes dans la nouvelle fenêtre. Grâce à l'histogramme et à des fonctions statistiques, on détermine la moyenne et l'écart-type de ces deux régions, puis le meilleur seuil qui les sépare. On calcule alors les centres de gravité G_1 et G_2 des deux régions. Si deux régions sont effectivement présentes, il doit exister une distance minimale entre les centres de gravité. Kundu propose ainsi le deuxième test suivant, avec λ_2 deuxième seuil défini par l'utilisateur :

$$T = | X_{G1} - X_{G2} | + | Y_{G1} - Y_{G2} |$$

Si $(T < \lambda_2)$, alors il n'y a qu'une région, le pixel n'est pas un point-contour (présence de bruit). Sinon on effectue un dernier test pour savoir si les deux régions sont

équitablement représentées. Si les deux régions ont à peu près le même nombre de pixels (par rapport à un troisième seuil), alors le point-contour est accepté, sinon il est refusé.

La phase 2 correspond à un affinage des contours et la phase 3 est une correction de la classification pour rejeter les points-contours isolés et combler les trous.

Cette technique, qui pourrait être qualifiée d'heuristique, est très intéressante. Le nombre de calculs et de tests étant faible, elle est très rapide et fournit des résultats relativement robustes.

- L'utilisation des champs de Markov pour la détection de contours a conduit à de nombreuses techniques [GREC 91], [GGGD 90]. Le principe est d'associer à chaque pixel une fonction d'énergie qu'il faut minimiser. Cette fonction exploite par exemple des informations sur le gradient, sur la présence d'autres pixels-contours dans le voisinage immédiat et un coût pour changer le label de pixel simple en pixel-contour. En utilisant des techniques de relaxation (par exemple le "recuit simulé"), l'énergie globale du système est minimisée petit à petit jusqu'à ce qu'un critère d'arrêt soit satisfait, en fonction de la qualité des résultats et de leur convergence.

- Grossberg et Mingolla exploitent les réseaux de neurones pour de nombreuses étapes de la vision, dont la détection des contours [GrMi 87]. L'idée est de faire apprendre automatiquement à partir d'une base d'exemples les poids synaptiques permettant une classification correcte en pixel simple ou pixel-contour. Ces réseaux de neurones artificiels tentent de simuler les réseaux de neurones situés près de la rétine. Un des problèmes les plus délicats de cette technique est le choix des exemples.

- La détection des lignes de crête a suscité également de nombreux travaux.

Chen propose par exemple une détection des contours de type ligne et de type marche en deux étapes [Chen 92]. Dans un premier temps, une étude de blocs 9x9 permet une classification des pixels en trois grandes catégories { homogène, contour ligne potentiel, contour marche potentiel }. Ensuite, deux algorithmes de détection spécifiques, l'un dédié à la reconnaissance de lignes, l'autre à la reconnaissance de contour de type marche, sont appliqués aux pixels en fonction de leur catégorie.

Citons aussi l'approche de Cox, Boie et Wallach qui ont modélisé une ligne de crête un peu de la même façon que Canny et qui ont intégré leur détection de ligne à une détection de contours de type "marche d'escalier" [CoBW 90]. Ziou propose également une solution en reprenant les critères de Canny appliqués à un modèle de ligne de crête bruitée [Ziou 91].

2.3 L'évaluation des contours

L'évaluation des algorithmes de détection de contours est problématique. En effet, dans la plupart des cas, non seulement il n'est pas possible de déterminer de façon précise quel est le résultat optimal, mais les critères que doivent satisfaire la description idéale sont généralement *ad hoc* et subjectifs.

Il existe trois grandes catégories d'évaluation des détecteurs de contours :

- La première, fondée sur l'expertise humaine, consiste à évaluer visuellement les résultats.
- La seconde utilise des mesures statistiques. Une fonction spécifique permet de calculer une distance entre une carte des contours résultats et une carte idéale des contours.
- La troisième, fondée également sur l'expertise humaine, consiste à évaluer non seulement l'image, mais aussi les valeurs numériques correspondant aux niveaux de gris.

2.3.1 Evaluation visuelle

L'image d'un visage humain est présentée page suivante avec le résultat d'un détecteur de contours. Quels critères pouvons nous utiliser pour apprécier visuellement ces résultats ?

- Intuitivement, nous pourrions considérer qu'une carte des contours idéaux devrait contenir au moins les contours du chapeau, de la bouche, du nez, du menton et du visage. Si des contours étaient détectés dans les régions du cou ou des joues, nous pourrions considérer que ce seraient de mauvais résultats. Cependant, nous nous sommes placés dans un cadre bien précis (Cf introduction) où le but de la détection des contours n'est pas de détecter les frontières des objets de la scène, mais de décrire l'image en localisant les discontinuités importantes qui y sont présentes. Par conséquent, nous ne devons pas tenir compte d'informations sémantiques comme les caractéristiques essentielles d'un visage. Cette évaluation ne peut être retenue.

- Nous pouvons essayer d'évaluer les contours en étudiant visuellement les variations de niveaux de gris et en faisant abstraction des objets présents dans la scène. Cependant, cela est-il réalisable ? Sommes-nous capables de définir ainsi des résultats optimaux ? De nombreux problèmes se posent. Par exemple, des petites discontinuités semblent être présentes dans la région du cou. Est-il nécessaire de les détecter ? Quelle est la plus petite discontinuité acceptable ? Un léger dégradé est présent dans la région de la joue. Est-ce une discontinuité ?

EVALUATION VISUELLE DES CONTOURS :

Exemple avec l'image de la femme.

La frontière entre le menton et l'arrière-plan est floue par endroits. Sans parler des problèmes de localisation, si nous considérons qu'une discontinuité doit être trouvée à cet endroit, y a-t-il un risque de détecter de nombreuses configurations similaires ailleurs ou dans d'autres images qui ne correspondent à aucune frontière intéressante ? Même sans prendre en compte d'informations sémantiques, il est donc très difficile de se faire une opinion sur la qualité des contours.

Dans la plupart des cas, nous avons une vue globale de l'image et une idée très approximative de ce qui doit être détecté. Les détecteurs de contours étant nécessairement sensibles aux variations locales, notre niveau d'observation n'est pas approprié. De plus, l'évaluation visuelle dépend fortement de la qualité de l'affichage. Par exemple sur les écrans Sun dont nous disposons, le contraste est insuffisant dans les parties sombres de l'image, lorsque les valeurs de gris sont inférieures à 50. Les difficultés sont beaucoup plus significatives lorsque l'image est imprimée sur du papier, puis photocopiée. Il suffit de regarder les actes des conférences ou les revues de notre discipline pour se rendre compte que la qualité des images est très médiocre. Finalement, nous sommes forcés de conclure qu'une telle présentation des résultats n'est pas adaptée. L'évaluation visuelle des contours permet de constater que l'algorithme fonctionne, mais il est très difficile de comparer deux détecteurs, si aucun des deux ne fait d'erreur grossière, comme le montre les résultats du rapport GRECO sur l'approche frontière [GREC 91].

2.3.2 Evaluation statistique

Pour effectuer une évaluation statistique, il faut d'abord choisir des images tests relativement simples, pour lesquelles il est possible de déterminer la carte idéale des contours (voir par exemple page suivante). Ensuite, différents détecteurs de contours sont appliqués à ces images ou à des versions bruitées et les résultats sont comparés en fonction de critères choisis à l'avance.

Par exemple, Peli et Malah [PeMa 82] compare des détecteurs sur des images de carrés et de cercles plus ou moins bruités, attribuent une note "Perfect edge", "Broken edge" ou "Perfect but Broken at critical points" et calculent également des fonctions d'erreurs comme la moyenne des écarts carrés entre le pixel contour détecté et le pixel contour idéal.

Plus récemment, toujours à partir d'images bruitées et d'une carte de contours de référence, Spreeuwens et Heijden proposent de calculer une fonction qu'ils appellent "Average Risk" qui prend en compte à la fois les erreurs de détection et les erreurs de localisation [SpHe 92].

EVALUATION A PARTIR D'UNE IMAGE DE SYNTHÈSE BRUITÉE :

Haralick propose une méthodologie générale d'évaluation, dont le principe est de déterminer la valeur seuil critique d'un paramètre déterminant, à partir de laquelle les résultats sont en dessous d'un critère de qualité [KJPH 93]. Pour la détection des contours, ce principe d'évaluation pourrait par exemple consister à trouver le rapport signal sur bruit à partir duquel un détecteur aurait moins de 90% de réponses exactes. Nous ferons deux remarques sur ces techniques d'évaluation.

- En premier lieu, même dans l'image bruitée, il n'y a ni dégradés locaux, ni contours étroits puis très larges, ni contours "step+roof", ni contours ambigus. En effet, un bruit artificiel ne peut être équivalent à une texture, une ombre, une tâche, une surface plus ou moins courbe, un coin arrondi, un reflet ou un objet de petite taille. Les problèmes auxquels sont confrontés généralement les détecteurs de contours ne sont donc pas présents dans les images proposées dans [PeMa 82] et [SpHe 92]. L'évaluation de Haralick est également très difficile à appliquer, car il n'existe pas un mais de très nombreux paramètres complexes qui interviennent dans la détection des contours.

- Deuxièmement, si nous examinons l'image bruitée, il n'est pas sûr que la carte idéale des contours reste inchangée. En particulier, la position des contours devient ambiguë. Les résultats sont ainsi subjectifs et l'évaluation statistique pourrait s'avérer trompeuse. Dans le cas extrême, si le contour de l'image originale non bruitée était chaotique, le détecteur le plus robuste détecterait peut-être une ligne droite dans l'image bruitée ! Par conséquent, la version bruitée d'une image est une nouvelle image et il n'est pas prouvé que la carte idéale des contours doive être identique pour les deux.

Même avec des images de synthèse, l'évaluation est donc insuffisante et le problème de la référence reste présent.

2.3.3 Evaluation pragmatique

Dés que l'image est un peu complexe, qu'elle soit réelle ou artificielle, le problème de la référence idéale se pose. Or, toutes les techniques d'évaluation statistiques dépendent de cette référence. Le cœur du problème ne réside donc pas à notre avis dans le choix du critère statistique, mais plutôt dans la manière par laquelle la carte idéale des contours est obtenue.

Si l'image est complexe, l'assistance d'un expert humain est nécessaire dans tous les cas pour évaluer le résultat ou pour construire une carte de référence approximative. Cependant, il n'est pas possible d'exploiter l'expertise visuelle humaine puisque nous ne

sommes pas conscients de son fonctionnement. Placé devant une image en niveaux de gris, il est très difficile pour nous d'expliquer et de justifier comment certains détails locaux contribuent à l'émergence des informations. Ce problème n'est pas dû à une déficience des capacités humaines de raisonnement, mais résulte plutôt d'une mauvaise présentation des informations.

Il existe pourtant un moyen d'effectuer une évaluation relativement efficace. **Nous proposons en effet une approche pragmatique qui consiste à focaliser notre attention sur les valeurs numériques de gris plutôt que l'image**, ce qui correspond au niveau d'observation le plus approprié, pour un facteur d'échelle donné. Il est alors possible d'apprendre, de manière qualitative, ce qui caractérise les indices visuels et comment les informations locales, globales ou contextuelles peuvent être exploitées afin de les détecter. Grâce à nos facultés de raisonnement, nous sommes capables de procéder à une évaluation qualitative des détecteurs de contours, et de proposer de nouvelles idées pour les améliorer. Cette méthode de travail est déjà largement répandue, mais il est cependant rare que les résultats soient présentés avec le tableau des valeurs numériques correspondant aux niveaux de gris, ce qui est dommage.

Dans la partie évaluation de ce chapitre, nous présentons à plusieurs reprises de tels tableaux pour illustrer des difficultés typiques de la détection des contours. De manière générale, nous avons très souvent recours à la visualisation des valeurs exactes des niveaux de gris afin de mieux comprendre quels sont les obstacles et les erreurs de nos algorithmes. C'est à notre avis l'évaluation la plus riche et la plus fiable qui soit. A partir de notre expertise des valeurs numériques, il est possible de construire une carte des contours de référence et d'effectuer des mesures statistiques. Cependant, il est parfois difficile de se prononcer sur la présence d'une discontinuité ou de la localiser. Toute mesure restant donc délicate, nous pensons que l'évaluation des détecteurs de contours doit rester qualitative. L'analyse des niveaux de gris permet de mieux comprendre les mécanismes qui rentrent en jeu dans la détection des discontinuités, ce qui nous renseigne finalement bien plus sur les erreurs et les faiblesses des algorithmes qu'une évaluation statistique.

2.4 Discussion

2.4.1 Les petites phrases

Nous reprenons ici les propos de plusieurs chercheurs qui ont commenté différentes approches et méthodologies pour la détection des contours. Cet aperçu est intéressant, car

il montre comment sont abordés les problèmes ainsi que les clivages entre différentes équipes de recherche.

- Torre et Poggio, justifiant un lissage au préalable de l'image [ToPo 86].

<< Edge detection is the process that attempts to characterize the intensity changes in the image in terms of the physical processes that have originated them. ... Thus, we consider this part of edge detection as a problem in numerical differentiation. We show that numerical differentiation of images is an ill-posed problem in the sense of Hadamard. Differentiation needs to be *regularized* by a regularizing filtering operation before differentiation. >>

- Shen et Castan, présentant leur filtre optimal [ShCa 92].

<< As differential operators are sensitive to noise, a pre-processing such as smoothing is often necessary. ... In the present paper, we deduce the optimal linear operator for edge detection from the point of view of signal processing. >>

- Haralick et Lee, en présentant un détecteur dépendant du contexte [HaLe 88].

<< The solution to edge detection on noisy images should not be image smoothing, because image smoothing alone tends to blur edges. The best solution, we believe, is to incorporate world knowledge and edge context information into the edge detection process. >>

- Kundu, appuyant les propos de Haralick (sauf l'utilisation du contexte) avant de proposer son nouveau détecteur présenté à la section 2.2.6. [Kund 90].

<< Canny's optimal operators use smoothing before edge detection like Marr's operator. This can blur weak edges; and the presence of impulse noise can adversely affect the performance of the edge operator. >>

- De Micheli, Caprile, Ottonello et Torre en étudiant la localisation et la résistance au bruit des détecteurs exploitant le gradient ou le laplacien [DCOT 89].

<< A simple but important conclusion of this correspondence is that the noise present in digital images of typical indoor scenes is small and the signal-to-noise ratio is high. In fact, the noise present in digital images is so small as to make the performances of a variety of filters almost indistinguishable. As a consequence, small filters can be used and the exact shape of the filter is not critical. >>

- Perona et Malik, en proposant une nouvelle modélisation [PeMa 90].

<< Unfortunately, step edges are an inadequate model for the discontinuities in the image that result from the projection of depth or orientation discontinuities in physical scene. ... As a consequence of these effects, real image edges are not step edge functions, but more typically a combination of step, peak and roof profiles. >>

- Rapport du GDR 134 du CNRS sur l'approche frontière en segmentation d'images [GREC 91].

<< La plupart des opérateurs présentés, à part ceux de l'approche Markovienne, sont inadaptés pour les images texturées. >>

- Bonnin, en commentant l'organisation des détecteurs de contours [Bonn 91].

<< L'organisation de la plupart des méthodes de segmentation en contour est linéaire : elle est constituée de diverses étapes, mises en cascade. C'est certainement une raison des échecs obtenus sur des parties de certaines images. ... Malgré ces considérations, cet état n'a pas changé dans les méthodes proposées depuis. >>

Les étapes citées par Bonnin sont l'extraction d'éléments locaux de contours, l'affinage des contours, le suivi et la prolongation des contours ainsi que le chaînage.

- Jain et Binford, en dénonçant l'ignorance, la naïveté et la myopie de certains travaux en vision [DIAL 91]. Nous citons plusieurs passages très critiques de leur "dialogue".

<< Examples of techniques that have difficulties in the presence of discontinuities are optimization techniques, such as regularization. ... These mathematically elegant and rigorous techniques have very limited use with real images, because they can not handle discontinuities and rely on optimization criteria that are not very meaningful in most situations due to the restrictive assumptions made for the formulation of a mathematically tractable problem. >>

<< Some of the popular general models used in early vision are "step edges", "smoothness of ..", "rigidity of objects". All these models are justifiable in limited contexts, but their popularity is more due to their generality and mathematical tractability than their suitability in computer vision....What is interesting is that we are reluctant to explore other models or other approaches to boundary detection, though research efforts during the past decade have shown that edge detection using the step model is inadequate. >>

- L'article de Jain et Binford a suscité de nombreuses remarques, nous en citons une, celle de Aloimonos et de Rosenfeld [DIAL 91].

<< The authors' criticism of regularization techniques is not justified. The growing attention of some researchers to regularization and their functional analysis techniques is a significant event, for several reasons. ... Regularization theory unified various approaches to segmentation, and has helped us understand why segmentation is difficult. >>

- Fleck, en analysant les résultats des détecteurs de Canny, Marr-Hidreth et Boie-Cox [Flec 92].

<< Perhaps the most surprising result of this study is the similarity of the algorithm outputs. In particular, the modernized Marr-Hildreth algorithm seems to perform as well as Canny's. Some of the differences between previous implementations seem to be due to differences in thinning, hysteresis, or subpixel interpolation algorithms, as well as differences in thresholds settings. A few may be due to small differences in the shape of

the smoothing filters or the derivative approximations, but there is no direct evidence to support this. Very different methods of suppressing noise could clearly have larger effects. >>

Fleck fait allusion à une méthode originale pour tenir compte du bruit, qui exploite des filtres unidirectionnels de différentes tailles effectuant des différences simples de niveaux de gris [Flec 90].

- Pavlidis, en essayant d'expliquer la lenteur des progrès en vision [Pavl 92].

<< I have come to the reluctant conclusion that a major reason that our rate of progress is slow is because we try to solve problems with models established many years ago that are oversimplified and unrealistic. ... This conclusion became apparent to me while I was comparing two edge detectors. It seemed that the difference between the two detectors when tested on the same image was relatively minor compared to the results obtained on different images using either detector. On some images both detectors did very well and on others both did very poorly. ... **Sb** rather than blaming an edge detector for poor results, we should blame ourselves for using the wrong tools. >>

A la vue de ces commentaires, la façon d'aborder la détection des contours n'est pas la même pour toutes les équipes de recherche. De très nombreuses techniques ont déjà été proposées, mais aucune ne semble satisfaire l'ensemble de la communauté en vision. Nous pourrions même ajouter que de nombreux désaccords existent sur le fond et sur la forme. Il n'est pas question ici de prétendre trouver la solution miracle. Nous proposons en revanche une étude des propos qui ont été tenus à partir des principes que nous avons présentés sur la gestion des informations.

2.4.2 La critique du lissage

La critique des méthodes de régularisation (Jain et Binford) a peut-être son origine dans les problèmes liés à l'émergence des informations. Si nous nous focalisons essentiellement sur les techniques effectuant un lissage de l'image en même temps que la détection, alors nous remarquons que cette étape ne permet pas (en général) de faire émerger les informations sur le bruit. Il n'existe en effet aucun paramètre permettant de savoir quelle est la distribution locale des variations de niveaux de gris. Quelque soit le lissage utilisé, s'il n'existe aucune information locale pour guider celui-ci, il est donc inévitable que des contours dont le gradient est faible risquent d'être perdus. Il faudrait donc effectuer un filtrage adaptatif. Cependant, comme nous l'avons déjà dit, un filtrage adaptatif nécessitant pratiquement une détection des contours au préalable, la meilleure solution est de sophistiquer la technique de filtrage adaptatif pour la transformer en

véritable détection de contours. L'approche de Kundu [Kund 90] nous semble le type même de détecteur dont les différentes étapes s'apparentent à une identification de la situation comme le ferait une technique de filtrage adaptatif. Cette identification de la situation correspond à une tentative pour faire émerger les informations et adapter le processus. En effet, Kundu peut facilement transformer son programme pour donner un label à tous les pixels "homogène", "texture" ou "contour", ce qui n'est pas possible avec les détecteurs comme celui de Canny ou de Marr-Hildreth. Nous pourrions en quelques sortes opposer les techniques de transformation du problème (régularisation) aux techniques d'adaptation aux problèmes, qui considèrent le bruit ou la texture comme des indices à détecter et à prendre en compte. Nous ne démontrons pas ici que les techniques de régularisation manquent d'efficacité, notre propos est uniquement de montrer que l'utilisation de telles techniques comportent certains risques car, en général, les informations n'émergent pas de façon satisfaisante, ce qui peut être préjudiciable pour toute détection de contours qui se voudrait localement adaptative.

2.4.3 La critique du modèle "step edge"

La deuxième critique, formulée entre autres par Pavlidis et Jain-Binford, concerne le modèle "step edge". Canny a proposé de définir un modèle de marche d'escalier noyé dans un bruit blanc Gaussien qui a ensuite été repris par de nombreux chercheurs. Le reproche qui est fait à ce modèle est sa trop grande simplicité et son inadéquation à résoudre certains problèmes classiques de détection de contours, ce qui correspondrait à un manque d'objectivité du modèle. Il est remarquable que de nombreux autres détecteurs de contours aient été proposés depuis, avec comparaison des résultats à ceux du détecteur de Canny en montrant quelques améliorations (nous présentons nous mêmes des comparaisons avec le détecteur de Deriche qui est une variante de celui de Canny). Il est donc vrai qu'il existe de nombreux détecteurs au moins aussi performants que celui de Canny et que le modèle de marche d'escalier bruité est sans doute trop simple. Toutefois, les travaux de Canny font toujours référence car sa solution est correctement formalisée et le comportement du filtre est ainsi connu avec précision.

Mais quelles sont les limitations de ce modèle ? Nous voyons quatre points faibles dans ce modèle :

- Le premier est relatif à la forme "marche d'escalier". Il arrive très souvent que la forme du signal soit beaucoup plus complexe, comme l'ont montré Perona et Malik [PeMa 90].

- Le deuxième point faible est le caractère isolé de la marche d'escalier. Une discontinuité pertinente dépend en fait fortement de la présence et de l'alignement des discontinuités du voisinage. La plupart des modèles proposés sont pratiquement unidimensionnels et ne tiennent pas compte de cette propriété. Notons que le succès des techniques de relaxation tient sans doute justement à la prise en compte des discontinuités du voisinage.

- Le troisième point faible concerne l'hypothèse du bruit blanc Gaussien. Selon De Micheli et al [DCOT 89], le bruit est relativement faible dans la plupart des images, ce qui est effectivement vérifiable en regardant les valeurs numériques de gris de celles-ci. Le problème n'est donc pas une sensibilité au bruit, mais plutôt une sensibilité à la texture (texture simple sans motif visible), comme il est indiqué dans le rapport du GDR 134 [GREC 91]. Or, une zone de texture ne couvre généralement pas toute l'image et ne peut certainement pas être assimilée à un bruit blanc Gaussien. Un lissage plus important n'étant pas justifié dans un tel cas, cette hypothèse entraîne la détection de nombreuses discontinuités ne correspondant à rien de visible, ce qui est gênant pour les étapes suivantes. Nous rejoignons ici les critiques de la section précédente.

Ces trois points faibles sont significatifs d'un manque relatif d'objectivité du modèle choisi (troisième de nos principes développés au premier chapitre).

- La quatrième limitation du modèle de Canny concerne les configurations particulières, comme la promiscuité d'un ou de plusieurs autres contours, ainsi que les coins. Voir l'article de Fleck [Flec 92] ainsi que celui de De Micheli et al [DCOT 89] pour les problèmes de localisation. Il manque ici des informations complémentaires pour traiter ces cas particuliers, ce qui correspond à une insuffisance au niveau de nos deux premiers principes (cumul des informations et adaptabilité en guidant les méthodes).

2.4.4 L'enchaînement des étapes

La critique concernant l'enchaînement des étapes (Bonnin [Bonn 91]) est liée à la critique du modèle. Si le modèle choisi au départ était satisfaisant, un simple seuillage suivi d'une étape de chaînage fournirait les résultats espérés. Or, en général, il faut une étape d'affinement des contours, la technique de seuillage utilisée est souvent le seuillage par hystérésis, il y a parfois des corrections de la carte des contours par prolongements de ceux-ci et il y a même suppression des chaînes de contours de faible longueur, toujours dans les premiers traitements, donc au bas-niveau de la vision.

L'enchaînement de telles étapes ne se justifie que si le modèle de contour n'est pas satisfaisant. Ainsi, la première étape de détection entraîne une succession d'étapes peu coordonnées et surtout très imparfaites, où de nombreuses décisions quelques peu arbitraires sont prises.

- Par exemple, en utilisant une technique de seuillage par hystérésis, nous essayons souvent cinq ou six valeurs différentes de seuils pour arriver au meilleur compromis sur une image spécifique et nous choisissons des valeurs différentes pour une image de la même scène, mais prise d'un autre angle. De plus, de meilleurs résultats seraient obtenus si nous pouvions donner des valeurs de seuil différentes pour chaque zone de l'image.

- Un des outils les plus performants pour la correction des contours est sans doute la suppression des petites chaînes. En effet, de nombreux contours apparaissent dans les zones texturées, mais généralement de faible longueur (si la texture n'est pas trop forte). La suppression des petites chaînes est alors un bon moyen de se débarrasser de ces contours. Ce procédé reste cependant *ad hoc*.

2.4.5 Conclusion

Nous concluons que la modélisation des contours telle qu'elle est proposée par Canny ou par Perona et Malik est insuffisante pour traiter certains types d'images. Leur modèle est sans doute insuffisamment objectif, ce qui explique les étapes de corrections qui sont souvent proposées. Comme Pavlidis l'a souligné, nous avons des techniques très poussées pour détecter les discontinuités de type marche noyées dans un bruit blanc Gaussien, mais ce ne sont pas toujours des outils adaptés pour envisager la localisation des contours des objets ([Pavl 92]).

En fait, il n'est peut-être pas possible de trouver une modélisation satisfaisante des contours. Dans ce cas, la conception fondée sur une stratégie heuristique, dont nous avons expliqué les grandes lignes au premier chapitre, est une solution de remplacement d'autant plus intéressante. C'est celle que nous avons choisie pour développer notre détecteur de contours que nous présentons dans la partie suivante.

3. Choix algorithmiques pour notre détecteur

Nous présentons un nouveau détecteur de contours de type "marche" et de type "trait" fondé sur des heuristiques. Le but est de détecter tous les indices visuels primaires correspondant à des discontinuités pertinentes, de façon automatique, c'est à dire sans l'aide de l'utilisateur pour fixer d'éventuels paramètres ou seuils. Nous commençons par expliquer la méthodologie de conception, puis nous décrivons notre algorithme en détail, étape par étape [Salo 93]. L'ajustement des heuristiques, que nous considérons indépendamment des choix algorithmiques est présenté dans la partie suivante.

3.1 Protocole de construction de l'algorithme

De très nombreux détecteurs de contours ont déjà été présentés dans la littérature, mais à notre connaissance, aucun n'est issu d'une analyse de la gestion des informations telle que nous la proposons. Nous rappelons les principes que nous avons dégagés au premier chapitre et dont il nous faut tenir compte pour construire l'algorithme.

(1) Le nombre d'informations théoriquement nécessaires pour prendre la meilleure décision est un problème majeur qui mérite une grande attention lors de la conception d'un algorithme.

Pour suivre ce principe, nous proposons de placer au premier niveau de la conception la recherche des informations nécessaires à la validation d'un contour.

(2) Il faut guider les méthodes pour qu'une coopération soit fructueuse et fournisse des informations complémentaires.

Pour appliquer ce principe, nous proposons d'étudier les relations existant entre les informations et d'en déduire l'enchaînement des méthodes et la structure de contrôle du détecteur.

(3) Un modèle de description des indices visuels est suffisamment "objectif", si il permet une exploitation efficace et une émergence d'indices visuels de plus haut niveau.

Il faut pour cela choisir judicieusement nos informations et les structures de données qui les représentent. Le problème se situe au niveau des choix algorithmiques qui traduisent l'expertise humaine. Par exemple, quel gradient utiliser pour représenter la force d'une transition ?

(4) Pour une gestion efficace des informations, il faut préserver la liberté de mouvement de celles-ci.

Nous allons tenir compte de ce principe en attachant à un objet "contour" toutes les informations susceptibles d'être intéressantes, et en plaçant les informations de base en variables globales, donc visibles à tout moment.

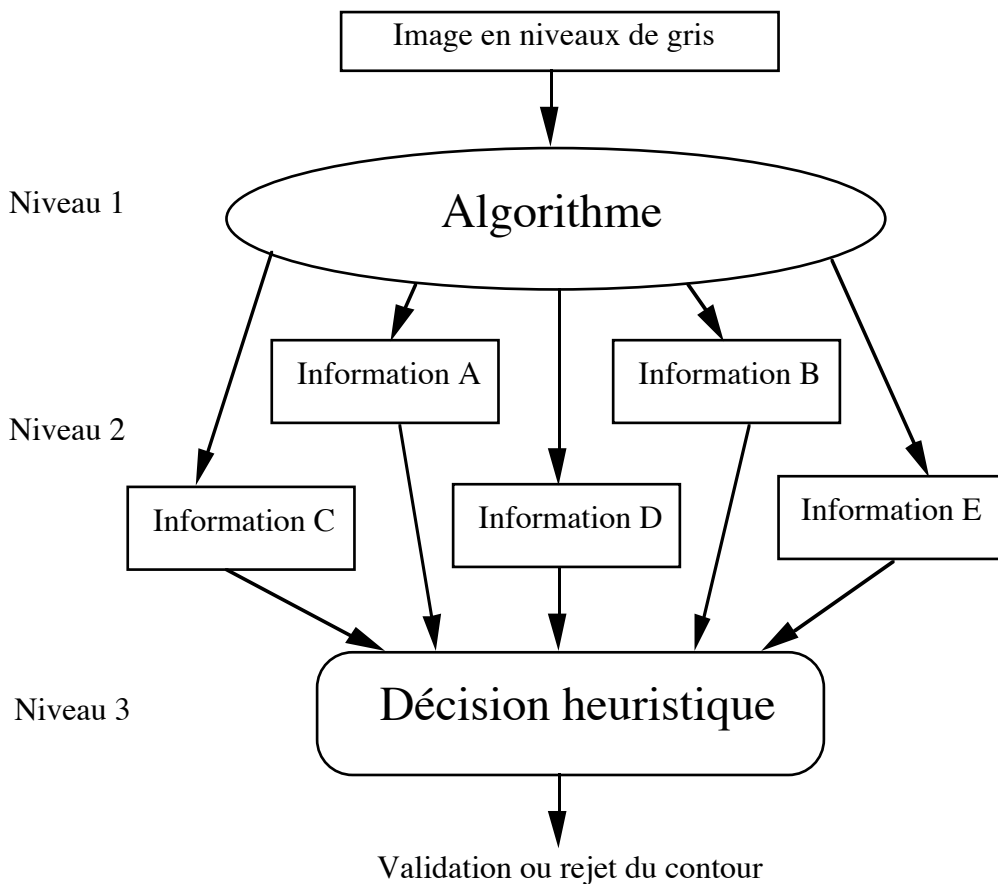


Figure 7 : Les différents niveaux de travail de notre protocole de construction

Nous adoptons finalement la démarche suivante pour construire notre détecteur :

- Nous nous attachons d'abord à déterminer quelles sont les informations nécessaires pour décider de la présence d'un contour. Par rapport au schéma de la figure 7, cela correspond au niveau 2. Ce travail nécessite une longue recherche expérimentale des informations qui sont pertinentes et si possible discriminantes pour déterminer la présence des contours, selon des critères de visibilité. Il y a également à ce niveau une recherche de la meilleure structure de données pour exprimer une information qui peut s'avérer très complexe et difficile à manipuler.

Remarquons que le choix des informations manipulées est indépendant de tout langage de programmation. Commencer par ce choix facilite ainsi la transcription algorithmique de notre expertise, ce qui est un problème particulièrement délicat.

- Il faut ensuite trouver le moyen par lequel les informations vont pouvoir être calculées. Il s'agit là d'un problème purement algorithmique, situé au niveau 1 de notre schéma. Il est important de noter que la complexité de l'information à calculer peut conduire à une

trop grande difficulté algorithmique, un temps de calcul prohibitif, ou une place mémoire trop importante. Il s'agit donc d'une part, de sélectionner des outils efficaces permettant d'obtenir ce que l'on cherche avec un bon rendement, et d'autre part d'obtenir une information suffisamment objective.

En outre, les relations existant entre les informations conduisent à des contraintes sur l'enchaînement des méthodes et l'organisation de l'algorithme dont il faut tenir compte.

- Nous étudions ensuite comment combiner les informations effectivement disponibles de manière à obtenir la meilleure détection possible. Nous observons expérimentalement la force de discrimination de certaines informations et nous en déduisons une formule générale pour nos heuristiques. En étudiant la distribution des points expérimentaux, nous ajustons alors nos fonctions de décision. Ces fonctions sont ce que nous appelons les décisions heuristiques du niveau 3 de la figure 7.

L'approche heuristique peut être considérée comme une démarche empirique. Toutefois, en ce qui concerne la détection des contours, comme nous l'avons déjà souligné, nous pensons que le modèle idéal est trop complexe pour pouvoir être défini *a priori*. Ainsi, l'approche que nous proposons est une alternative qui permet de trouver un modèle implicite sophistiqué, grâce à une démarche fondée sur une gestion intelligente des informations et un ajustement guidé par les résultats expérimentaux.

3.2 Spécifications générales

Nous spécifions ici les informations que nous voulons obtenir avant de décider de la présence d'un contour. Nous en déduisons ensuite le schéma général de notre détecteur. Nous proposons la recherche des informations suivantes :

1. Il nous faut des informations sur la force et la forme de la transition au niveau d'un pixel. Nous proposons pour cela de calculer un gradient pour tous les pixels, avec des indications sur sa norme, sa direction et son type "marche" ou "trait".

2. Nous illustrons figure 8 et 9 l'insuffisance de l'information sur le gradient pour un signal monodimensionnel. Pour s'assurer de la pertinence d'une transition, nous proposons de rechercher des informations concernant les régions du voisinage, comme indiqué figure 9. Ces informations, telles que moyenne, écart-type et gradient doivent être locales.

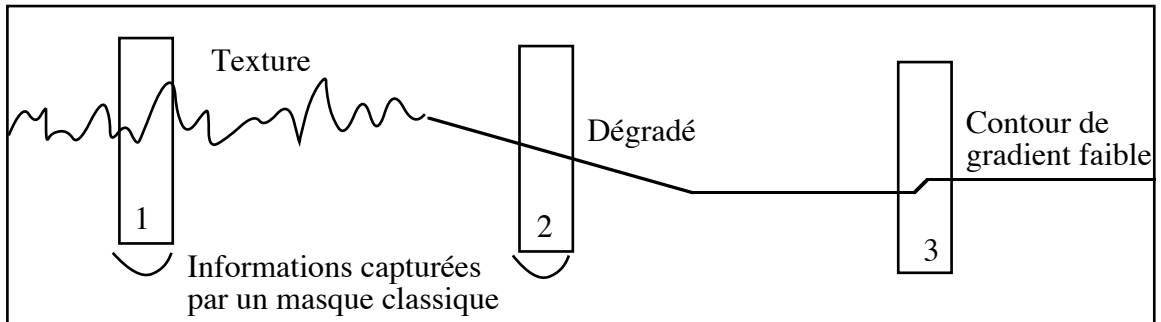


Figure 8 : La valeur des gradients locaux ne suffit pas à déterminer la présence d'un contour.

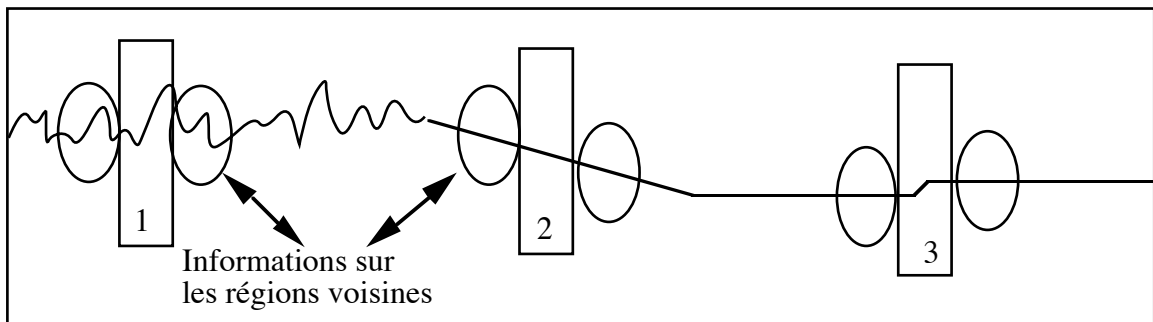


Figure 9 : Des informations sur les régions voisines permettent de lever l'ambiguïté.

3. Enfin, il nous semble important de tenir compte de la présence et de la localisation des pixels-contours voisins pour valider un autre pixel-contour. Par conséquent, nous proposons de rechercher des informations sur les pixels-contours voisins qui ont déjà été identifiés.

Les informations 1 et 2 peuvent être obtenues indépendamment des autres. L'information 3 pose en revanche un problème de fond. Connaître les pixels-contours du voisinage, avant de décider de la présence et de la localisation d'un autre pixel-contour, n'est que partiellement possible puisqu'il faut bien commencer par valider des pixels-contours isolés. Nous sommes donc amenés à choisir une structure algorithmique qui permette de tenir compte des résultats intermédiaires obtenus à un même niveau de décision. Nous aurions pu essayer une technique de relaxation, mais nous proposons une technique de croissance de contour incrémentale, inspirée des travaux de Martelli [Mart 76]. Etant donné un germe, nous essayons donc de chaîner petit à petit d'autres pixels-contours en étudiant chaque extrémité au fur et à mesure que le contour croît.

En tenant compte des informations à rechercher, nous proposons finalement le schéma algorithmique de la figure 10.

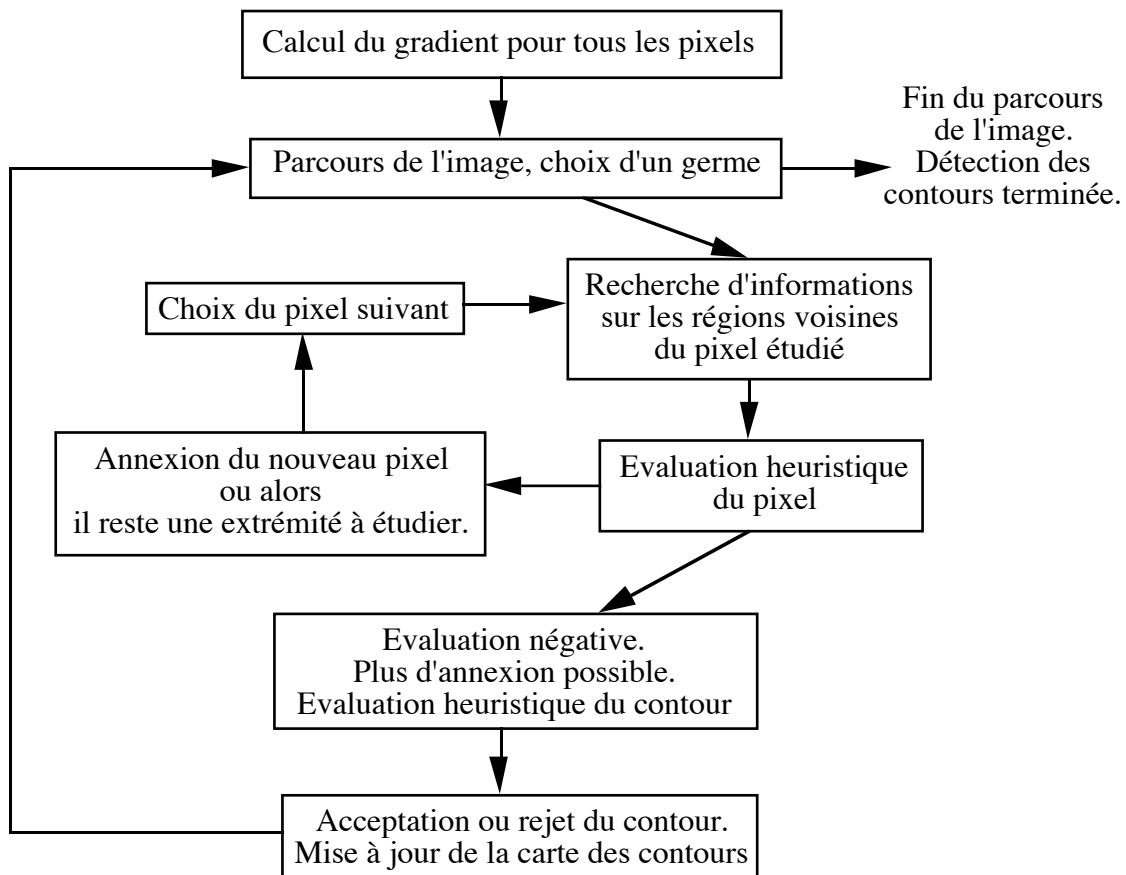


Figure 10 : Schéma général du détecteur de contours.

Nous détaillons par la suite chacune des étapes de la détection, en justifiant nos choix algorithmiques.

3.3 Calcul du gradient

3.3.1 Les différents choix possibles

Comme nous l'avons vu dans la partie 2 de ce chapitre, de nombreuses techniques existent pour calculer le gradient. Nous avons essayé deux méthodes. La première consiste à appliquer deux masques pour obtenir la valeur du gradient dans deux directions orthogonales et la deuxième s'inspire des masques de Kirsch. Il se pose cependant un problème spécifique à notre détecteur puisque nous désirons avoir un type de gradient "trait" ou "marche", selon la disposition des niveaux de gris. La caractérisation de ce type nous semble importante car en présence d'un trait, le gradient de type marche est pratiquement nul au centre, alors qu'en présence d'un contour de type marche, c'est le gradient de type trait qui est très faible. Ces deux discontinuités sont en fait

complémentaires, c'est pourquoi il nous apparaît judicieux de caractériser l'une et l'autre par des types différents.

Nous avons étudié finalement trois calculs possibles du gradient :

- CHOIX 1 : masques 3x3 de Sobel, pas de détection des traits.
- CHOIX 2 : masques 5x5 des figures 11 et 12 ("masques pleins"). Le gradient, l'orientation et le type "marche" ou "trait" sont déterminés par le masque qui a obtenu la plus forte réponse.

1 1 1 1 1	1 1 0 -1 -1	1 1 1 1 0	0 1 1 1 1
1 1 1 1 1	1 1 0 -1 -1	1 1 1 0 -1	-1 0 1 1 1
0 0 0 0 0	1 1 0 -1 -1	1 1 0 -1 -1	-1 -1 0 1 1
-1 -1 -1 -1 -1	1 1 0 -1 -1	1 0 -1 -1 -1	-1 -1 -1 0 1
-1 -1 -1 -1 -1	1 1 0 -1 -1	0 -1 -1 -1 -1	-1 -1 -1 -1 0

Figure 11 : Masques pour les "gradients escaliers".

1 1 1 1 1	1 1 -4 1 1	1 1 1 1 -4	-4 1 1 1 1
1 1 1 1 1	1 1 -4 1 1	1 1 1 -4 1	1 -4 1 1 1
-4 -4 -4 -4 -4	1 1 -4 1 1	1 1 -4 1 1	1 1 -4 1 1
1 1 1 1 1	1 1 -4 1 1	1 -4 1 1 1	1 1 1 -4 1
1 1 1 1 1	1 1 -4 1 1	-4 1 1 1 1	1 1 1 1 -4

Figure 12 : Masques pour les "gradients traits".

- CHOIX 3 : calcul du gradient avec les masques des figures 13 et 14. Le gradient, l'orientation et le type "marche" ou "trait" sont déterminés par le masque qui a obtenu la plus forte réponse.

0 1 1 1 0	0 0 0 0 0	0 1 1 0 0	0 0 1 1 0
0 1 1 1 0	1 1 0 -1 -1	1 1 0 0 0	0 0 0 1 1
0 0 0 0 0	1 1 0 -1 -1	1 0 0 0 -1	-1 0 0 0 1
0 -1 -1 -1 0	1 1 0 -1 -1	0 0 0 -1 -1	-1 -1 0 0 0
0 -1 -1 -1 0	0 0 0 0 0	0 0 -1 -1 0	0 -1 -1 0 0

Figure 13 : Masques pour les gradients de type marche.

0 1 1 1 0	0 0 0 0 0	0 1 0 0 0	0 0 0 1 0
0 0 0 0 0	1 0 -2 0 1	1 1 0 -2 0	0 -2 0 1 1
0 -2 -2 -2 0	1 0 -2 0 1	0 0 -2 0 0	0 0 -2 0 0
0 0 0 0 0	1 0 -2 0 1	0 -2 0 1 1	1 1 0 -2 0
0 1 1 1 0	0 0 0 0 0	0 0 0 1 0	0 1 0 0 0

Figure 14 : Masques pour les gradients de type trait.

Figure 15 : Evaluation des choix algorithmiques

Evaluation des différents choix pour calculer le gradient.

3.3.2 Choix retenu

Nous présentons figure 15 un résultat global sur l'image du muscle avec la norme et le typage de chaque gradient (la plus forte réponse détermine le type, si elle est supérieure à 3). Le problème est ici d'avoir une valeur, une direction et un type de gradient correspondant le mieux possible avec la véritable pente, selon nos critères visuels et notre appréciation des valeurs numériques.

EVALUATION :

- Pour le choix numéro 1, le gradient est calculé sur un voisinage trop étroit. En particulier pour les discontinuités étendues, la valeur du gradient est inférieure à la différence de moyenne entre les deux régions locales situées de part et d'autre de la discontinuité, ce qui pénalise l'objectivité de la valeur du gradient. En zone texturée ou complexe, la valeur du gradient est également trop irrégulière car trop sensible aux variations très locales. En revanche, les résultats sont meilleurs dans les dégradés et la localisation près des coins est facilitée, en raison de la diminution des effets de bords.

- Pour le choix numéro 2, le gradient est au contraire sensible "aux effets de bords", c'est à dire que le masque est parfois trop large. En effet, une discontinuité présente dans le voisinage immédiat peut entraîner une valeur de gradient non négligeable alors que le gradient calculé en 3x3 est nul. Le type attribué à un tel gradient est alors arbitraire. Ces approximations peuvent entraîner la détection de quelques petites fausses discontinuités. En revanche, la prise en compte de nombreux pixels pour calculer le gradient améliore la robustesse du typage et l'orientation de celui-ci. Notons également que les masques sont trop caractéristiques de la direction du gradient qu'ils privilégient. Si l'orientation de la discontinuité ne correspond pas tout à fait à la direction d'un masque, le gradient est plus faible que la valeur réelle en raison de l'influence des coefficients centraux du masque.

- Le choix numéro 3 est un compromis entre les deux premiers. De meilleurs résultats sont obtenus mais des imperfections subsistent. Il y a un léger décalage près de certains coins (1 à 2 pixels par rapport au choix 1) qui s'explique par le fait que le calcul du gradient sur les pixels situés exactement à la jonction donnent des valeurs inférieures à la pente réelle. Le type du gradient est aussi imprécis près des jonctions comportant un trait car certains effets de bords sont encore présents.

BILAN :

Quel que soit le choix retenu, il est important de noter qu'aucune des méthodes proposée n'est adaptée à un changement d'échelle. En effet, pour une résolution très fine

de l'image, une transition peut s'effectuer sur une dizaine de pixels et nos masques ne peuvent alors capturer qu'une partie de celle-ci. Pour simplifier le problème, **nous supposons donc que toutes les images présentées à notre détecteur ont le même facteur d'échelle**. La largeur d'une transition ne doit pas excéder la taille de nos masques. Pour des transitions très étroites, de l'ordre de un pixel, présentes essentiellement dans les images de synthèse simples, nos masques capturent toute la pente, mais il existe un biais systématique sur le calcul du gradient avec des répercussions au niveau de la localisation qui peut devenir ambiguë.

Pour que la valeur du gradient soit objective pour les deux types de gradient, avec des effets de bords limités, nous préférons le choix numéro 3. Toutefois, une des conclusions majeures de cette étude est la faible différence qui existe entre les résultats. Pour progresser, il faudrait en fait tenir compte des configurations particulières. Nous pourrions par exemple utiliser des masques spécifiques adaptés aux jonctions. Il manque sans doute également des masques intermédiaires, tant au niveau de l'orientation qu'au niveau du type du gradient.

Nous choisissons donc la troisième solution. Nous stockons les informations de gradient, norme, direction et type dans deux tableaux d'octets de la taille de l'image. Le premier tableau correspond à une carte de la norme des gradients et le deuxième permet de stocker le type ainsi que l'orientation de celui-ci sous forme codée. L'orientation ne peut prendre ici que huit valeurs possibles, selon le code de Freeman. Toutes ces informations sont stockées en variables globales pour être disponibles à tous les niveaux de l'algorithme, en accord avec notre principe de liberté de l'information.

3.4 Choix des germes

3.4.1 Les différents choix possibles

Pour sélectionner les germes, nous procédons de la façon suivante. Après le calcul du gradient, l'image est parcourue pixel par pixel. Dès qu'un pixel satisfait les conditions d'un germe, une croissance de contour est appliquée. Un germe de contour ne garantit aucunement la présence d'un contour, il sert uniquement de point de départ, la validation du contour intervenant bien plus tard. Lorsque le contour a été validé ou rejeté, le parcours de l'image reprend là où il s'était arrêté. Pour éviter de détecter plusieurs fois le même contour, nous marquons chaque pixel-contour dans la carte des contours résultats.

Un pixel est considéré comme germe s'il respecte les conditions suivantes :

- 1) Il ne doit pas être marqué, c'est à dire appartenir déjà à un contour.
- 2) Il doit avoir un gradient supérieur ou égal à un seuil de visibilité. Après étude expérimentale de la visibilité de plusieurs contours de faible gradient, nous proposons une valeur de 4 pour ce seuil.
- 3) La valeur du gradient doit être localement maximum dans la direction de celui-ci.

Si les conditions 1 et 2 ne posent pas de problème algorithmique, la condition 3 peut être exprimée de nombreuses façons différentes. Pour cette étape de sélection des germes, nous proposons donc d'étudier deux aspects :

- L'influence du sens de parcours de l'image.
- L'expression algorithmique de la condition 3.

Nous prenons la notation suivante :

$P(x)$ est le pixel étudié, $P(x+i)$ et $P(x-i)$ sont les pixels voisins de P situés dans une direction donnée à une distance de i pixels, D_{gradient} est la direction du gradient, $G(x)$ est la valeur du gradient à la position x .

- CHOIX 1 : Le parcours de l'image se fait de haut en bas et de gauche à droite. La condition 3 de validation d'un germe est exprimée par :

dans la direction D_{gradient} , $(G(x) \geq G(x+1))$ et $(G(x) \geq G(x-1))$

- CHOIX 2 : Le parcours de l'image se fait de bas en haut et de droite à gauche. La condition 3 reste identique.

- CHOIX 3 : Comme le choix 1 pour le sens de parcours. L'expression de 3 est :

dans la direction D_{gradient} , $(G(x) \geq G(x+1))$ et $(G(x) \geq G(x-1))$

et dans la direction $D_{\text{gradient}} + 45$ degrés, $(G(x) \geq G(x+1))$ et $(G(x) \geq G(x-1))$

et dans la direction $D_{\text{gradient}} - 45$ degrés, $(G(x) \geq G(x+1))$ et $(G(x) \geq G(x-1))$

- CHOIX 4 : Comme le choix 1 pour le sens du parcours, mais la valeur du gradient doit être maximale dans la direction de celui-ci sur 5 pixels alignés au lieu de 3, ce que nous écrivons :

dans la direction D_{gradient} , $(G(x) \geq G(x+2))$ et $(G(x) \geq G(x+1))$

et $(G(x) \geq G(x-1))$ et $(G(x) \geq G(x-2))$

Figure 16 : Evaluation des choix algorithmiques.

Evaluation des différents choix pour obtenir des germes de contours.

- CHOIX 5 : Comme le choix 1 pour le sens du parcours. Pour l'expression de la condition 3 de validation d'un germe, nous proposons une sophistication qui prend en compte la spécificité de la direction diagonale, dûe à la discrétisation :

Si Dgradient est horizontale ou verticale,

alors { le gradient est examiné sur les 3 pixels alignés de la direction Dgradient }
dans la direction Dgradient, $(G(x) \geq G(x+1))$ et $(G(x) \geq G(x-1))$

sinon { le gradient est donc diagonal }

{ le gradient est examiné sur les 3 pixels alignés de cette direction diagonale,
sur 3 pixels alignés horizontalement et sur 3 pixels alignés verticalement }
dans la direction Dgradient, $(G(x) \geq G(x+1))$ et $(G(x) \geq G(x-1))$

et (dans la direction Dgradient + 45 degrés, $(G(x) \geq G(x+1))$ et $(G(x) \geq G(x-1))$
ou dans la direction Dgradient - 45 degrés, $(G(x) \geq G(x+1))$ et $(G(x) \geq G(x-1))$
)

3.4.2 Choix retenu

Nous présentons figure 16 une évaluation des résultat de notre détecteur avec l'image des cellules qui présente des zones texturées difficiles avec de nombreux germes de contours. Pour chaque choix, nous proposons la carte des germes, sauf pour le deuxième car seule la comparaison des résultats finaux est intéressante.

EVALUATION :

- Notre détection des contours est pratiquement indépendante du sens de parcours de l'image. Sur plus de la moitié des images étudiées, aucune différence n'a été relevée. Il est en fait difficile de déterminer tous les cas qui conduisent à une détection différente. Le seul cas qui a été identifié correspond à un petit contour oublié lors du parcours haut-bas et qui a pu être détecté avec un parcours bas-haut. Etant détecté en premier, il a en effet été prolongé sur une partie plus contrastée (voir figure 17). Les différences dues au sens de parcours sont toutefois négligeables en comparaison des différences qui existent entre deux détecteurs quelconques.

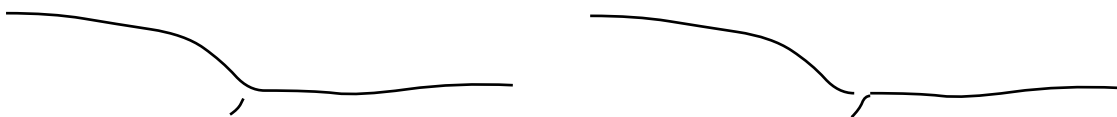


Figure 17 : A gauche, parcours de haut en bas, le petit contour ne sera pas retenu; à droite, parcours de bas en haut, il n'y a plus de petit contour, donc la détection est complète.

- Le choix 1 est trop tolérant. La carte des germes révèle des lignes épaisses de germes. En effet, il existe des configurations particulières **diagonales** où le gradient est maximal localement sans appartenir à un réel contour. Un exemple d'une telle configuration est montré figure 18. Les faux germes sont en caractères gras, la discontinuité est en caractères soulignés. Comme nous n'interdisons pas un départ de contour juste à côté d'un autre contour, il arrive dans des cas particuliers qu'un faux contour soit créé à partir de ce pixel et qu'il prenne appui sur les "contrebas" d'une discontinuité large. Ces faux-contours sont extrêmement gênants et le choix 1 ne peut donc être retenu.

2	4	19	70	<u>90</u>
1	5	64	<u>93</u>	68
3	37	<u>92</u>	<u>60</u>	28
42	<u>85</u>	71	42	9
<u>80</u>	64	34	9	4

Figure 18 : Valeurs de gradients pour un contour de direction diagonale. En souligné, les gradients des pixel-contours, en gras, les gradients de pixels quelconques qui sont tout de même maximum locaux dans la direction diagonale.

- En revanche, pour le choix 3, la carte des germes est incomplète. En effet, si la courbure locale de la discontinuité est importante et que le pixel-contour voisin a une forte valeur de gradient, un "bon" germe peut être éliminé. Toutefois, cela n'empêche généralement pas la détection de la discontinuité car un des pixels voisins a presque toujours les qualités requises pour être un bon germe, sauf cas exceptionnel (1 petit contour important a été oublié pour cette raison dans l'image du muscle).

- Le choix 4 permet une plus grande robustesse dans la texture, mais certains germes sont oubliés dans le cas de contours proches ou près des jonctions. Le problème d'épaisseur des lignes de germes n'est également pas résolu pour les contours d'orientation proche d'une diagonale. Cette version n'est donc pas non plus satisfaisante.

- Le cinquième choix est en fait notre proposition pour résoudre à peu près correctement les problèmes soulevés précédemment, comme le montre la carte des germes qui y correspond. C'est un compromis entre le 1 et le 3 qui prend en compte la particularité de la direction diagonale (les autres directions ne posent pas de problème d'épaisseur des lignes de germes) et le changement de courbure (passage d'un gradient horizontal ou vertical à un gradient diagonal). C'est bien sûr ce choix qui a été implanté dans notre algorithme.

3.5 Choix du pixel suivant

3.5.1 Les différents choix possibles

La croissance de contour consiste à annexer un à un de nouveaux pixels au germe initial pour trouver la discontinuité complète. Nous nous intéressons ici au choix du nouveau pixel sans nous préoccuper pour l'instant de sa validation. Pour cela, il est nécessaire d'orienter la discontinuité, de choisir une extrémité et d'étudier les pixels qui peuvent la prolonger.

La chaîne de pixels formant le contour est orientée en fonction de l'orientation du gradient. Toutefois, en ce qui concerne les traits, le gradient possédant une direction mais pas d'orientation (voir figure 19), le début et la fin de la chaîne sont choisis de façon arbitraire. A chaque nouvelle annexion, les attributs d'orientation sur le dernier pixel-trait sont éventuellement harmonisés aux valeurs précédentes, en vérifiant que l'orientation n'a pas changé de plus de 90 degrés.



Figure 19 : Problèmes d'orientation du contour.

- A gauche, orientation d'un contour de type marche de la région la plus foncée vers la région la plus claire.
- A droite, orientation arbitraire d'un contour de type trait.

Nous proposons de choisir le pixel suivant parmi trois pixels situés dans la direction perpendiculaire au gradient à 45 degrés près, comme indiqué figure 20. Parmi ces 3 pixels, nous imposons les contraintes suivantes :

- 1) Afin de préserver la cohérence des propriétés locales du contour, le pixel choisi doit avoir le même type de gradient ("marche" ou "trait") que les autres pixels du contour.
- 2) Si le pixel choisi est déjà marqué dans la carte des contours, alors l'extrémité du contour est atteinte, il faut étudier l'autre extrémité ou, le cas échéant, passer à l'étape de validation du contour.



Figure 20 : Trois pixels sont examinés, en fonction de la direction du gradient sur le dernier pixel.

Nous avons différentes stratégies possibles pour sélectionner le pixel parmi les trois candidats, nous en exposons deux :

- Choix 1 : Le pixel choisi est celui qui a la plus forte valeur de gradient parmi les trois candidats.
- Choix 2 : Nous essayons de tenir compte du cas particulier où le contour se rapproche d'un autre contour avec fortes valeurs de gradient (voir figure 21).

En effet, dans ce cas précis, le gradient n'est plus un maximum local dans le prolongement du contour. Le risque est alors de joindre le contour voisin et de manquer la véritable suite du contour.

Nous proposons de traiter ce cas avec l'heuristique suivante.

Soit p le pixel situé à l'extrémité du contour, soit p_1 , p_2 et p_3 les 3 candidats, tels que p_2 soit situé exactement dans la direction perpendiculaire au gradient en p . Soit $G(x)$ et $D(x)$ respectivement la norme et la direction du gradient sur un pixel donné x . Soit p_{max} le pixel qui a la plus forte valeur de gradient parmi p_1 , p_2 et p_3 . Nous envisageons les cas suivants :

- si $G(p_{max})$ est un maximum local du gradient (condition 3 de la section précédente), alors le pixel choisi est p_{max} .
- sinon si $(p$ était le candidat précédent qui avait la plus forte valeur de gradient) et $(p$ n'était pas un maximum local du gradient), alors le pixel choisi est p_2 , sinon le pixel choisi est p_{max} .

Ligne	Choix du pixel : plus fort gradient	Choix théorique optimal du pixel
1		
2		
3		
4		
5		

Figure 21 : Illustration du problème du choix du pixel lorsque le contour (en noir) se rapproche d'un autre contour (en "briques"). Chaque colonne représente symboliquement le niveau de gris d'un pixel.

3.5.2 Choix retenu

- Pour évaluer les deux choix, l'image la plus intéressante est celle de l'objet polyédrique avec de très nombreux coins et donc de nombreuses configurations problématiques. Nous présentons les résultats obtenus avec ces deux choix figure 22.

EVALUATION :

- Avec le choix 1, certaines jonctions sont très mal localisées : trois à quatre pixel-contours manquent et les raccords sont trop rapides.

Figure 22 : Evaluation des choix algorithmiques

Evaluation des choix algorithmiques pour la sélection du pixel suivant dans le processus de croissance de contour.

- Avec le choix 2, des erreurs subsistent mais les résultats sont tout de même meilleurs. C'est ce choix que nous avons retenu et implanté dans notre algorithme. L'heuristique proposée permet en fait de retarder la jonction entre les deux contours, en ne prenant qu'une fois sur deux le pixel qui a la plus forte valeur de gradient, dans un contexte où aucun des candidats n'est un maximum local du gradient. Cette heuristique n'est pas optimale, car il existe de nombreuses configurations qui mériteraient un traitement à part. Il doit être possible de les identifier et de les traiter correctement, mais cela reste toutefois difficile. Notons que les contours semblent parfois "hésitants" en raison de l'orientation du gradient qui n'est pas toujours bien calculée près des jonctions. Sur les autres images, les différences sont insignifiantes et ne méritent pas d'être signalées. Ce problème est donc spécifique aux jonctions entre plusieurs arêtes.

Notons un point intéressant de la poursuite du contour : comme nous choisissons obligatoirement un pixel dont le type de gradient est le même que celui de la discontinuité, nous pourrions nous attendre à des oublis dus à des changements de type. En fait, cela ne pose pas vraiment de problème car la discontinuité est alors détectée en deux parties. Comme il est facile de le constater pour les résultats concernant les contours complexes, il arrive fréquemment qu'un trait soit détecté isolément et qu'un peu plus loin ce soit le contour escalier en bordure de trait qui ait été détecté.

3.6 Recherche d'informations sur les régions voisines

3.6.1 Les différents choix possibles

Comme nous l'avons déjà montré section 3.2, les informations concernant le gradient ne sont pas suffisantes. Nous proposons donc une étape de recherche d'informations sur les régions voisines du pixel candidat. Nous proposons de calculer les informations de moyenne et d'écart-type des niveaux de gris pour deux régions situées de part et d'autre du pixel-contour étudié. Il se pose à ce niveau plusieurs problèmes :

- Comment prendre en compte le facteur d'échelle ?
- Quel voisinage étudier ?
- Comment obtenir les informations voulues ?

En ce qui concerne le facteur d'échelle, comme pour le gradient, nous nous affranchissons du problème en considérant que celui-ci est constant pour toutes les images. Cela nous permet de rechercher des informations dans un voisinage limité,

sinon, il faudrait envisager une taille de voisinage qui soit fonction du facteur d'échelle, ce qui complique le problème.

Le voisinage intéressant se situe dans la direction du gradient, à quelques pixels de part et d'autre du pixel étudié pour prolonger le contour.

Plusieurs choix algorithmiques s'offrent à nous pour obtenir ces informations.

- Choix 1 : Nous pouvons effectuer une croissance locale de régions par agrégation de pixels. L'avantage de cette technique est de pouvoir opérer une sélection dynamique adaptative des pixels qui vont être pris en compte. Nous proposons de caractériser cette croissance de région par deux paramètres, n le nombre de pixels à agréger et $G(x,y)$ la position approximative du germe.

Pour s'adapter à la difficulté locale, nous pouvons faire varier n en fonction de la valeur du gradient sur le pixel-contour examiné. Plus le gradient est faible, plus l'étude du voisinage requiert de l'importance avec donc une valeur de n plus grande. Nous proposons de choisir n de la façon suivante :

- gradient ≥ 20 ---> $n = 5$
- 12 \leq gradient < 20 ---> $n = 6$
- 8 \leq gradient < 12 ---> $n = 8$
- gradient < 8 ---> $n = 10$

En ce qui concerne le germe de la croissance de région, nous le caractérisons de la façon suivante :

Soit $M(x_0,y_0)$ le pixel-contour étudié (marqué avec un point d'interrogation figure 23).

Soit $P(x,y)$ le pixel situé à une distance de deux pixels de M (en 8 voisinage) dans la direction du gradient (le pixel pointé par la flèche figure 23).

- pixel contour
- ? pixel contour examiné
- ↙ direction du gradient
- germe possible pour l'agrégation de points
- pixel d'une région après agrégation

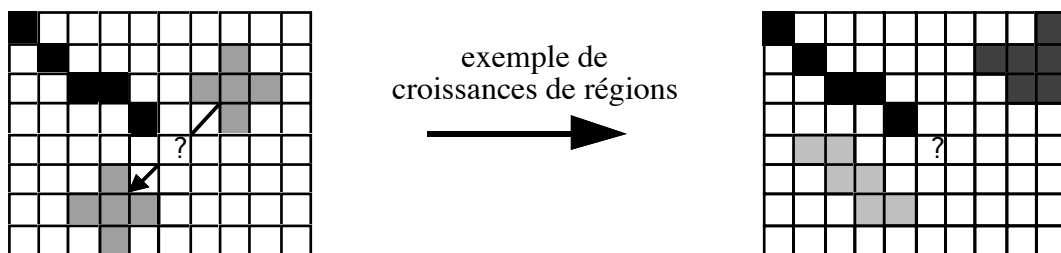


Figure 23 : Croissances locales de régions.

Nous définissons le critère de qualité Q d'un pixel p inversement proportionnellement à l'écart-type des niveaux de gris dans un fenêtre 3×3 centré sur p . Le germe $G(x,y)$ de la croissance de région est le pixel avec la plus forte valeur de Q parmi un ensemble de 5 pixels, définis par $P(x,y)$ et ses quatre voisins (voir figure 23). Ensuite, $n-1$ agrégations de pixels connexes sont effectuées, toujours en fonction de Q . La moyenne et l'écart-type des niveaux de gris de la région sont calculés dynamiquement pendant l'agrégation.

- Choix 2 : On calcule la moyenne et l'écart-type local dans deux fenêtres 3×3 centrées sur le pixel situé à une distance n du pixel contour dans la direction du gradient. Nous retenons ensuite la fenêtre qui a le plus petit écart-type local.

En exprimant la distance en nombre de pixels, nous proposons deux sous-choix, illustrés figure 24 :

Choix 2.1 : contours marches : $n = 2$
 contours traits : $n = 3$

Choix 2.2 : contours marches : $n = 3$
 contours traits : $n = 4$

La distance pour les traits est plus importante car ces configurations sont en général plus larges que les configurations de type marche. En effet, pour atteindre la bordure de la région à partir d'un pixel-contour de type trait, il faut franchir la transition de type marche qui lui est toujours associée.

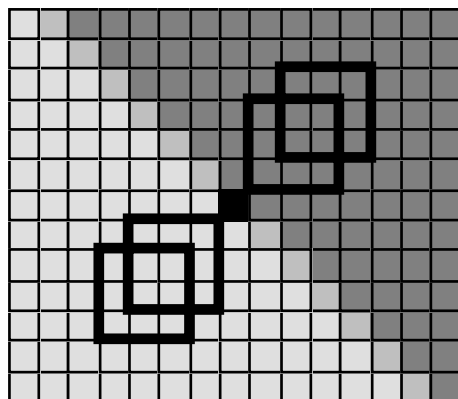


Figure 24 : Fenêtres 3×3 utilisées pour obtenir des informations sur les régions dans le cas des contours marches, choix 3.

- Choix 3 : Nous proposons une alternative au choix 2, qui consiste à retenir la meilleure des **quatre** fenêtres 3×3 . Ces fenêtres 3×3 sont donc situées aux distances n suivantes :

contours marches : $n = 2$ et $n = 3$

contour traits : $n = 3$ et $n = 4$

Nous conservons la fenêtre qui minimise l'écart-type local 3×3 .

3.6.2 Choix retenu

Nous présentons figure 25 une évaluation des 4 choix avec l'image de la femme. Les critères utilisés pour la validation des contours sont présentés dans la partie 4.

Figure 25 : Evaluation des choix algorithmiques

EVALUATION :

- Peu de différences sont constatées entre chaque choix en comparaison des différences notables qui existent par exemple avec un autre détecteur.

- Le choix 1 fournit des résultats intéressants, mais ils sont encore loin de la perfection. Dans la texture en particulier, les pixels sélectionnés ont un voisinage très homogène, ce qui entraîne une région avec un écart-type trop faible (objectivement) ce qui cause parfois la détection de faux contours. En fait, l'évaluation heuristique a été adaptée pour que cela n'arrive que rarement, un peu au détriment de la détection des contours dans les zones texturées. L'inconvénient majeur de cette croissance de régions est qu'elle ne fait pas émerger correctement les informations pertinentes qui caractérisent la zone voisine du contour. En l'occurrence, l'écart-type trouvé n'est pas toujours réaliste. Notons que la croissance de régions est responsable d'environ la moitié du temps de calcul de notre détecteur (3 minutes pour une image 256x256). Le gain en qualité de détection par rapport à un détecteur classique, ainsi que l'automatisme de notre détecteur, se paye à ce niveau, ce qui contraint les tentatives de sophistication de la croissance de régions, au moins en ce qui concerne le développement sur nos ordinateurs.

- Les problèmes des choix 2.1 et 2.2 se rencontrent dans les cas difficiles. Dans certains cas, une distance de 3 est trop faible et dans d'autres cas c'est la distance de 4 qui est trop forte. Ainsi des erreurs commises avec le choix 2.1 (des oublis essentiellement) ne sont pas faites avec le choix 2.2 et réciproquement. Ces erreurs interviennent lorsque la transition est complexe ou que plusieurs discontinuités sont proches l'une de l'autre : le carré 3x3 est situé sur une autre discontinuité et l'écart-type très fort entraîne un rejet du pixel-contour.

- C'est le troisième choix qui semble donner les meilleurs résultats. Le détecteur est un peu plus robuste à la texture tout en étant relativement sensible aux transitions en zone texturée. C'est l'analyse des défauts des choix 2.1 et 2.2 qui nous a suggéré d'essayer cette méthode. En effet, la sélection des fenêtres 3x3 permet une adaptation aux cas particuliers relativement satisfaisante. L'intérêt est de regarder à la bonne distance comment sont les régions voisines, en imposant une fenêtre d'observation compacte qui rend l'information de texture plus objective, au contraire de la croissance de régions (choix 1) qui tend à découvrir en pleine texture des zones homogènes fortement découpées. Notons finalement que le choix 3 permet un gain de temps d'au moins un facteur 2 par rapport au choix 1.

BILAN :

Le choix 1 a donné lieu à plusieurs publications [SaGa 93a], [SaGa 93b]. Le choix retenu, étudié ultérieurement [SaGa 94], est toutefois le numéro 3 car les résultats sont meilleurs.

L'implantation de cette méthode ne comporte pas de difficulté algorithmique majeure. Pour obtenir la position du centre de chaque fenêtre, il faut tenir compte de la position du pixel-contour étudié et de la direction du gradient. Pour simplifier les calculs, nous avons associé à chaque objet pixel, quatre attributs de type entier, qui stockent le décalage positionnel nécessaire pour retrouver le pixel voisin situé "devant", "derrière", "à_droite" ou "à_gauche" en fonction de la direction du gradient. Pour trouver un pixel situé devant à une distance de 3, il suffit donc d'ajouter 3 fois "devant" à la position du pixel-contour étudié.

En ce qui concerne l'écart-type et la moyenne pour une fenêtre 3x3 donnée, ils sont calculés en flottant pour garder une bonne précision.

3.7 Décisions heuristiques

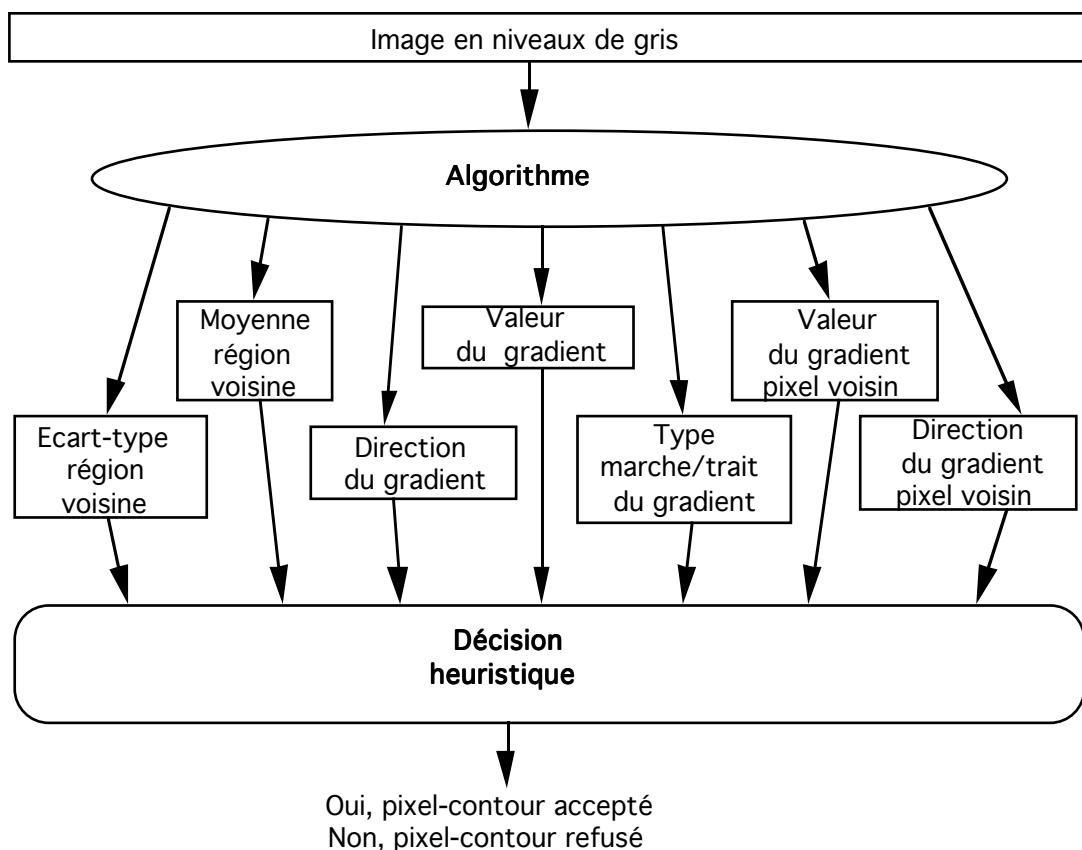


Figure 26 : Décision heuristique au niveau du pixel-contour.

Nous avons deux niveaux de décisions heuristiques. Le premier concerne l'évaluation d'un pixel-contour. Pour cela, nous disposons d'informations sur le gradient et d'informations sur les régions voisines. Comme indiqué figure 26, le but de l'heuristique est de valider ou de rejeter un pixel-contour candidat.

Le deuxième niveau de décision concerne la validation d'un contour en considérant le nombre de pixels, le gradient moyen et toutes les informations collectées lors de la croissance du contour, qui peuvent s'avérer intéressantes pour accepter ou refuser celui-ci, comme indiqué figure 27.

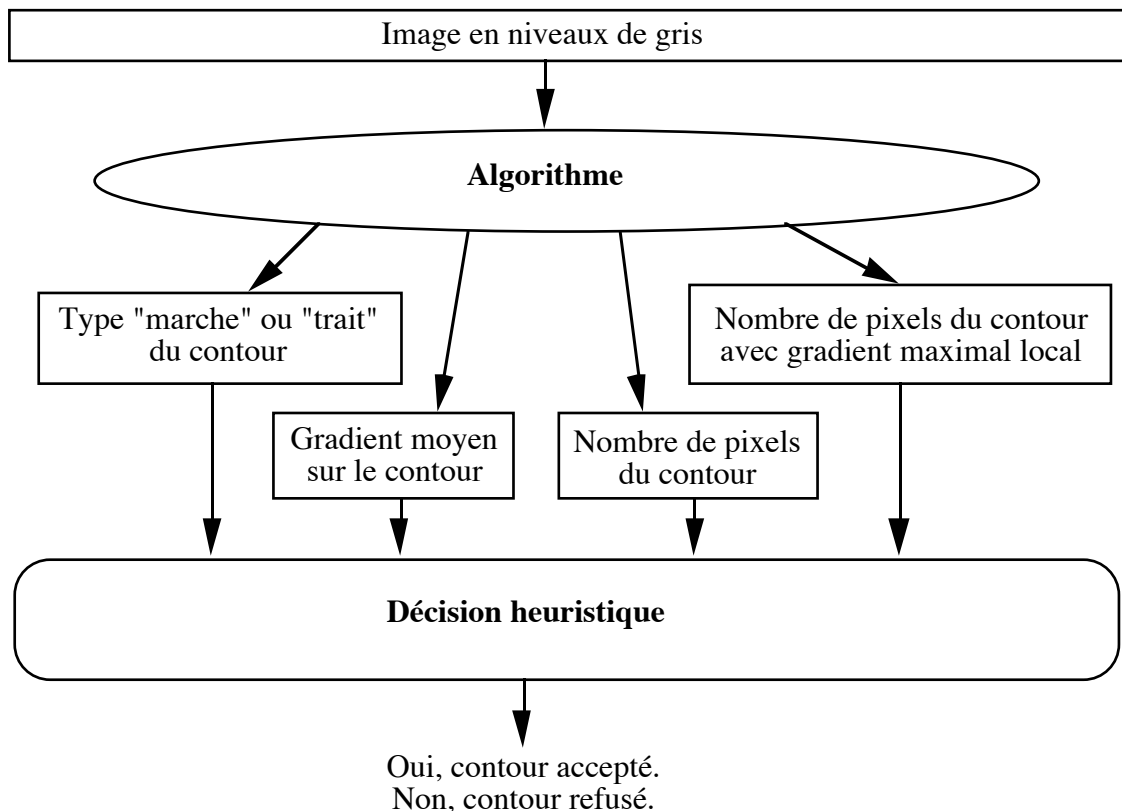


Figure 27 : Décision heuristique au niveau du contour.

Ces deux niveaux d'heuristiques ne comportent pas de difficulté de type algorithmique, mais plutôt de type conceptuel. Il s'agit en fait de trouver la fonction de décision qui permet d'obtenir les meilleurs résultats. Cette étape nécessitant une étude expérimentale poussée et une analyse conceptuelle spécifique, nous terminons ici l'étude des choix algorithmiques, et nous proposons de consacrer toute la partie suivante à l'étude de ces heuristiques.

4. Ajustement des heuristiques

Après avoir trouvé les moyens algorithmiques permettant d'obtenir des informations pertinentes pour la validation d'un contour, il nous faut maintenant combiner ces informations pour construire la meilleure fonction de décision possible. Nous allons pour cela décomposer notre expertise en identifiant ce qu'est un contour, puis nous proposons une démarche expérimentale pour traduire notre expertise en une fonction de décision heuristique.

4.1 Décomposition de notre expertise

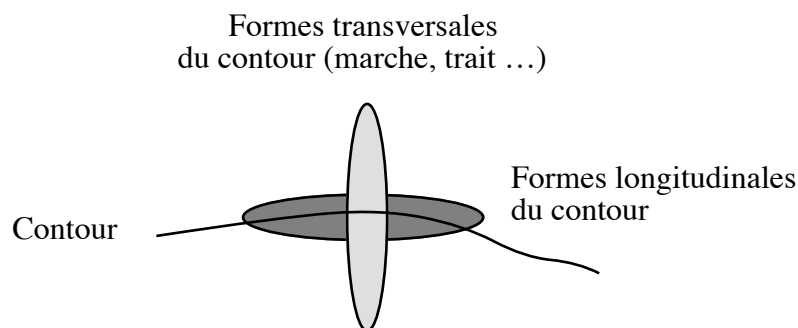


Figure 28 : Les différents aspects d'un contour.

Nous avons proposé un nouvel algorithme pour la détection des contours. Il nous reste toutefois à préciser les heuristiques exactes qui vont déterminer la décision finale d'acceptation ou de rejet d'un contour. La définition de ces heuristiques va en fait déterminer le modèle de contour qui est implicitement "caché" dans notre algorithme. Pour définir ces heuristiques, il faut expliciter notre expertise de la détection des contours. Nous distinguons deux grands aspects, présentés figure 28.

- Le premier est transversal et concerne la validation d'un pixel-contour. Nous disposons pour cela d'informations uniquement locales telles que la valeur du gradient sur le pixel étudié, sa direction, son type et les valeurs de moyenne et d'écart-type des quatre fenêtres 3x3 situées devant et derrière ce pixel. Il s'agit donc maintenant de déterminer comment combiner ces informations pour obtenir une décision heuristique de qualité. Puisqu'il s'agit d'informations latérales par rapport au contour, nous cherchons donc ici à caractériser notre expertise de la visibilité transversale du contour.

- Le deuxième aspect est longitudinal, il concerne la validation complète du contour. Il s'agit ici de prendre en compte tous les pixels du contour pour déterminer sa visibilité globale.

Pour ces deux aspects, nous essayons de définir un critère de visibilité. Ensuite, nous proposons de construire des heuristiques qui se rapprochent autant que possible de la meilleure détection relativement à ces critères.

4.2 Aspect transversal des contours

4.2.1 Critère de visibilité

Nous avons défini un indice visuel primaire comme un indice simple pouvant être directement déterminé par des calculs sur les niveaux de gris et la position des pixels. Le problème est ici de caractériser l'indice contour. Puisqu'il s'agit d'un indice visuel, celui-ci respecte des propriétés locales qui sont "visibles". Toute la difficulté est de savoir quelle est la limite de visibilité d'un contour et qu'est-ce qui la caractérise.

La visibilité d'un contour dépend avant tout du facteur d'échelle. Or, pour calculer le gradient et les informations sur les régions voisines, nous avons été amenés à prendre comme hypothèse de base que le facteur d'échelle était le même pour toutes les images à appliquer à notre détecteur. D'un point de vue visibilité humaine, cette condition est également nécessaire, car à un niveau de résolution donné, un contour est visible, alors qu'à un autre il ne l'est plus. Or, nos yeux ne peuvent changer automatiquement de résolution. Différentes largeurs de transition sont toutefois acceptables, mais dans un intervalle de valeurs limité. Nous gardons donc comme hypothèse que la largeur d'une transition visible doit être inférieure ou égale à la taille de nos masques, soit 5 pixels.

La visibilité d'un contour reste tout de même une propriété subjective. Si nous considérons des informations strictement transversales, il est parfois difficile de se prononcer sur la visibilité d'un pixel-contour. Pour diminuer le caractère subjectif de cette visibilité, il est nécessaire d'examiner de nombreuses configurations et d'essayer de généraliser l'expertise acquise. Le choix des configurations n'est pas limité. Pour que notre expertise soit générale et juste, il est préférable d'examiner un grand nombre d'images et toutes sortes de configurations possibles. De cette manière, nous arrivons finalement à déterminer dans la plupart des cas si un pixel-contour est visible, indépendamment de tout contexte et de toute application.

4.2.2 Recherche du critère de visibilité

Nous cherchons ici à caractériser le critère de visibilité d'un pixel-contour, en fonction des informations disponibles.

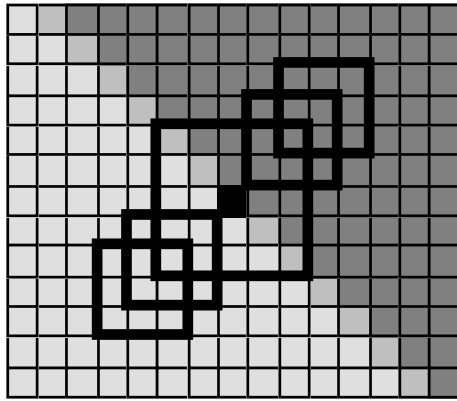
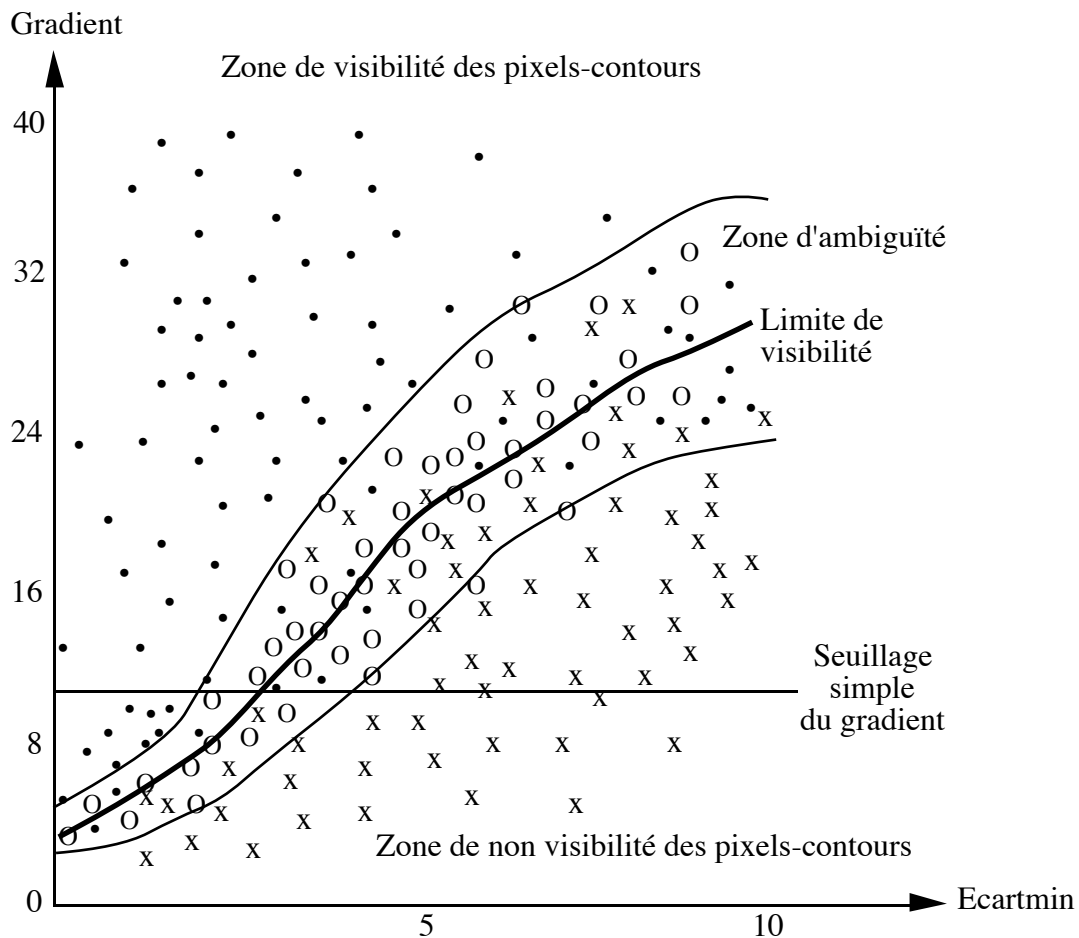


Figure 29 : Fenêtres 3x3 et 5x5 utilisées pour le calcul des écart-types et du gradient.

Nous présentons figure 29 les quatre fenêtres 3x3 utilisées pour obtenir des informations sur les régions voisines et la fenêtre 5x5 utilisée pour le calcul du gradient sur le pixel marqué en noir. Intuitivement, un pixel-contour est visible si les variations dans les régions voisines sont faibles par rapport à la valeur du gradient. Suivant cette idée, nous proposons d'étudier la visibilité expérimentale d'un grand nombre de pixels, en fonction d'une part, de la valeur du gradient notée Gradient, et d'autre part du plus faible des écart-types des quatre fenêtres 3x3, noté Ecartmin.

Nous résumons figure 30 le résultat de nos expériences, effectuées sur un grand nombre de configurations appartenant à une dizaine d'images de type différent. Notons que ce schéma n'est qu'une représentation symbolique du véritable diagramme, pour des raisons de clarté de présentation. Les pixels correspondant à des points expérimentaux sont tous de type marche, un schéma semblable existant pour le type trait.

Le diagramme peut être divisé en trois grandes régions. En haut et plutôt vers la gauche, la zone des pixels-contours évidents représente la majeure partie des pixels-contours existants dans les images (nous n'avons pas indiqué les points dont le gradient est supérieur à 40). Ainsi, pour détecter correctement de nombreux contours, un seuillage simple du gradient suffit. Toutefois, en présence de forte texture ou de bruit, ce simple seuillage entraînera également la détection de faux pixels-contours situés sous la ligne de "Seuillage simple" (voir en annexe 2 le résultat d'un seuillage par hystérésis). En bas, et plutôt vers la droite, nous avons les points correspondant à du bruit ou de la texture évidente. Entre les deux zones, nous avons la zone des points où la visibilité est incertaine. Plutôt que les classer approximativement en "visible" ou "non visible", nous avons préféré en faire une classe à part.



Légende x : point expérimental où aucun pixel-contour n'est visible
 • : point expérimental où un pixel-contour est visible
 o : point expérimental où il est difficile de se prononcer

Figure 30 : Diagramme expérimental de visibilité.

Une des caractéristiques essentielles de ce schéma est de montrer clairement la dépendance de la visibilité vis-à-vis du Gradient et de Ecartmin. Même si la zone d'ambiguïté n'est pas nulle, les deux paramètres Gradient et Ecartmin sont relativement discriminants pour le critère de visibilité. Notons également que de nombreux points expérimentaux situés dans la zone d'ambiguïté correspondent à des régions en dégradé régulier, où donc aucune discontinuité ne doit être détectée. En ajoutant l'information de dégradé comme troisième information discriminante, la zone de recouvrement serait ainsi plus fine.

En ce qui concerne les contours de type trait, un schéma similaire a été trouvé, toujours avec en fonction du Gradient et de Ecartmin. Ces deux variables sont là aussi des paramètres relativement discriminants pour exprimer la visibilité. Il n'y a en revanche aucun problème lié au dégradé.

Nous proposons finalement de construire des heuristiques fondées sur les conclusions que nous avons tirées de ce schéma.

4.2.3 Construction de l'heuristique

Nous avons caractérisé approximativement le critère de visibilité d'un pixel-contour. Il nous faut maintenant obtenir une formule heuristique correspondant à ce critère. Nous proposons deux aspects à prendre en compte dans cette heuristique, la qualité de la visibilité, exprimée en fonction de Gradient et Ecartmin et l'identification des dégradés. La formule générale de cette heuristique est la suivante :

```
Si Type_gradient = Marche
alors si Identification d'un dégradé
    alors pixel-contour refusé
    sinon si visibilité (Gradient, Ecartmin) > 0
        alors pixel-contour accepté
        sinon pixel-contour refusé
sinon /* Type_gradient = Trait */
    si visibilité (Gradient, Ecartmin) > 0
        alors pixel-contour accepté
        sinon pixel-contour refusé
```

- Pour simplifier les problèmes et pour obtenir une première solution, nous proposons d'approcher la courbe expérimentale de limite de visibilité par une droite, pour les deux types de contour.

Pour trouver l'équation de cette droite, nous proposons d'appliquer la technique des moindres carrés aux points expérimentaux de visibilité ambiguë. En effet, la droite qui minimise la distance à ces points se situe au cœur de la zone d'ambiguïté, ce qui correspond exactement à ce que nous recherchons. Pour choisir les points expérimentaux dont la visibilité transversale nous semble ambiguë, nous avons procédé de la façon suivante. Dans un premier temps, à partir d'une image quelconque, nous cherchons à l'œil nu des configurations locales où la présence d'une discontinuité semble incertaine. Ensuite, nous observons les valeurs numériques des niveaux de gris et nous décidons de la pertinence et de la localisation exacte d'un éventuel pixel de visibilité ambiguë. Le choix des configurations locales se fait dans diverses zones de l'image et dans différents types d'images. Nous essayons de choisir un ensemble de points représentatifs, en

répartissant nos choix de telle sorte que toutes les valeurs de Gradient soient rencontrées un nombre à peu près équivalent de fois. Nous essayons également de diversifier les configurations, en choisissant des problèmes de dégradé limite, des problèmes de texture, des problèmes de bruit ou une combinaison de ces problèmes. Nous calculons ensuite les valeurs de Gradient et Ecartmin pour chaque point et nous formons ainsi notre graphe. En tout, nous avons pris 85 points expérimentaux pour le type marche et 64 pour le type trait.

Il est important de donner les hypothèses de travail que nous avons fixées pour cette étude :

- 1) Notre expertise de la visibilité est satisfaisante.
- 2) Le nombre de points expérimentaux est suffisant.
- 3) Le poids de chaque couple de valeurs (Gradient, Ecartmin) est le même pour tous les points expérimentaux.
- 4) La répartition des configurations testées est satisfaisante.
- 5) La prise en compte d'éventuelles configurations non testées ne modifieraient pas l'allure générale du graphe expérimental.

Il est difficile de juger la validité de nos hypothèses. La visibilité étant fondée sur un critère local, le nombre de configurations à étudier est limitée, mais il en existe toutefois un grand nombre. Nous présentons en annexe 1 les deux graphes correspondant aux points expérimentaux choisis, avec les deux droites issues de l'approximation aux moindres carrés. La répartition des couples de valeurs est relativement homogène avec des écarts à la droite parfois significatifs, ce qui justifie partiellement nos hypothèses et nous laisse espérer une approximation suffisante de la courbe de visibilité minimale.

Pour appliquer la méthode des moindres carrés, si nous appelons X_i la valeur de Ecartmin et Y_i la valeur du Gradient, nous avons minimisé l'expression :

$$\sum_i (Y_i - a X_i - b)^2$$

En cherchant le passage par 0 des dérivées par rapport à a et à b , nous obtenons le système d'équations suivant :

$$(1) (\sum_i X_i Y_i) - a (\sum_i X_i^2) - b (\sum_i X_i) = 0$$

$$(2) (\sum_i Y_i) - a (\sum_i X_i) - b = 0$$

Nous trouvons finalement :

- Type marche : $a = 2.92$ $b = 6.04$

- Type trait : $a = 2.63$ $b = 3.96$

Les pixels-contours respectent alors l'inégalité suivante :

$$(3) \text{ Gradient} > a * \text{Ecartmin} + b$$

Notons que nous avons ici caractérisé une technique de seuillage adaptatif. Toutefois, nous cherchons à exprimer la qualité de la visibilité, nous préférons donc introduire une nouvelle grandeur qui soit grossièrement proportionnelle à la visibilité d'un pixel-contour. Sur le schéma de la figure 30, la visibilité est d'autant grande que le gradient est fort et que Ecartmin est faible. Nous proposons donc la notion intuitive de Gradient Relatif pour caractériser la visibilité. Nous dirons alors qu'un pixel-contour est visible si et seulement si le gradient relatif est supérieur à un seuil de visibilité, que nous fixons par convention à 0.

Pour définir le gradient relatif, nous pouvons reformuler l'équation (3) d'une façon strictement équivalente.

$$(3) \iff \text{Gradient} - a * \text{Ecartmin} - b > 0$$

$$\text{D'où :} \quad \text{Gradient relatif} = \text{Gradient} - a * \text{Ecartmin} - b$$

Le Gradient Relatif est ainsi une grandeur à peu près proportionnelle à la visibilité. En dessous de la valeur 0, rien n'est visible, au-dessus, nous considérons qu'il y a une visibilité transversale suffisante.

- Identification des dégradés :

Là encore, nous nous trouvons confrontés à un problème de visibilité. Quelle est en effet la limite entre un contour flou visible et un dégradé où aucune discontinuité ne doit être détectée ? Même avec un facteur d'échelle constant, le problème se pose. Selon notre expertise, un dégradé est une suite de gradients d'amplitude et de direction à peu près constante. Nous proposons donc de vérifier dans la direction du gradient les valeurs et direction de celui-ci. La formule générale de l'heuristique identifiant un dégradé sur un pixel P est :

soit Gradient et D la norme et la direction du gradient en P

soit Pdevant le pixel situé à 2 pixels devant P

soit Pderrière le pixel situé à 2 pixels derrière P

(devant et derrière étant définis en fonction de l'orientation de D)

soit Graddevant et Ddevant la norme et la direction du gradient en Pdevant

soit Gradderrière et Dderrière la norme et la direction du gradient en Pderrière

```

si ( ( (Graddevant > a*Gradient)
      et (Graddevant < 1/a*Gradient)
      et (Ddevant - D < Seuil) ) /* dégradé devant */
    et ( (Gradderrière > a*Gradient)
         et (Gradderrière < 1/a*Gradient)
         et (Dderrière - D < Seuil) ) /* dégradé derrière */
    )
alors le pixel-contour n'est pas visible

```

Le nouveau paramètre "a" est ce que nous appelons la limite de proportionnalité du gradient voisin. Comme nous l'avons défini, "a" est inférieur à 1. Après étude expérimentale sur une trentaine de points à la limite du dégradé, sélectionnés sur plusieurs images (notamment celles de la souris et de la femme), nous proposons les valeurs approximatives de a et Seuil suivantes :

$$a = 0.75$$

$$\text{Seuil} = 90 \text{ degrés} \quad (\text{ce qui correspond à un changement d'orientation de } 2 \text{ avec l'échelle de Freeman})$$

Ces valeurs ne sont sans doute pas optimales, mais elles permettent un traitement simple et relativement efficace des dégradés.

4.3 Formes longitudinales des contours

4.3.1 Critère de visibilité

L'aspect longitudinal ou global du contour se manifeste à plusieurs niveaux :

- (1) Au niveau du choix du pixel suivant lors de la croissance de contour. Cette étude a déjà été faite au 3.5 de ce chapitre.
- (2) Au niveau de la conservation du type du gradient pour tous les pixels du contour.
- (3) Enfin, au niveau de la taille minimale du contour permettant sa visibilité. C'est ce dernier aspect qui est discuté ici.

La longueur minimale d'un contour dépend du facteur d'échelle. Comme nous l'avons déjà précisé par ailleurs, nous considérons pour simplifier le problème que toutes les images fournies à notre détecteur ont le même facteur d'échelle, de telle sorte que la largeur d'une transition visible ne dépasse pas la taille de nos masques utilisés pour le calcul du gradient.

Pour déterminer cette longueur minimale, nous proposons d'observer des exemples de contours dont la visibilité longitudinale est limitée en raison de leur faible nombre de pixels. Nous présentons en annexe 3 des images diverses pour évaluer notre détecteur. Deux images nous intéressent plus particulièrement, celle de la souris et celle du muscle. L'image de la souris est particulièrement instructive car la texture du bois est composée d'un grand nombre de petites fibres claires ou foncées nettement visibles. En revanche, dans l'image du muscle, aucune discontinuité ne peut être identifiée dans les cellules claires qui sont pourtant texturées. Nous différencions donc deux types de texture :

- Les textures simples où les variations locales sont fortes mais où aucune discontinuité n'est visible. Le bruit comporte généralement des variations du même type.
- Les textures complexes avec des motifs ou des éléments simples dont les limites visuelles constituent des discontinuités à détecter.

Comme pour le gradient relatif, nous proposons de prendre des points expérimentaux de visibilité limitée et de déterminer une fonction de décision simple pour valider ou rejeter un contour. Pour cela, nous choisissons les paramètres du Gradient Moyen sur le contour et la longueur de celui-ci. En effet, intuitivement, un petit contour dont le gradient est fort est aussi visible qu'un contour à peine plus long mais de gradient plus faible.

Nous présentons nos résultats expérimentaux figure 31.

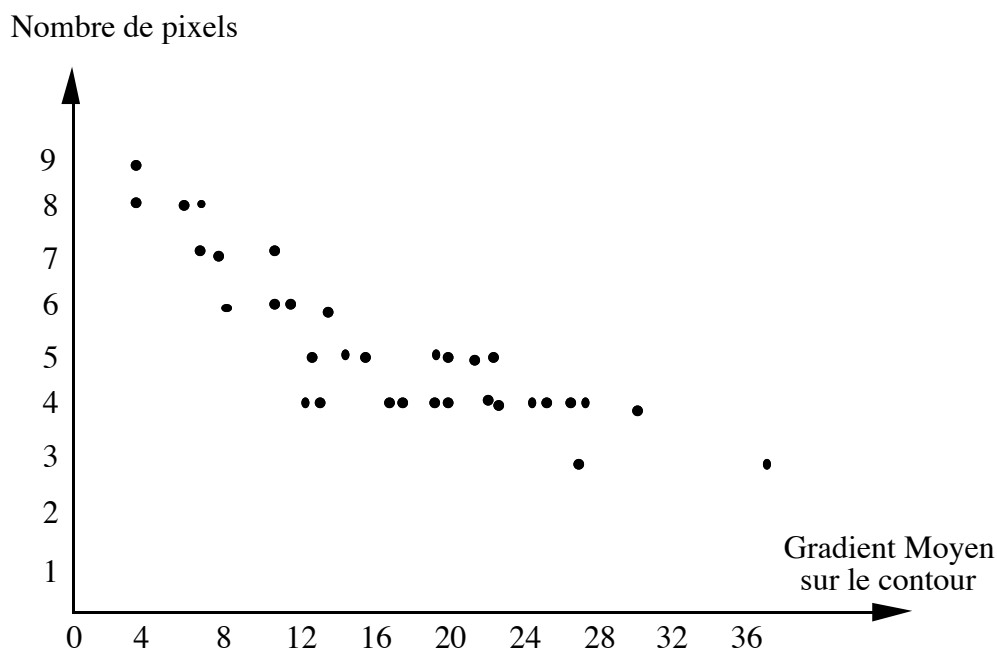


Figure 31 : Contours de visibilité limitée.

4.3.2 Proposition d'heuristique

Pour approcher simplement la limite de visibilité longitudinale de la figure 31, nous proposons la formule générale suivante :

- (1) si (nbpixels > seuil1) alors contour accepté
- (2) sinon si (nbpixels < seuil2) alors contour refusé
- (3) sinon si $\text{Visibilité}(\text{nbpixels}, \text{grad_moyen}) > \text{Seuil3}$
alors contour accepté
sinon contour refusé

Nous pourrions approcher la courbe (3) par la technique des moindres carrés, mais étant donné les approximations qui sont faites, nous proposons simplement une fonction en escalier, illustrée figure 32.

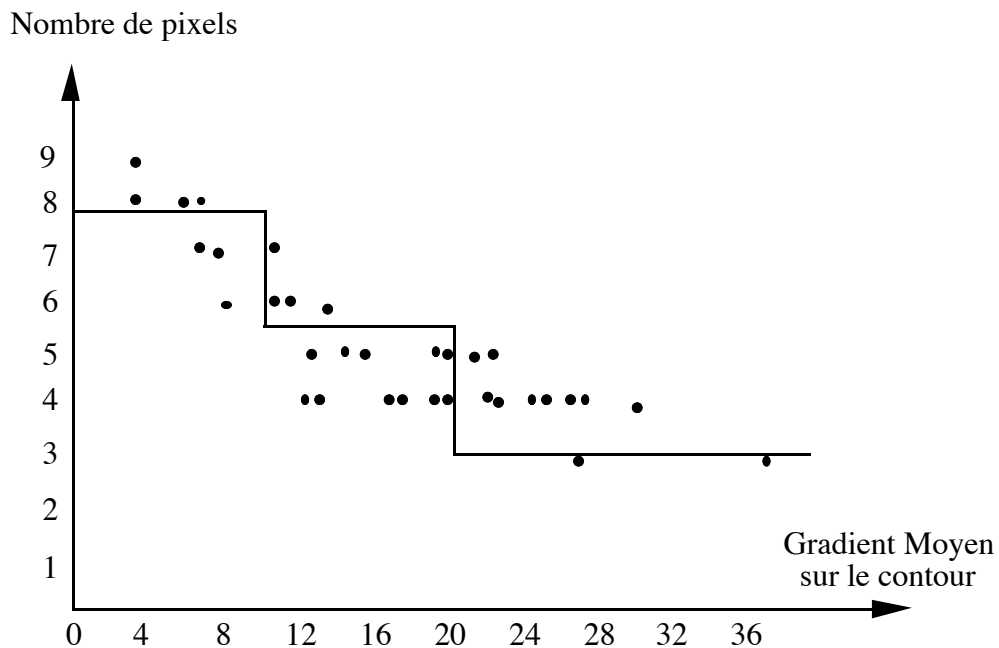


Figure 32 : Fonction escalier approchant la limite de visibilité.

Les fonction en escalier a été obtenue en distinguant quatre intervalles de valeurs pour la variable "nombre de pixels", et en calculant pour chacun d'eux la moyenne des gradients arrondie à la valeur entière la plus proche. La formule exacte est la suivante :

- si (nbpixels \geq 8) alors contour accepté
sinon si ((grad_moyen > 10) et (nbpixels \geq 6)) alors contour accepté
sinon si ((grad_moyen > 20) et (nbpixels \geq 4)) alors contour accepté
sinon contour refusé

Notre heuristique reste toutefois insuffisante pour un cas très particulier qui conduit à un dédoublement de contour : en effet, si un germe est validé tout près d'un contour avec forts gradients, le contour formé rejoint celui-ci un peu plus loin en suivant des gradients de plus en plus forts, ce qui lui confère une bonne visibilité longitudinale d'après notre formule et occasionne ainsi une erreur. Pour remédier à ce problème, nous proposons d'imposer à tous les contours que trois pixels au moins aient une valeur de gradient respectant la valeur de maximum local dans la direction de celui-ci, ce qui est peu mais suffisant pour éviter ces erreurs.

4.4 Ajustement manuel

La détermination des coefficients du gradient relatif reste approximative. En effet, l'équation des deux droites dépend directement des points expérimentaux choisis. Or, ces points sont approximativement situés à la limite de visibilité, mais il est possible que nous ayons introduit un biais systématique qui pénalise la détection. De plus, les droites que nous avons proposées sont des approximations de courbes expérimentales dont l'allure n'est pas tout à fait rectiligne. Pour certaines valeurs de gradient et de Ecartmin, le gradient relatif peut donc être un peu trop faible ou un peu trop fort.

Pour remédier à ce problème, nous proposons de laisser à l'utilisateur la possibilité d'affiner les coefficients du Gradient Relatif à l'aide d'un ajustement "manuel". Pour la plupart des images, les coefficients que nous proposons par défaut donnent de bons résultats (voir la partie évaluation de ce chapitre), mais il arrive parfois qu'un ou deux contours soient mal détectés. Dans ce cas, un changement de coefficient peut s'avérer intéressant pour obtenir le résultat recherché. Il est d'ailleurs remarquable que dans la plupart des images, très peu de contours sont à la limite de visibilité. Une petite variation des coefficients permet alors d'optimiser la détection des contours délicats sans perturber le résultat des contours "évidents".

L'ajustement manuel permet également de s'adapter au type de l'image. Par exemple, pour des images de synthèse fortement bruitées, la détection des contours de visibilité difficile conduit à des valeurs particulièrement fortes de Ecartmin. Il suffit en fait de diminuer la pente "a" de la droite (de l'ordre de 10 à 20%) pour détecter correctement ces contours tout en préservant la robustesse au bruit.

De même, remarquons que la valeur de "b" pour le type marche est supérieure à 6. Cela implique que les contours de gradients très faibles, inférieurs à 6, ne peuvent être détectés. Si un tel contour est présent dans une image, il suffit de choisir une valeur de "b" plus petite, par exemple 4, pour obtenir sa détection. Comme en général il existe peu

de contours de visibilité ambiguë, cette modification conduit en moyenne à de meilleurs résultats pour l'image traitée.

Notons également que si nous choisissons $a = 0$, notre détecteur effectue un simple seuillage du gradient avec la valeur b . Il est intéressant de comparer les résultats de notre détecteur avec la valeur de a par défaut et $a = 0$. Nous observons ainsi le gain obtenu grâce à la prise en compte des informations sur les régions voisines.

Enfin, nous laissons également à l'utilisateur la possibilité de seuiller le gradient ou le gradient relatif par la technique d'hystérésis, ceci afin de pouvoir comparer chaque méthode et optimiser le traitement pour une image donnée. Ces choix multiples permettent également un approfondissement de notre expertise de la détection des contours, ce qui est rappelons-le, un des problèmes majeurs pour la construction d'un détecteur selon une démarche expérimentale.

5 Evaluation

5.1 Protocole expérimental

L'évaluation est un problème très important, de plus en plus souligné par la communauté scientifique en vision par ordinateur [Pavl 92], [DIAL 91], [GREC 91]. Le problème est qu'il n'existe pas de protocole précis permettant d'effectuer une évaluation détaillée pertinente des résultats. Nous avons présenté au 2.3 plusieurs méthodes d'évaluation et nous avons proposé une approche pragmatique pour l'évaluation des détecteurs de contours. Celle-ci consiste à confronter les résultats des algorithmes à l'expertise humaine, en examinant les valeurs numériques des niveaux de gris, ceci afin de placer notre analyse au niveau approprié d'observation et d'enrichir notre expertise de la détection des indices visuels.

Notre approche étant de plus expérimentale, l'évaluation revêt ici une importance d'autant plus grande. Nous étudions dans cette partie le comportement de notre algorithme pour différents types d'images et différents types de problèmes.

Nous avons choisi d'étudier la réponse de notre détecteur à une batterie de tests, en utilisant des échantillons d'images réelles. Après avoir sélectionné six images de nature différente dans lesquelles sont présents les principaux types de difficulté, nous retenons dans chacune d'elle des échantillons correspondant à une fenêtre d'environ 15 pixels sur 15 (voir annexe 2). Chaque échantillon représente un type de difficulté que nous désirons étudier.

Dans un premier temps, nous proposons une étude qualitative de chaque type de difficulté, en analysant les niveaux de gris d'un échantillon représentatif de la classe. Cette étude qualitative correspond en fait à une explicitation grossière de notre expertise. Nous identifions ainsi les problèmes spécifiques de chaque type et nous essayons de donner le résultat théorique optimal qui fait référence. Munis de ces références, nous sommes alors capables d'effectuer une évaluation globale qualitative de notre détecteur.

L'avantage de cette évaluation est de pouvoir d'adapter à la difficulté. Certaines erreurs sont en effet relativement acceptables alors que d'autres sont manifestement grossières. Cette évaluation pragmatique permet donc de s'adapter à la difficulté réelle et à l'erreur relative au contraire de l'évaluation quantitative qui est régie par la loi du "juste ou faux" sans véritable nuance. L'essentiel est selon nous que les erreurs soient comprises. C'est un des points clés de notre approche dont le but est, rappelons le, d'améliorer notre expertise pour construire un détecteur fondé sur des décisions heuristiques.

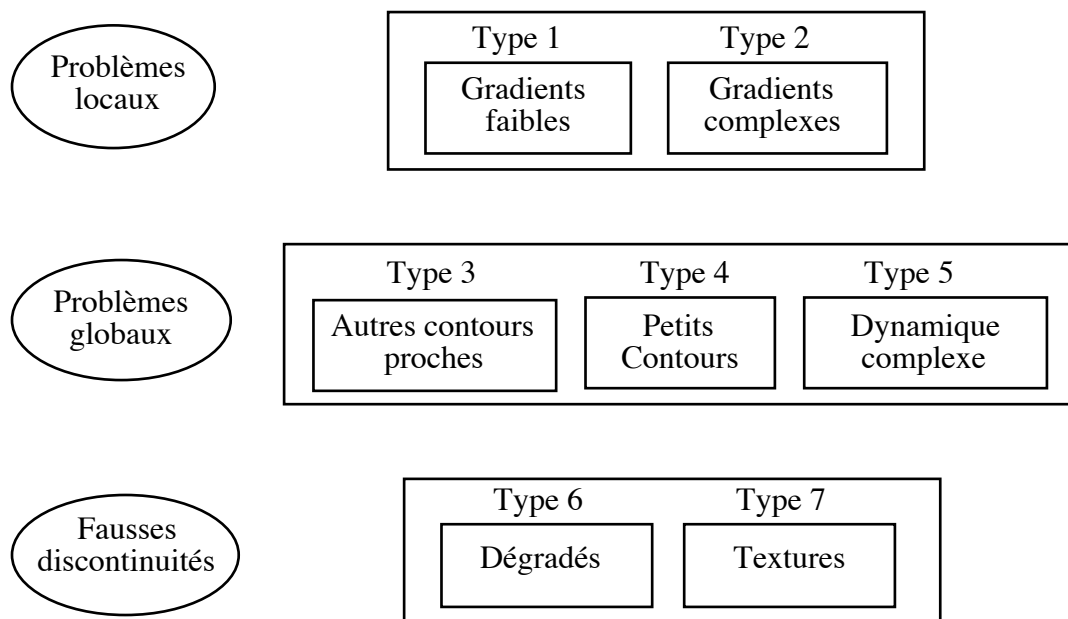


Figure 33 : Différentes catégories de tests.

Nous regroupons les tests en plusieurs grandes catégories (voir figure 33). Nous distinguons en premier lieu les problèmes locaux des problèmes globaux. Dans le premier cas, nous avons deux types de problèmes : la détection des discontinuités avec gradients faibles et la forme plus ou moins composite de la transition (entre marche et trait, pente longue, etc...).

Dans le deuxième cas, nous avons le problème de la présence d'autres contours dans le voisinage immédiat (jonctions incluses), des petites discontinuités, ainsi que les problèmes liés à la dynamique du contour. Par dynamique du contour, nous entendons tous les changements de gradient, de voisinage, de forme composite le long de la discontinuité.

Enfin, la dernière catégorie de tests, appelée "tests négatifs" regroupent les zones texturées (texture simple sans discontinuité visible) et les dégradés dans lesquelles aucun contour n'est désiré. Pour chaque type de difficulté, nous proposons souvent des tests complémentaires pour étudier la réponse du détecteur en fonction de l'orientation de la discontinuité.

6 images tests marquées avec 48 échantillons sont présentées en annexe 2. Nous ne présentons section suivante que quelques uns de ces échantillons en commentant les difficultés qui s'y rattachent. Bien que nous présentions des résultats spécifiques aux 48 échantillons, nous avons souvent été amenés à examiner les valeurs numériques d'autres régions provenant parfois d'autres images pour comprendre les difficultés que notre algorithme avait du mal à résoudre.

5.2 Tests locaux

5.2.1 Gradients faibles (type 1)

Les 5 échantillons 1, 2, 3, 4 et 5 représentant les difficultés de type gradient faible correspondent, respectivement, aux échantillons numérotés 6, 7, 8, 9 et 12 marqués sur l'image du muscle (voir annexe 2).

Les gradients de faibles amplitudes posent le problème de la sensibilité du détecteur. Pour des détecteurs classiques, c'est l'utilisateur qui choisit le seuil minimal de gradient et qui détermine donc sa sensibilité de manière quelque peu arbitraire, en fonction de l'image et de la nécessité. Quoiqu'il en soit, le problème majeur est ici de détecter les discontinuités dont le gradient est faible tout en restant robuste à la texture et au bruit. Lors de l'évaluation, il est ainsi judicieux de présenter simultanément les résultats de la robustesse à la texture et de la détection des discontinuités avec faibles gradients. Pour illustrer les problèmes de type 1, nous nous sommes intéressés en particulier à l'image des cellules. Nous présentons ci-dessous les valeurs numériques d'un échantillon.

- GRADIENT FAIBLE, ECHANTILLON 1 : image du muscle, numéro 6

Position du pixel en haut à gauche : 87 72

Largeur de la fenêtre : 14

Hauteur de la fenêtre : 21

57	57	58	62	61	58	60	83	<u>100</u>	144	164	160	174	181
59	59	58	60	60	58	57	69	<u>132</u>	168	152	143	155	177
58	58	57	59	60	60	57	65	<u>114</u>	160	151	128	142	162
57	57	56	57	58	59	59	64	<u>110</u>	164	171	123	111	114
55	55	53	55	57	58	59	62	<u>85</u>	132	180	195	181	159
54	53	53	54	56	58	59	61	80	131	175	204	124	198
52	53	52	54	56	59	60	61	89	159	175	156	<u>120</u>	<u>88</u>
52	52	52	54	56	57	59	62	97	189	165	<u>97</u>	68	62
52	52	52	53	55	57	60	63	88	130	95	67	60	59
52	52	52	53	55	58	60	63	<u>74</u>	72	61	58	57	58
52	52	52	54	56	59	60	65	<u>68</u>	62	58	57	58	57
53	52	53	55	56	59	63	<u>70</u>	67	58	55	56	58	57
53	53	53	55	58	61	67	<u>73</u>	62	56	54	55	55	56
54	54	54	57	60	64	70	<u>73</u>	58	54	53	54	54	56
55	55	56	60	63	67	<u>77</u>	68	57	54	52	54	55	56
58	59	60	63	67	76	96	71	57	54	52	54	56	58
66	68	70	72	82	116	119	74	57	54	52	53	55	56
<u>102</u>	<u>97</u>	<u>106</u>	<u>129</u>	166	197	143	76	58	54	54	54	55	57
138	147	160	177	198	206	134	72	57	54	55	55	55	57
176	179	196	178	144	153	<u>119</u>	71	57	55	55	56	56	56
170	180	192	186	150	119	<u>102</u>	70	57	55	54	56	56	56

Commentaires :

Les cellules noires sont très homogènes et présentent des variations de niveaux de gris internes presque négligeables. Cette image est en fait très difficile car il y a souvent

des petits paliers (avec gradients de 4 ou 5) présents sur quelques pixels seulement, et la décision de validation se fait parfois à un pixel près. Certaines transitions sont à peine visibles dans l'image à cause d'un manque de contraste de nos consoles, ce qui pourrait laisser penser que des faux contours ont été détectés. L'examen des valeurs numériques permet d'identifier l'éventuelle discontinuité de façon plus objective. La frontière entre deux cellules noires est souvent composée de faibles gradients de type trait. Les différences de niveaux de gris entre la frontière et la cellule sont de l'ordre de 10, parfois moins. Dans ce cas précis, la frontière est de plus de petite longueur, ce qui rend le problème plus complexe. Notons également que le trait est en fait un double contour qui se sépare en deux parties pour faire le tour de chaque cellule. Il en résulte des problèmes de jonction et de positionnement difficiles, marqués en caractères gras dans l'échantillon présenté, alors que les pixels frontières sont en caractères soulignés.

5.2.2 Gradients complexes (type 2)

Les échantillons 1, 2, 3, 4, 5, 6 et 7 représentant les difficultés de type gradient complexe correspondent, respectivement, aux échantillons 3, 4, 5 de l'image du muscle, au 4 de l'image du bateau et aux 2, 3 et 6 de l'image IRM. Les échantillons sont tous marqués sur les images présentées en annexe 2.

Nous appelons gradients complexes les pentes qui n'ont ni la forme de marches, ni la forme de toits, mais une forme composite singulière (figure 30). Par exemple, un reflet sur le bord d'un objet se manifeste souvent par un pic étroit au milieu de la transition.



Figure 30 : A \Leftrightarrow signal simple; B, C, D \Leftrightarrow signal complexe

- GRADIENT COMPLEXE, ECHANTILLON 1 : image du muscle, numéro 3

Position du pixel en haut à gauche : 35 132

Largeur de la fenêtre : 18

Hauteur de la fenêtre : 18

169 172 180 182 182 195 195 194 203 205 205 201 209 210 200 179 176 180
 203 197 197 197 184 193 197 199 191 184 190 201 187 186 166 165 170 171
 210 189 181 186 183 189 175 189 207 193 202 207 181 163 155 161 164 151
 180 193 187 184 178 174 171 178 203 213 212 212 214 193 166 175 181 166
 157 170 179 169 174 192 203 195 195 192 193 172 177 178 179 162 145 128
 152 143 149 159 160 178 172 179 176 172 186 190 193 182 168 148 136 146
 152 145 158 168 157 164 177 190 192 158 165 186 180 183 174 164 206 206
 161 167 184 168 178 177 171 163 163 183 198 181 198 209 222 222 179 157
 156 176 198 178 188 194 194 178 141 157 184 209 230 223 192 172 167 163
 157 161 167 195 203 179 166 195 212 228 215 206 191 173 164 172 156 171
 119 126 121 138 187 219 219 216 196 189 168 151 148 159 166 173 173 173
 148 173 187 203 202 187 160 155 168 176 164 183 193 189 181 157 168 164
203 176 144 127 143 149 153 177 171 177 196 200 179 171 181 188 179 182
 101 118 115 125 144 153 145 156 177 170 165 187 195 193 203 194 183 189
 118 127 142 143 163 178 169 159 175 186 174 188 202 202 171 171 182 174
 110 134 151 144 177 205 195 186 183 202 190 172 181 187 188 189 192 188
 137 158 174 166 180 178 165 165 179 185 197 185 183 194 179 175 157 154
 153 146 155 170 186 202 185 180 187 193 198 195 205 222 212 192 186 180

Commentaires :

A propos de l'échantillon présenté : bien qu'à l'œil nu un trait simple semble présent, le signal a une forme très irrégulière en raison de la texture des cellules grises de part et d'autre du trait. L'analyse des valeurs de gris révèle un problème bien plus complexe qu'il n'y paraît car le trait ne semble pas très marqué par rapport aux autres variations locales du voisinage (le trait correspond aux caractères soulignés). Il semble que l'alignement des forts gradients joue un rôle non négligeable sur la visibilité de ce trait, ce qui renverrait le problème au niveau de la dynamique et à une vision plus globale. Ce cas très délicat doit être étudié en même temps que la robustesse à la texture et au bruit. Notons la présence de valeurs de gris supérieures à 200 au centre et au-dessus de la discontinuité (soit du même ordre que pour celle-ci) mais qui ne correspondent à rien de visible, les variations étant très irrégulières et mal réparties.

D'autres cas compliqués existent dans l'image du muscle. Par exemple à la frontière entre une cellule grise et une cellule noire, la pente du gradient est souvent longue du côté de la cellule noire, ce qui occasionne des gradients loin de la discontinuité non négligeables, parfois maximum locaux dans la direction de celui-ci, et provoque des faux germes de contour. En bordure interne des cellules grises texturées, il existe des zones plus foncées qui pourraient être identifiées parfois comme traits et parfois comme dégradés de textures, ce qui rend le gradient local particulièrement complexe.

Dans presque toutes les images il existe des gradients complexes. La plupart du temps, celui-ci est dû à la frontière entre une zone de texture et une autre région (bord de la souris, bords du visage dans l'image de la femme, bord des organes dans l'image RMN). Dans d'autres cas, il correspond à un trait séparant deux régions d'intensité différente (jointures du bateau par exemple).

5.3 Tests globaux

5.3.1 Proximité d'autres contours (type 3)

Les échantillons 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 et 11 représentant les difficultés de type "contour proche" correspondent, respectivement, aux échantillons 10 et 11 de l'image du muscle, aux 1, 2 et 3 de l'image du bateau et aux 1, 2, 3, 4, 5 et 6 de l'image de l'objet polyédrique.

Nous abordons ici les problèmes liés à la détection des croisements de contours ainsi que les problèmes dûs à la présence d'autres contours dans le voisinage immédiat. La localisation précise du contour est ici une des difficultés majeures.

- CONTOURS PROCHES, ECHANTILLON 1 : image du muscle, numéro 10

Position du pixel en haut à gauche : 69 1

Largeur de la fenêtre : 18

Hauteur de la fenêtre : 14

50	47	45	46	50	70	<u>120</u>	112	86	94	103	134	152	146	144	129	115	132
51	48	46	48	55	<u>82</u>	140	<u>130</u>	<u>87</u>	92	105	122	127	127	143	135	128	131
52	49	46	48	58	<u>94</u>	138	<u>118</u>	77	<u>80</u>	110	127	125	142	139	132	149	154
51	47	46	48	69	<u>151</u>	180	<u>144</u>	90	<u>74</u>	98	106	114	132	145	145	131	133
47	46	46	49	<u>90</u>	182	221	220	<u>160</u>	98	<u>80</u>	<u>100</u>	110	136	169	148	169	142
48	46	45	53	<u>90</u>	137	180	216	219	<u>179</u>	130	124	119	125	129	133	149	143
45	45	47	53	<u>58</u>	<u>62</u>	<u>78</u>	<u>127</u>	181	206	<u>191</u>	129	107	114	131	150	142	130
45	48	52	<u>54</u>	55	56	60	69	<u>96</u>	<u>144</u>	<u>190</u>	179	119	113	150	160	148	151
48	52	<u>55</u>	54	54	55	58	60	63	74	<u>103</u>	140	139	116	136	153	161	166
<u>56</u>	<u>56</u>	54	52	51	52	55	57	58	60	64	<u>81</u>	118	134	144	162	162	150
55	52	51	49	49	50	51	53	54	56	57	61	<u>75</u>	126	154	150	136	137
51	49	49	49	48	48	49	49	49	50	52	54	60	<u>75</u>	<u>128</u>	143	110	112
49	48	48	48	48	47	46	47	47	48	47	49	53	58	71	<u>114</u>	157	141
47	47	47	47	48	46	46	46	46	46	46	48	49	51	55	66	<u>107</u>	140

Commentaires :

A propos de l'échantillon présenté : un trait entre deux cellules noires avec très faibles gradients vient rejoindre une discontinuité correspondant au contour d'une petite zone de fond marquée en gras. Un trait noir juste à côté de la zone blanche est en bordure interne d'une cellule texturée. Incontestablement une zone à problèmes. Visuellement, l'identification de chaque discontinuité est plus aisée (annexe 2), en particulier si le lecteur n'est pas habitué à cette présentation.

La présence d'autres discontinuités dans le voisinage immédiat biaise souvent le calcul du gradient, occasionnant des erreurs, c'est ce que nous appelons les "effets de bords". Là encore, cette difficulté est très fréquente dans toutes les images que nous présentons. Par exemples, le bord de la souris et le contour de la structure partiellement

sphérique sont très proches. Les cordages des bateaux posent également des problèmes de proximité et de jonctions.

L'image d'un objet polyédrique est présentée en annexe 2 pour illustrer les problèmes de détection et de localisation près des coins. Le problème de telles configurations est la pertinence de la valeur du gradient. En effet, aucun de nos masques n'est adapté aux coins et le calcul est donc biaisé. De plus, comme nous l'avons déjà fait observer, le gradient n'est pas toujours maximal localement près des jonctions ce qui peut entraîner une intersection mal localisée.

5.3.2 Les petits contours (type 4)

Les échantillons 1, 2, 3 et 4 représentant les difficultés de type petit contour correspondent, respectivement, aux échantillons numérotés 3, 4, 5 et 6 sur l'image de la souris présentée en annexe 2.

Le problème des petits contours est leur visibilité. Comme nous pouvons le constater sur l'image de la souris, il est possible visuellement de déterminer dans la plupart des cas si une petite discontinuité est présente ou pas. Il reste des cas ambigus bien sûr, mais il suffit d'en tenir compte pour garder une évaluation qualitative satisfaisante.

Notons que les petites discontinuités sont parfois très importantes. Par exemple, les textures complexes sont souvent composées d'éléments plus petits. La détection de ces petites discontinuités est sans doute le point de départ essentiel à l'identification de ce genre de texture.

La difficulté majeure est ici de faire la part entre zone texturée ou bruitée où aucune discontinuité n'est visible et une petite discontinuité réelle. Il s'agit également de ne pas détecter des discontinuités de 20 pixels lorsque celle-ci n'en fait que 4 ou 5.

ECHANTILLON 1 : image de la souris, numéro 6 (voir page suivante).

Position du pixel en haut à gauche : 100 224

Largeur de la fenêtre : 15

Hauteur de la fenêtre : 19

Commentaires :

A l'œil nu, de nombreux petits traits noirs ou blancs sont visibles dans la région texturée correspondant au bois. Sur l'échantillon présenté, celui-ci est blanc et mesure environ 8 pixels (caractères soulignés). Ces traits constituent la texture de ce bois. Ils sont même sans doute les indices caractéristiques de cette texture et ils permettraient son identification si nécessaire. De manière générale, nous sommes d'avis que les petits traits sont des indices élémentaires d'une extrême importance pour l'identification de textures

complexes, qui correspondent elles à des indices visuels de plus haut niveau. Quoiqu'il en soit, toutes les discontinuités visibles doivent être détectées et il est donc souhaitable de détecter les nombreuses petites discontinuités présentes dans l'image de la souris.

D'autres petites discontinuités sont présentes dans les autres images. Nous avons déjà discuté des petites zones plus foncées en bordure interne des cellules grises de l'image du muscle. Il existe également des petits contours dans les cheveux de l'image de la femme ou parmi les détails de l'image du bateau. Ces petits problèmes sont en fait loin d'être rares.

120 143 148 125 113 99 105 119 137 127 111 104 116 140 131
 103 127 174 156 125 115 123 134 142 128 108 106 114 145 132
 95 89 139 191 133 123 147 132 120 130 122 117 124 137 133
 120 104 102 129 141 136 156 135 102 121 125 124 133 136 145
 134 130 122 128 139 137 154 162 134 127 131 130 136 133 143
 150 149 132 128 142 150 167 175 154 132 135 137 141 136 134
 137 148 147 127 131 163 192 180 153 128 128 133 143 134 126
 144 154 156 130 124 157 194 188 154 125 136 149 152 149 148
 152 178 186 159 129 159 196 188 153 130 151 155 147 146 148
 151 154 165 164 131 155 194 192 173 145 140 147 143 146 151
 154 145 139 144 135 156 192 198 187 156 137 145 144 136 129
 142 134 142 159 163 159 183 202 198 174 136 125 126 127 120
 150 134 147 168 175 164 171 202 200 186 147 124 122 137 117
 140 134 156 172 166 161 161 193 194 176 150 140 140 142 128
 145 148 163 172 148 148 162 163 164 152 140 148 151 135 135
 137 145 162 176 148 131 136 102 117 148 133 140 145 137 143
 141 142 162 180 153 127 128 108 122 149 142 141 145 147 141
 134 144 167 177 152 129 129 125 140 144 145 153 155 153 144
 112 135 169 171 154 141 132 130 139 143 144 149 147 135 147

5.3.3 Dynamique complexe (type 5)

Les échantillons 1, 2, 3, 4, 5, 6, 7, 8, 9 et 10 représentant les difficultés de type dynamique complexe correspondent, respectivement, aux échantillons numérotés 7, 8, 9 et 10 de l'image de la souris (contour de celle-ci), aux 7, 8 et 9 de l'image de la femme (bord du visage) et aux 1, 4 et 5 de l'image IRM (voir annexe 2).

Un autre problème difficile de la détection des contours est celui du changement de propriétés le long de la discontinuité. Par exemple, le gradient peut chuter brutalement, la forme de la pente peut varier et les régions voisines peuvent également changer de propriétés. Il est difficile d'étudier le comportement de l'algorithme dynamiquement. Pour cela, nous proposons une focalisation sur plusieurs endroits d'une même discontinuité présentant une dynamique complexe. Nous proposons également une étude attentive des zones où se situent les difficultés.

ECHANTILLON 5, 6 et 7 : image de la femme, numéros 7, 8 et 9

Numéro 5 :

Position du pixel en haut à gauche : 168 141

Largeur de la fenêtre : 18

Hauteur de la fenêtre : 14

174	169	116	121	151	166	158	156	168	<u>136</u>	69	57	<u>83</u>	144	156	165	156	159
168	177	124	147	153	163	154	154	167	<u>125</u>	72	<u>86</u>	133	146	158	164	166	160
158	174	140	164	157	158	157	156	159	<u>125</u>	120	<u>118</u>	137	143	163	153	165	158
156	168	149	165	159	158	156	152	152	<u>132</u>	<u>123</u>	138	137	151	153	153	162	155
152	156	156	152	160	162	153	155	145	141	137	145	141	160	149	155	153	151
153	151	154	151	158	163	146	149	144	156	151	146	154	160	153	151	148	161
149	150	153	149	159	151	144	144	149	160	160	148	162	161	155	157	157	159
152	148	152	154	148	149	135	<u>143</u>	152	164	163	153	164	161	156	159	157	164
150	151	152	154	143	143	135	<u>144</u>	155	165	159	153	162	158	164	157	164	157
143	153	154	153	127	133	<u>128</u>	150	153	160	155	165	154	165	157	156	164	163
142	157	148	147	133	133	<u>141</u>	156	146	160	158	167	156	168	159	160	163	159
151	151	152	146	139	<u>139</u>	148	155	154	157	165	162	164	162	161	160	160	163
145	152	152	143	140	<u>137</u>	154	161	153	157	161	164	168	154	162	159	155	155
147	149	154	141	137	<u>145</u>	156	164	155	157	159	165	52	160	157	163	156	161

Numéro 6:

Position du pixel en haut à gauche : 164 161

Largeur de la fenêtre : 18

Hauteur de la fenêtre : 13

118	122	137	142	129	147	<u>148</u>	160	163	166	157	163	160	165	165	156	161	160
118	124	138	143	146	151	<u>146</u>	167	157	166	160	158	161	163	162	161	166	155
121	130	140	146	153	152	<u>152</u>	158	165	161	161	160	162	159	162	167	153	161
128	136	139	157	156	<u>149</u>	158	160	159	167	154	166	158	160	164	165	158	163
141	145	149	159	152	<u>143</u>	166	153	159	163	153	166	160	161	166	164	163	166
148	148	153	152	149	<u>146</u>	164	163	152	167	152	171	159	167	163	169	159	160
160	148	155	151	151	<u>146</u>	167	156	163	163	163	162	163	158	165	163	164	158
157	154	158	151	<u>148</u>	155	158	157	166	159	155	164	156	168	158	170	152	159
156	152	157	151	153	156	157	162	167	153	166	154	159	165	158	166	157	156
153	160	152	155	155	163	157	161	159	155	164	156	157	162	161	163	161	164
162	154	156	<u>148</u>	163	159	165	159	162	159	165	157	156	166	163	157	166	161
164	152	150	<u>143</u>	161	162	162	167	156	157	160	157	158	159	166	155	162	166
156	146	151	<u>141</u>	163	156	163	165	161	153	162	154	168	159	162	153	169	158

Numéro 7 :

Position du pixel en haut à gauche : 150 181

Largeur de la fenêtre : 18

Hauteur de la fenêtre : 14

161	142	145	147	145	155	156	155	167	154	164	151	162	153	152	146	163	158
160	166	153	157	161	158	155	164	156	163	158	159	153	157	145	153	167	155
164	178	164	161	166	159	168	161	161	161	153	160	157	148	148	157	159	163
172	176	172	163	169	166	174	163	173	155	159	157	156	146	147	163	158	165
178	175	175	179	167	175	179	170	166	161	157	159	150	150	148	166	159	165
185	173	185	181	174	178	169	175	160	159	161	163	149	152	156	165	163	163
190	180	182	185	178	175	167	178	158	158	158	157	148	157	163	157	169	154
192	175	184	181	192	171	173	183	160	163	163	156	151	158	165	160	155	167
180	178	177	186	185	169	176	173	164	164	159	154	157	161	164	160	160	164
179	188	178	179	183	173	173	165	163	154	159	148	169	163	159	163	163	162
178	189	185	173	180	176	169	167	158	154	154	148	162	160	160	164	158	169
171	189	187	179	182	174	169	165	156	159	153	154	159	161	158	164	158	159
171	183	182	180	184	171	161	156	163	151	157	154	161	160	163	158	164	159
167	175	174	180	183	165	164	153	156	155	154	167	162	156	157	169	159	160

Commentaires :

Le bord de la joue est particulièrement délicat à détecter, bien que cela ne semble pas poser de problème à l'œil humain. D'un côté, une zone légèrement texturée et de l'autre un dégradé irrégulier correspondant à la joue de la femme. Mais le problème se complique encore vers le bas avec une baisse du gradient et un début plus foncé du côté du visage ce qui rend le gradient complexe. Encore un peu plus bas, l'étude des valeurs numériques ne permet plus de localiser la frontière exacte du visage qui se confond alors avec le fond. Nous avons donc ici presque toutes les difficultés réunies autour de la même discontinuité. Il est important de remarquer qu'une vision globale n'est pas possible pour un détecteur qui se veut local et qu'il est donc normal de ne détecter ici qu'une partie de la frontière du visage. Nous pouvons même rajouter que l'examen des valeurs numériques est plus global et plus adaptatif que n'importe quel détecteur ce qui nous permet de qualifier les régions voisines de fin de dégradé, de fin de texture et de localiser correctement une grande partie du contour du visage. Un détecteur est en revanche plus "myope" et la décision est d'autant plus difficile que les informations disponibles peuvent tout aussi bien correspondre à celles d'un dégradé ou d'une texture. Il ne faut donc pas que la détection de ce contour entraîne des erreurs dans d'autres parties de l'image ou dans d'autres images. D'ailleurs, il nous semble qu'une sous-détection correcte est préférable à une détection un peu plus poussée mais comportant quelques erreurs. En effet, dans le premier cas, il s'agit de rajouter des informations et de perfectionner les heuristiques pour améliorer les résultats alors que dans le deuxième cas, il s'agit d'opérer une correction, ce qui est beaucoup plus délicat. Quoiqu'il en soit, le contour situé dans l'échantillon numéro 6 doit pouvoir être détecté, mais une mauvaise détection ne sera pas considérée ici comme une grosse erreur. Notons qu'une détection des contours est sans doute insuffisante pour de telles images, si le but est d'identifier le visage. Une coopération avec un détecteur de régions est ici souhaitable pour augmenter le nombre et la qualité des indices visuels.

Les problèmes de dynamique du contour sont très fréquents et sont la cause de nombreuses erreurs ou imperfections. Il s'agit là, selon nous, d'une des plus grandes difficultés de la détection des contours. Il est sans doute dommage que celle-ci ait été un peu ignoré lors de la conception de la plupart des détecteurs.

5.4 Tests négatifs

5.4.1 Dégradés

Les échantillons 1, 2 et 3 représentant les difficultés de type dégradé correspondent, respectivement, aux échantillons numérotés 4, 5 et 6 sur l'image de la femme présentée en annexe 2.

Des dégradés apparaissent souvent dans les images réelles et suscitent de nombreux problèmes. Au niveau de la détection, il est parfois difficile de se prononcer sur la présence ou pas d'une discontinuité. Un dégradé peut en fait être considéré comme un contour "marche" dégénéré. Les configurations intermédiaires entre le dégradé et le contour "marche" ne sont pas toujours faciles à classer, surtout si la dynamique est complexe. Il y a de plus un problème de localisation car le gradient est sensiblement identique d'un pixel à l'autre. L'évaluation reste ici difficile et dans certains cas, plusieurs solutions sont acceptables. A travers les problèmes des dégradés, nous voyons les limites de la détection des contours. Un dégradé est un indice visuel comme un autre, mais il ne rentre dans aucune catégorie précise de discontinuité. La détection de ce type de configuration en même temps que les discontinuités classiques pourrait sans doute apporter un plus non négligeable.

Pour les échantillons relatifs au dégradé que nous proposons, provenant essentiellement de l'image de la femme, il existe souvent une ligne de gradients un peu plus forts et il est donc difficile de déterminer si oui ou non de telles discontinuités doivent être détectées. Dans d'autres images, le dégradé peut correspondre à un contour un peu flou, un angle un peu raboté d'objet polyédrique ou plus généralement une surface courbe comme le dessus de la souris (voir l'image du même nom). Des informations plus globales notamment sur les régions voisines sont souvent nécessaires pour obtenir une décision plus fiable sur la présence d'une discontinuité. Il faudrait alors une coopération avec un détecteur de régions.

5.4.2 Textures

Les échantillons 1, 2, 3, 4, 5, 6 et 7 représentant les difficultés de type texture correspondent, respectivement, aux échantillons 1 et 2 de l'image du muscle, aux 1 et 2 de l'image de la souris et aux 1, 2 et 3 de l'image de la femme. Les échantillons sont tous marqués sur les images présentées en annexe 2.

Nous nous intéressons ici à la réponse des détecteurs en présence de textures simples, où les variations locales sont relativement importantes sans former de motif particulier. La texture ne peut être assimilée à un bruit simple car elle est souvent localisée dans l'image et ses variations sont d'origine sémantique et donc difficiles à modéliser. En revanche, le bruit peut raisonnablement être assimilé à une texture, c'est pourquoi nous regroupons ces deux problèmes dans cette même section. Nous proposons de nombreux exemples de texture. Le problème majeur est ici de ne rien détecter sans pour autant perdre en qualité de détection sur les discontinuités réelles. En particulier, les résultats de la robustesse à la texture et au bruit sont fortement corrélés avec les résultats sur les configurations de type 1 et de type 4 (faibles gradients et petits contours).

- TEXTURE, ECHANTILLON 1: image des cellules, numéro 1

Position du pixel en haut à gauche : 70 104

Largeur de la fenêtre : 15

Hauteur de la fenêtre : 20

```
170 186 194 194 193 190 184 177 194 199 211 198 196 199 174
196 194 189 180 176 198 204 180 191 194 195 192 191 191 175
174 178 189 194 168 160 172 175 183 188 181 185 173 184 181
183 177 175 198 198 191 196 199 197 185 185 193 178 179 181
157 179 176 191 206 211 204 210 196 196 194 186 204 200 194
180 176 178 184 191 188 190 197 201 195 191 195 211 206 205
177 161 153 166 185 187 188 185 186 203 208 196 198 198 210
197 176 185 192 185 188 191 197 198 209 198 193 192 181 189
195 173 173 182 204 206 198 206 206 211 214 209 204 181 180
198 190 170 192 188 179 189 196 191 208 211 205 198 203 208
189 190 201 195 193 200 186 171 166 190 198 188 182 190 202
185 192 189 172 180 198 195 162 166 190 189 179 190 200 194
196 197 182 178 192 193 186 177 189 206 215 214 196 194 182
172 181 185 189 184 191 198 199 205 199 209 201 204 190 184
153 160 179 190 186 183 193 199 212 189 186 170 182 201 194
180 188 188 185 180 175 177 182 202 200 181 193 204 202 196
191 181 170 174 180 179 197 193 178 192 190 183 195 187 196
189 197 189 179 184 185 183 180 193 204 196 193 186 186 200
191 170 174 179 183 184 180 183 200 195 192 203 202 197 185
182 162 173 172 192 206 204 196 200 205 187 166 182 198 192
```

Commentaires :

Notons la présence de fortes variations locales dues à la texture, mais aucune ne ressort par rapport aux autres, rien ne doit donc être détecté.

Il existe des zones de texture dans la plupart des images présentées (cheveux de la femme, plateau en bois dans l'image de la souris, coque du bateau, ciel, etc...) mais les textures les plus difficiles à traiter sont incontestablement celles de l'image du muscle. Nous rappelons ici que nous distinguons deux types de texture :

- 1) Texture simple : aucune discontinuité n'est visible, donc rien ne doit être détecté. Exemple : cellules claires de l'image du muscle.
- 2) Texture complexe : des éléments de petite taille sont visibles, il faut donc détecter leurs frontières ou les assimiler à des traits.

5.5 Evaluation de notre détecteur

5.5.1 Evaluation des coefficients du gradient relatif

L'utilisateur ayant la possibilité de modifier certains paramètres, nous proposons d'évaluer dans un premier temps notre détecteur en changeant certaines valeurs. Pour cela, nous proposons les choix suivants :

Nous rappelons la formule du gradient relatif :

- Gradient Relatif = $\frac{\text{gradient} - a * \text{Ecartmin} - b}{\text{Ecartmax} - \text{Ecartmin}}$
- Type marche : a = 2.92 b = 6.04
 - Type trait : a = 2.63 b = 3.96

- CHOIX 1 : Calcul du gradient relatif avec valeurs des paramètres laissées par défaut.
- CHOIX 2 : Calcul du gradient relatif avec augmentation de 10% de la valeur de "a" pour les deux types marche et trait.
- CHOIX 3 : Calcul du gradient relatif avec diminution de 10% de la valeur de "a" pour les deux types marche et trait.
- CHOIX 4 : Détection des contours uniquement de type "marche". Pour cela, nous prenons les valeurs par défaut pour le type marche et nous prenons a = 0 et b = 255 pour le type trait.
- CHOIX 5 : Détection des contours uniquement de type "trait". Pour cela, nous prenons les valeurs par défaut pour le type trait et nous prenons a = 0 et b = 255 pour le type marche.

RESULTATS :

Nous présentons en annexe 2 les résultats des 5 versions pour l'image du muscle.

EVALUATION :

- La première observation des résultats obtenus en conservant les paramètres par défaut montre que notre détecteur n'est sensible qu'aux discontinuités visibles ou à la limite de la visibilité. L'image du muscle comporte un grand nombre de difficultés avec beaucoup de contours ou de faux-contours de visibilité incertaine. Les résultats obtenus étant globalement satisfaisants sur cette image, nous avons donc déjà une validation partielle de notre démarche.

- En ce qui concerne la variation du coefficient "a", une modification de 10% de sa valeur ne modifie pas beaucoup les résultats. La raison de cette stabilité s'explique d'une part, par le fait qu'il existe généralement dans une image donnée peu de contours de visibilité difficile, et d'autre part par la force de discrimination du gradient relatif. En effet, en augmentant ou en diminuant de 10% la valeur de "a", nous restons encore dans la zone de visibilité ambiguë. Même si l'incertitude sur la position des points expérimentaux est importante, la droite calculée par la technique des moindres carrés fournit déjà une solution permettant d'obtenir des résultats de bonne qualité.

- La carte des contours de type marche est assez complète. Toutefois, pour chaque contour de type trait, nous devrions avoir deux contours de type marche situés de part et d'autre de celui-ci. Or, ce n'est pas toujours le cas. Lorsque seulement le bord de la transition est de type marche, le masque qui calcule le gradient fournit une valeur inférieure à la pente réelle, ce qui peut hypothéquer la poursuite du contour. D'autre part, si la forme de la transition n'est pas vraiment de type marche ou de type trait, le contour peut s'interrompre brusquement et être détecté en plusieurs petits contours de type différent.

La somme des deux cartes de contours fournit finalement une carte générale assez complète, tout en préservant la robustesse aux faux contours.

5.5.2 Comparaison avec le détecteur de Deriche

Nous proposons de comparer notre détecteur à celui de Deriche [Deri 87]. Après lissage et calcul du gradient selon cette technique, nous effectuons un seuillage par

hystérésis avec un seuil haut noté sh et un seuil bas noté sb . Nous fixons le paramètre α à 1. Nous proposons d'évaluer les choix suivants :

- CHOIX 1 : Détecteur de Deriche. $sh = 15$, $sb = 8$.
- CHOIX 2 : Détecteur de Deriche. $sh = 25$, $sb = 13$.
- CHOIX 3 : Détecteur de Deriche. $sh = 30$, $sb = 15$.
- CHOIX 4 : Détecteur Salotti. Technique d'hystérésis appliquée au gradient de type marche. Le gradient relatif n'est pas calculé, les informations sur les régions voisines ne sont donc pas prises en compte. Le principe de la croissance de contour est le même : étant donné un germe dont le gradient est supérieur à un seuil haut, d'autres pixels sont annexés tant que leur gradient est supérieur à un seuil bas. Les contours de type trait ne sont pas traités pour faciliter la comparaison avec le détecteur de Deriche.
Nous proposons les seuils : $sh = 30$, $sb = 20$.
- CHOIX 5 : Détecteur Salotti. Nous choisissons les paramètres par défaut pour les contours de type marche. Les contours de type trait ne sont pas traités ($a = 0$ et $b = 255$).

RESULTATS :

Nous proposons en annexe 2 les résultats des cinq versions pour l'image du muscle.

EVALUATION

- Malgré différents choix de seuils, le détecteur de Deriche ne parvient pas à résoudre simultanément le problème de la robustesse à la texture et la détection des contours de faibles gradients. Pourtant, les contours existant entre les cellules noires ont une visibilité évidente. Le contraste n'est pas excellent sur le papier, mais l'étude des niveaux de gris ne laisse aucune ambiguïté quant à l'existence de ces discontinuités. De même, certains contours détectés dans les cellules claires n'ont aucun support réel dans l'image. Les erreurs du détecteur de Deriche sur cette image sont donc assez importantes.

- En revanche, les quelques erreurs de notre détecteur sont mineures car la plupart correspondent à des contours dont la visibilité est limitée. Ces mêmes contours pourraient d'ailleurs sans doute être détectés en ajustant les valeurs de "a" et "b" dans la formule du gradient relatif.

- Notons que pour cette image, de nombreux détecteurs donnent des résultats du même ordre que celui de Deriche avec le même type d'erreur. Si nous nous référons à

l'étude réalisée dans le rapport GRECO [GREC 91], les résultats de notre détecteur nous semblent meilleurs.

- Si nous comparons les résultats du quatrième choix (notre détecteur avec seuillage par hystérésis du gradient) avec ceux de Deriche, nous remarquons qu'ils sont du même ordre. En effet, comme pour le détecteur de Deriche, quelques faux-contours ont été détectés dans la texture et des contours avec faibles gradients ont été oubliés. Cela montre que la façon de calculer le gradient n'est pas le problème majeur. Alors que de nombreuses études ont été faites au niveau du calcul du gradient et de la régularisation, c'est sans doute la prise en compte d'informations sur les régions voisines qui permet une meilleure discrimination des contours.

5.5.3 Evaluation globale

Nous présentons en annexe 2 les résultats de notre détecteur pour les six images tests dont nous avons tiré des échantillons. Tous les paramètres ont été choisis par défaut.

Les résultats de notre détecteur sont de bonne qualité pour les six images présentées. Il est en fait remarquable de constater que pour chaque contour détecté, il existe une transition visible ou à la limite de la visibilité, et que réciproquement, les contours oubliés ont pratiquement tous une visibilité difficile.

- Pour l'image de la femme, il existe de nombreux contours correspondant à une zone du visage, mais localement, il existe presque systématiquement une transition visible, ce qui valide la discontinuité. Il semble toutefois que notre détecteur soit tout de même un peu trop sensible à certains dégradés. (Notons que cette image ne présente d'intérêt que parce que de nombreux contours difficiles sont présents, mais il est bien évident que les contours situés sur le visage sont difficilement exploitables.)

- Pour l'image de la souris, le traitement du dégradé s'est avéré efficace. Les petits contours de la texture du bois ont souvent été correctement détectés, sans prolongement excessif, ce qui constitue pourtant une difficulté pour d'autres détecteurs. La détection des contours de faibles gradients est également satisfaisante.

- Pour l'image du bateau, les cordages ont été correctement détectés en tant que "traits" malgré la texture et les gradients souvent faibles et irréguliers. Les résultats sont globalement satisfaisants.

- Pour l'image IRM, là aussi, la robustesse à la texture et au bruit est conservée et la plupart des contours à dynamique complexe ont été relativement bien détectés.

- Pour l'image de l'objet polyédrique, le problème essentiel est la localisation des contours au niveau des jonctions. Bien qu'imparfaite, cette localisation est à peu près correcte.

Les résultats montrent donc que nos heuristiques exprimant la visibilité des contours sont pertinents et que les valeurs de gradient et d'ecartmin permettent une discrimination efficace des contours.

Nous présentons finalement tableau 1 une étude comparative qualitative entre notre détecteur avec paramètres par défaut et notre détecteur avec seuillage par hystérésis du gradient. Nous avons examiné les 7 types de difficulté définies au 5.1. Pour réaliser cette comparaison, nous avons expérimenté les deux versions sur les 6 images tests, présentées en annexe 2, en prêtant attention aux résultats obtenus pour chaque échantillon numéroté.

	Détecteur Salotti avec seuillage par hystérésis	Détecteur Salotti, paramètres par défaut
Gradients faibles	Insuffisant	Satisfaisant
Gradients complexes	Passable	Passable
Petits contours	Insuffisant	Satisfaisant
Contours proches	Passable	Passable
Dynamique complexe	Passable	Passable
Texture	Insuffisant	Satisfaisant
dégradé	Insuffisant	Satisfaisant
BILAN	Passable	Satisfaisant

Tableau 1 : évaluation finale pour chaque type de difficulté (notes entre "insuffisant", "passable" ou "satisfaisant")

La conclusion de cette évaluation est finalement que notre heuristique du gradient relatif est efficace et qu'elle permet de se rapprocher de la meilleure détection théorique des contours visibles d'une image.

5.6 Les temps de calculs

Il est très difficile de donner des résultats précis sur les temps d'exécution de notre algorithme, car le nombre de calculs par pixel dépend du contenu de l'image et nous n'avons pas optimisé le code que nous avons écrit en C++. Nous donnons néanmoins des exemples obtenus sur un Sun Sparc Station 2 pour la version actuelle.

Pour le calcul et le typage du gradient, le temps est presque indépendant du contenu de l'image. En effet, il faut 6 secondes en moyenne pour des images 256x256, mais comme nous ne typons pas le gradient qui a une valeur inférieure à 4, le nombre d'opérations n'est pas constant et le temps de calcul varie de plus ou moins 1 seconde.

Pour le reste de l'algorithme, il faut entre 6 et 30 secondes pour détecter tous les contours, toujours pour des images 256x256 (affichage simultané compris). Il faut multiplier environ par 4 pour des images de taille 512x512 (de même pour les calculs de gradients). Les temps de calcul sont en fait pratiquement proportionnels au nombre de germes puisqu'ils correspondent à peu près à tous les pixels à partir duquel va être calculé le gradient relatif. Les opérations les plus coûteuses sont en effet le calcul d'écart-type et de moyenne dans les fenêtres 3x3 ainsi que la formule du gradient relatif.

Les images avec textures conduisant à un plus grand nombre de germes, elles nécessitent plus de temps que les autres. Si aucune texture n'est présente, la détection est proportionnelle au nombre de contours qui existent dans l'image.

6 Conclusion sur la détection des contours

6.1 Conclusion sur l'évaluation

En premier lieu, nous voudrions insister sur la qualité des résultats déjà obtenus et ce de façon automatique en conservant les mêmes paramètres et seuils pour toutes les images. Si nous comparons nos résultats à ceux des détecteurs présentés dans le rapport GRECO [GREC 91], notre détecteur est selon nous plus performant, en particulier pour l'image du muscle où les différences sont assez significatives, même avec l'approche Markovienne qui manque de nombreux contours et qui localise mal les frontières complexes. De plus, nous détectons également les traits, ce qui améliore la description tout en augmentant la difficulté.

En ce qui concerne la méthodologie, nous pensons avoir montré son potentiel et ses qualités tout au long de l'évaluation. Nous avons effectué une analyse en profondeur des difficultés et nous avons souvent proposé des solutions algorithmiques originales. Il faut sans doute souligner que l'heuristique du gradient relatif a pu être suggérée grâce à notre démarche expérimentale. De manière générale, l'approche que nous proposons nous semble d'autant plus intéressante que nous ne connaissons pas vraiment les difficultés de la détection des contours. En analysant les valeurs numériques, en essayant différentes versions et en confrontant les résultats avec notre perception visuelle, nous sommes amenés à comprendre les problèmes et à nous forger une véritable expertise, qui peut alors être avantageusement exploitée pour proposer une détection de contours à l'aide d'heuristiques, ce que nous avons fait.

6.2 Perspectives

- Il est tout à fait possible d'améliorer encore ce détecteur en poursuivant notre démarche expérimentale. Nous proposons diverses voies de recherche qui pourraient conduire à des résultats intéressants :

- La forme n'est pas exploitée lors du choix du pixel suivant de la discontinuité. Or il y a au moins deux cas qui pourraient avantageusement en bénéficier. Le premier est illustré par la détection du contour elliptique de la souris (échantillons 8 9 et 10). Arrivé à une jonction, notre détecteur choisit avant tout de poursuivre vers le plus fort gradient. Or dans ce cas précis, la texture présente parfois de plus fortes variations, ce qui entraîne une détection autour de l'élément de texture qui s'arrête assez vite. En exploitant des informations sur la courbure très régulière du contour elliptique de la souris, le choix

pourrait être différent et le contour pourrait être détecté en une seule fois. Des informations sur la forme pourraient également être exploitées pour mieux situer les jonctions entre segments ou même pour aider la poursuite des discontinuités avec dynamique complexe comme le contour du visage de la femme.

- En présence de fort bruit impulsionnel, notre détecteur est en sous-détection. La première solution est de baisser les valeurs de "a" et "b" jusqu'à la détection désirée. Une autre solution est de tenir compte différemment des informations sur les régions voisines. Pour cela, il faut identifier la situation, ce qui pourrait se faire par une analyse statistique des répartitions juste avant ou en même temps que se calcule le type du gradient dans un voisinage 5x5. Un type différent pourrait être attribué si les variations locales sont disproportionnées par rapport à la valeur du gradient. Ensuite, pour calculer le gradient relatif, il est possible de tenir compte d'un voisinage plus important qu'une fenêtre 3x3 ou d'éliminer les valeurs extrêmes, ce qui est finalement assez proche d'un lissage adaptatif.

- Pour positionner correctement les jonctions, il est possible d'ajouter des masques adaptés à ces configurations particulières. D'autres masques diagonaux pourraient également être ajoutés pour augmenter la précision de l'orientation du gradient. Nous avons déjà commencé une étude en ce sens, mais la problématique est complexe et nous n'avons pas eu le temps d'effectuer une analyse suffisante pour proposer des modifications intéressantes.

- Un mot sur la parallélisation de notre algorithme. Nous n'avons pas étudié en détail l'aspect parallèle de notre détection des contours et cela n'a pas influencé nos choix algorithmiques. Toutefois, le calcul des gradients, le typage et la sélection des germes sont aisément parallélisables, avec de plus une répartition possible du calcul de chaque masque sur un processeur différent, ce qui irait sans doute plus vite qu'un calcul classique de gradient. Ensuite, le chaînage est difficilement parallélisable, mais il en est de même pour tous les détecteurs. En revanche, nous pouvons calculer les écart-types et moyennes des fenêtres 3x3 une fois pour toutes en début de programme et les stocker dans des tableaux. Certaines valeurs ne nous sont pas utiles, mais le calcul se faisant en parallèle, il n'y a pas de perte de temps. Nous pouvons alors continuer et calculer le gradient relatif pour tous les pixels de façon parallèle. Il ne resterait plus ensuite qu'à effectuer un chaînage comme celui que nous proposons et la détection des contours pourrait sans doute se faire en temps réel.

Il faut noter que dans la version actuelle, le gradient relatif ne varie pas en fonction du nombre de pixels de la discontinuité, de sa courbure ou du gradient moyen ce qui

permet un calcul grâce à des informations uniquement locales et donc la parallélisation de l'algorithme.

Cette perspective quelque que peu inattendue place ainsi notre détecteur parmi les meilleurs tant au niveau de la qualité des résultats qu'au niveau de la rapidité potentielle. Ajoutons à cela que nous n'effectuons aucun lissage préalable sur l'image, ce qui prend nécessairement du temps, même en parallèle, et que nous détectons de plus les contours de type trait.

6.3 Exploitation des contours

Il serait dommage de conclure la détection des contours sans envisager leur exploitation. De plus, les contours sont des indices visuels primaires et nous avons abordé au chapitre 1 le problème de l'émergence d'indices visuels de plus haut niveau. Il est donc intéressant d'examiner notre apport véritable.

Précisons avant tout quelles sont les informations disponibles en sortie de notre détecteur. Nous avons bien sûr la carte des contours mais nous proposons aussi un fichier contenant la liste des discontinuités :

- Pour chaque pixel-contour, nous donnons les coordonnées, le gradient ainsi que les moyennes et écart-types locaux des régions voisines (les fenêtres 3x3).
- Pour chaque chaîne de pixels, nous donnons le type du contour, le gradient moyen, le nombre de pixels et bien sûr la liste ordonnée des pixel-contours qui s'y rattachent.
- Enfin, nous donnons au début du fichier le nombre de discontinuités qui ont été détectées.

Les informations disponibles à la sortie de notre détecteur sont donc très riches, avec notamment des informations sur les régions voisines, ce qui est rarement proposé.

Nous ne rentrerons pas dans le détail de tout ce qui a été fait pour exploiter les contours. Nous nous y sommes intéressés de près, mais nous ne pouvons développer ici toute la littérature qui s'y rattache (voir par exemple les livres de Ballard et Brown [BaBr 82] ou de Marr [Marr 82]). Nous en donnons donc un bref résumé. En règle générale, il y a deux étapes :

- La première étape consiste à choisir une représentation plus sophistiquée comme par exemple des segments ou des courbes du second degré. Cette étape est donc une modélisation plus abstraite des discontinuités.
- La deuxième étape consiste à associer les segments ou les courbes en fonction de leurs propriétés pour former des groupes d'indices ou des indices plus élaborés. Des courbes

peuvent être associées en fonction de leur symétrie, de leur proximité ou en fonction d'un modèle connu de forme à obtenir. La théorie Gestaltiste a suscité de nombreux travaux dans ce domaine [RoPa 91].

En ce qui concerne l'exploitation de nos discontinuités, nous offrons donc des possibilités supplémentaires d'association avec la prise en compte des propriétés des régions voisines.

Nous ne présentons aucun algorithme de modélisation ni d'association particulier, mais nous proposons en perspectives des travaux que nous voudrions développer ultérieurement :

- Les formes simples telles que les cercles ou les ellipses, même grossières, sont des indices visuels qui semblent émerger facilement, comme en témoigne la carte des contours de l'image du muscle. Bien qu'aucune symétrie ne soit véritablement présente, il doit être possible de regrouper ensemble les contours qui forment des figures simples et de construire ainsi des indices visuels de plus haut niveau. Pour augmenter les contraintes et éviter des associations inutiles, les informations sur les régions voisines peuvent être avantageusement exploitées.

- La modélisation des discontinuités est une étape cruciale dont dépend beaucoup les étapes suivantes. Il nous semble que les modèles proposés ne sont pas toujours objectifs dans le sens où ils ne permettent pas une exploitation facile et une émergence simple des indices de plus haut niveau. En particulier, la proximité, la position relative et la ressemblance des formes sont difficiles à exprimer à partir de modèles tels que les B-splines. Peut-être faudrait-il passer par des indices visuels plus explicites tels qu'une famille de courbes modèles de courbure spécifique. Un arc de cercle est d'ailleurs une figure simple et il devrait donc à ce titre pouvoir émerger facilement de nos représentations.

De manière générale, la démarche que nous avons adoptée mérite sans doute d'être appliquée également au problème de construction d'indices visuels de plus haut niveau. Bien que de nombreux travaux aient déjà été réalisés dans ce domaine, il semble qu'y ait encore beaucoup à faire pour parvenir à une description satisfaisante.

CHAPITRE 3

La segmentation descriptive

1. Introduction

La segmentation en régions est un des problèmes les plus difficiles de la vision par ordinateur. Elle consiste à décrire l'image à l'aide de régions pertinentes regroupant tous les pixels respectant une même propriété. De nombreux travaux sont fondés sur une segmentation idéale de l'image, où chaque région correspond à une partie bien précise d'un objet. Le problème est alors ramené à une mise en correspondance directe entre les régions et les objets de la scène. La segmentation est donc une étape très importante, pratiquement incontournable pour de nombreuses applications.

Les premiers travaux sur la segmentation datent d'une trentaine d'années. Les espoirs étaient grands de réaliser un système de vision performant en s'aidant des techniques de segmentation. Cependant, après une période d'évolution rapide, les progrès ont été de plus en plus lents et les "vieilles" techniques ont bien peu à envier des méthodes actuelles. Il suffit de lire les excellents "surveys" de Zucker [Zuck 76] et de Riseman et Arbib [RiAr 77] pour se rendre compte que les problèmes sont toujours les mêmes et qu'ils n'ont pas été résolus. Après tant d'années de travail, les chercheurs butent toujours sur le problème de la segmentation.

Nous pouvons classer les méthodes de segmentation régions en deux grandes catégories (voir notre rapport de recherche [Salo 92]) :

- Celles qui exploitent des connaissances sur les objets pour aider la segmentation. Voir par exemples les approches de Tenenbaum et Barrow [TeBa 77].
- Celles qui ne les exploitent pas.

Dans le cadre de cette thèse, nous considérons uniquement la segmentation descriptive, c'est à dire que les régions fournies par l'algorithme ne correspondent pas obligatoirement aux objets de la scène, mais plutôt à des ensembles cohérents de pixels fournissant les premiers indices visuels. La segmentation idéale serait alors obtenue après exploitation de ces indices (plus éventuellement d'autres), retour dans l'image et reconnaissance complète des objets de la scène. Nous nous restreignons donc à l'étude des méthodes de segmentation régions n'exploitant aucune connaissance sémantique ou *a priori*.

Une segmentation descriptive ne peut être évaluée par rapport à la correspondance idéale avec les objets de la scène. Nous insistons sur ce fait, car beaucoup de méthodes de segmentation n'exploitent pas les connaissances sémantiques. Elles devraient donc être classées dans la catégorie des segmentations descriptives et être évaluées par rapport à la qualité de la description. Or ce n'est que rarement le cas et il y a souvent confusion entre segmentation descriptive et système complet de vision. Nous approfondissons ce problème dans la partie discussion de ce chapitre.

Dans un premier temps, nous abordons brièvement les techniques les plus connues pour obtenir des régions, puis nous nous focalisons sur la coopération régions-contours qui permet de cumuler à la fois les indices visuels contours et régions. Comme au chapitre 2, nous essayons de faire le point sur la méthodologie, sur l'orientation de la recherche dans ce domaine, ainsi que sur les grands problèmes irrésolus de la segmentation descriptive à travers les grands principes énoncés au premier chapitre.

Dans un deuxième temps, nous présentons notre technique de coopération incrémentale, fondée sur l'aide simultanée d'un processus d'agrégation de points adaptatifs et du processus de détection des contours présenté au chapitre précédent. Nous terminons enfin en présentant nos résultats et en les commentant.

2. Etat de l'art

2.1 Les méthodes simples de segmentation

2.1.1 Exploitation de l'histogramme

Il existe des méthodes de segmentation simples pour obtenir des résultats très performants dans les applications où les objets ont une couleur uniforme et bien distincte des autres objets et du fond. Ces méthodes exploitent l'histogramme des niveaux de gris. En effet, lorsque les objets se détachent du fond, l'histogramme comporte plusieurs blocs (appelés modes) correspondant à chaque objet. Le principe de base est de détecter les vallées entre chaque mode pour obtenir les seuils sur les niveaux de gris qui segmentent l'image en régions de façon satisfaisante.

Il existe de très nombreuses variantes que nous classons en deux grandes catégories décrites brièvement ci-dessous.

- La première catégorie regroupe toutes les techniques qui calculent des seuils globaux appliqués à toute l'image. La classification de Fisher [Fish 58] détermine le où les seuils qui minimisent la somme des variances de niveaux de gris de chaque classe, le nombre de modes étant demandé à l'utilisateur. Pour améliorer la forme de l'histogramme et faciliter le choix du seuil, certaines techniques sélectionnent les pixels qui vont être pris en compte dans le calcul de l'histogramme. Il existe deux approches possibles, celles qui tentent de ne pas tenir compte des pixels situés à la frontière entre les objets et le fond, et celles au contraire qui tentent de ne prendre en compte que ces pixels, où une partie d'entre eux. Weszka et *al* proposent ainsi de sélectionner les pixels qui ont une forte valeur de Laplacien [WeNR 74]. Ceci entraîne normalement un histogramme bimodal équilibré car la valeur du Laplacien est forte juste au début et à la fin d'une forte transition mais pas au milieu d'une région homogène, ni au milieu de la transition. Le seuil choisi est donc au creux de l'histogramme. Katz propose de sélectionner les pixels avec forte valeur de gradient et de choisir la valeur de gris moyenne de ces pixels [HaSh 85].

- La deuxième catégorie regroupe les techniques qui calculent des histogrammes locaux. Nous en citons deux. Chow et Kaneko proposent de calculer la variance et l'histogramme des valeurs de gris pour un ensemble de fenêtres de l'image. Si la variance dépasse un certain seuil et si l'histogramme est plutôt bimodal qu'unimodal, la fenêtre est partagée en fonction du seuil choisi dans la vallée de celui-ci. Pour les pixels n'appartenant à aucune fenêtre, la classification est déterminée par interpolation des seuils trouvés dans les fenêtres voisines [HaSh 85]. Ohlander et al [OhPR 78] proposent une division récursive de l'image. Etant donné une région connexe, l'histogramme est calculé

pour chaque couleur. Un lissage de l'histogramme est effectué, puis le meilleur pic est sélectionné. Ce pic est censé déterminer la moyenne des valeurs d'une région pour une couleur donnée. Deux solutions se présentent. Si les limites de cette région apparaissent clairement, c'est à dire si des vallées importantes sont trouvées de part et d'autre de ce pic, alors ces vallées correspondent aux seuils choisis pour identifier tous les pixels de cette région. Sinon, la région est correctement segmentée, il ne faut plus y toucher. A chaque fois qu'une décomposition a lieu, les régions résultantes sont traitées pour que les trous soient éliminés, puis les composantes connexes des régions formées sont réexaminées par le même principe de façon récursive.

Pour plus de détails, voir les surveys de Weszka [Wesz 78], Fu et Mui [FuMu 80], Haralick et Shapiro [HaSh 85], Sahoo et *al* [SSWC 88] ainsi que l'étude comparative de Lee et Chung [LeCh 90].

Nous terminons cette étude par une remarque concernant la gestion des informations. Les techniques qui viennent d'être présentées exploitent exclusivement les informations de niveau de gris sans tenir compte de la position des pixels (à quelques exceptions près). Ce sont donc les principes 1 et 2 (cumul et complémentarité des informations) qui marquent les faiblesses de ces approches, qui restent toutefois intéressantes de par leur simplicité et leur efficacité pour des applications simples.

2.1.2 Croissance de régions

Nous regroupons dans cette section toutes les techniques d'agrégation de points, de fusion et de division des régions. Là encore, il existe de très nombreuses méthodes. Nous en résumons les grands principes en détaillant quelques approches.

Une des premières idées pour regrouper les pixels a été de comparer leur niveau de gris et de lier tous ceux dont la différence est inférieure à un seuil. Pour améliorer la robustesse, il est préférable de prendre en compte une partie du voisinage de chaque pixel. L'inconvénient de cette technique est son caractère uniquement local. Après l'étape de chaînage, la connexité entraîne parfois la présence dans la même région de pixels ayant des propriétés très différentes [HaSh 85].

Les techniques de fusion et de division des régions permettent la prise en compte d'informations plus globales. Le principe est de partitionner progressivement l'image en régions, selon des critères plus ou moins complexes faisant intervenir par exemple la différence de moyenne des deux régions, leurs variances, leur nombre de pixels et le nombre de pixels adjacents. Nous présentons trois techniques en détail, le "split and

merge" de Horowitz et Pavlidis, la croissance hiérarchique de régions de Monga ainsi que la pyramide adaptative inspirée des travaux de Jolion, Montanvert, Meer et Rosenfeld.

- Horowitz et Pavlidis proposent un algorithme récursif de division de l'image par un parcours arborescent (quadtree), suivi par une technique de fusion de régions [HoPa 76]. En premier lieu, à partir d'une région dont la dimension est une puissance de 2, (si l'image ne possède pas cette propriété, il suffit de prendre la puissance de 2 immédiatement supérieure), la région est découpée en 4 sous-régions carrées (dont la dimension est encore une puissance de 2), et ainsi de suite récursivement jusqu'au niveau du pixel. Ensuite, à chaque fois que 4 sous-régions satisfont un critère d'homogénéité, elles sont regroupées à un niveau supérieur du quadtree. La fusion continue tant qu'il est possible de le faire, c'est à dire tant que le critère d'homogénéité est satisfait. Lorsqu'il n'est plus possible de fusionner, il reste encore une étape pour examiner les régions adjacentes qui n'étaient pas au même niveau dans le quadtree et les fusionner si elles satisfont le critère d'homogénéité.

Cette méthode originale laisse une empreinte spécifique dans la carte des régions avec des frontières marquées par de nombreux traits verticaux et horizontaux. De plus, aucun critère de forme n'est exploité pour contraindre la fusion, ce qui occasionne parfois des régions particulièrement découpées.

Notons que le critère d'homogénéité joue un rôle très important dans la fusion des régions mais que celui-ci n'a pas fait l'objet d'une évaluation, ce qui est dommage. L'intérêt de cette technique est avant tout la rapidité et la taille mémoire relativement faible qui est nécessaire pour segmenter l'image.

- Gagalowicz et Monga proposent d'effectuer une technique de fusion de régions en plusieurs étapes, dans lesquelles les critères sont de plus en plus tolérants [GaMo 85].

- 1) Une première carte des régions est obtenue par split and merge en utilisant un critère très strict qui impose que la différence entre deux niveaux de gris d'une même région soit inférieure à un seuil. Ce seuil est choisi très faible, de l'ordre de 5.

- 2) Tous les couples de régions adjacentes sont classés par ordre de différence de moyenne de niveau de gris. Tous les couples qui ont une différence de moyenne inférieure à un seuil sont fusionnées. Chaque fusion entraîne une mise à jour dans le graphe d'adjacence et le classement des couples.

- 3) Le même principe de fusion est appliqué en classant les couples de régions adjacentes en fonction de la valeur de la variance résultant de l'union des deux régions. Un seuil de variance maximale acceptable est utilisé.

- 4) Tous les couples de régions qui ont un gradient moyen à la frontière inférieur à un seuil sont fusionnées.

5) Les petites régions de taille inférieures à un seuil sont rattachées aux plus grandes.

La qualité première de cette approche est de s'attaquer au problème épineux des critères de fusion des régions. Les critères proposés sont de plus en plus tolérants ce qui permet théoriquement une segmentation de bonne qualité. Toutefois, il est dommage que l'évaluation de leur algorithme soit restée presque exclusivement au niveau théorique. Il aurait été intéressant de voir le résultat lorsque l'ordre d'application des critères est différent (pour mieux situer les avantages des critères hiérarchiques), lorsque d'autres seuils sont choisis et surtout, il n'est pas précisé dans quels cas des erreurs ou des résultats moins satisfaisants sont obtenus. En particulier, il nous semble que la succession des critères est trop stricte et que la fusion idéale devrait exploiter tous les paramètres de façon simultanée. Par exemple, l'étape numéro 4 doit être appliquée avec plus de précautions. Nous avons vu lors de la détection des discontinuités qu'un gradient de 4 pouvait être pertinent, à condition que les régions soient très homogènes, c'est à dire que leur variance soit très faible. Or, la variance n'est pas exploitée dans cette étape. Un autre problème lié à la fusion des régions est l'annexion des pixels proches d'une discontinuité. Ces pixels ont en effet une valeur un peu plus forte (ou plus faible) que la moyenne de la région et contribuent néanmoins au calcul de moyenne et de variance. Cet effet de bord n'est pas pris en compte dans cet algorithme. Enfin, que se passerait-il en présence de régions texturées ? Bien que notre vue soit très critique, cet algorithme reste néanmoins très intéressant car il pose clairement le problème du critère de fusion à appliquer. Nos critiques sont de plus très générales et resteraient probablement valables pour de nombreux algorithmes de segmentation.

- L'approche "pyramide adaptative" s'inspire de la stratégie de fusion optimale des régions (voir [JoMo 89] et [MoMR 91]). Etant donné un partitionnement initial de l'image en régions (un pixel est considéré comme une région), il est possible de classer toutes les fusions possibles selon un critère d'homogénéité et d'effectuer à chaque fois la meilleure fusion. Après chaque fusion, il y a mise à jour du graphe d'adjacence et du classement des fusions potentielles. Cette stratégie est optimale par rapport au critère de fusion des régions. Pour éviter la structure très séquentielle de cette méthode, l'approche pyramidale consiste à effectuer les meilleures fusions locales de façon simultanée, puis à recommencer avec les régions formées au niveau suivant et ainsi de suite jusqu'à stabilité de la carte des régions. L'optimalité est perdue mais les résultats restent du même ordre avec une parallélisation possible de l'algorithme. La difficulté est de maintenir la cohérence du graphe d'adjacence et de choisir une seule fusion locale en conservant l'aspect parallèle de la méthode. En mémorisant la qualité du voisinage des niveaux précédents (les nœuds pères transmettent les informations aux nœuds fils), la fusion peut exploiter des informations locales et globales.

Après avoir résolu la technique de fusion, les difficultés sont ici d'ordre relationnels dans le sens où il s'agit de trouver quels sont les critères qui doivent être appliqués pour décider de la fusion de deux régions. Comme nous l'avons déjà souligné pour les précédentes méthodes de croissance de régions, le critère idéal de fusion dépend théoriquement de très nombreuses informations comme, entre autres, la moyenne et l'écart-type des niveaux de gris mais aussi la forme des régions et la qualité de la frontière. Or là encore, comme pour le split and merge de Pavlidis, la technique de segmentation est bien étudiée mais elle n'a pas été bâtie pour faciliter la décision particulièrement délicate de fusion des régions. Même si le critère de fusion s'avère finalement assez sophistiqué, en présence de dégradés, de textures, de discontinuités avec faibles gradients ou de formes régulières, les informations disponibles ne sont pas assez objectives et les régions trouvées ne sont pas toujours pertinentes. Toutefois, si la critique est facile, cet algorithme figure tout de même parmi les plus performants, ce qui souligne à la fois que le problème de la segmentation est loin d'être résolu et qu'il est d'une grande complexité.

2.2 La coopération régions-contours

2.2.1 Les approches classiques

La coopération entre un processus de croissance de régions et un détecteur de contours a suscité un regain d'intérêt ces dernières années. Les premiers travaux importants dans ce domaine sont pourtant relativement vieux, avec par exemple l'approche de Brice et Fennema en 1970 [BrFe 70]. Bien qu'ils ne présentent pas leur méthode de segmentation comme une coopération, ils exploitent l'information du gradient à la frontière ainsi que la longueur de celle-ci pour déterminer la fusion des régions à partir de deux heuristiques qu'ils appellent "phagocyte" et "weakness". Plus récemment, Nazif et Levine ont proposé un système expert de segmentation coopérative. Nous consacrons à leur approche la section suivante (Cf 2.2.2).

La coopération a souvent consisté à effectuer d'abord une détection des contours, puis à exploiter la carte des contours pour contraindre le processus de segmentation en régions, c'est par exemple le cas de l'approche de Wrobel et Monga [WrMo 87]. Melkemi et Chassery proposent également une coopération avec détection préliminaire des contours, suivie d'une technique de croissance de régions fondée sur les polygones de Voronoï [MeCh 92]. La "segmentation coopérative" de Bonnin est fondée sur une comparaison des résultats d'un processus de fusion et d'un détecteur de contours, avec plusieurs étapes de raffinements [Bonn 91]. Même si les contours sont remis en question,

ceux-ci sont calculés également au début de la coopération, sans l'aide des informations sur les régions. De manière générale, la coopération se fait essentiellement des contours vers les régions. Ceci est probablement dû à l'incapacité des détecteurs de contours à exploiter les informations globales véhiculées par les régions. Le "gradient relatif", proposé au chapitre précédent est, à notre connaissance, le seul critère de validation d'un contour qui combine à la fois informations sur le gradient et moyennes et écart-types de niveaux de gris des régions voisines.

Une autre coopération a été proposée par Pavlidis et Liow [PaLi 90]. Après une première étape de pré-segmentation par l'algorithme de "split and merge" de Pavlidis, une procédure de fusion est appliquée en tenant compte des gradients à la frontière. La segmentation régions n'est ensuite plus remise en cause, mais les frontières entre celles-ci sont approximatives. Une technique de contours déformables permet alors de repositionner correctement les contours, en exploitant les informations de direction du gradient et de courbure. Cette approche est intéressante car il est très rare que les algorithmes de segmentation utilisent des informations sur la forme pour localiser la frontière des régions.

La coopération entre plusieurs techniques de segmentation a donc souvent consisté en une succession d'étapes de segmentation régions ou contours. La coopération se situe surtout au niveau de la confrontation des résultats, mais pas au niveau de la prise de décision de chaque processus, notamment en ce qui concerne la validation d'un contour. Les problèmes majeurs sont, selon nous, l'intégration des informations complémentaires et le contrôle de la coopération qui a pour charge la gestion et l'enchaînement des étapes. Ces problèmes étant liés à la gestion des informations, nous allons nous attacher tout particulièrement à leur étude.

2.2.2 La système expert de Nazif et Levine

Nazif et Levine ont été les premiers à présenter un système expert pour effectuer la segmentation d'images [NaLe 84]. Cette approche s'inscrit un peu à contre courant de la tendance de l'époque où l'explicitation des connaissances était réservé à l'étape suivante d'interprétation. Néanmoins, nous allons nous intéresser de près à leurs travaux car le système qu'ils proposent constitue une coopération régions-contours très originale.

En premier lieu, ils obtiennent la carte des contours par un seuillage simple des gradients, ainsi qu'une carte des premières régions homogènes avec de nombreux attributs dont la liste des régions adjacentes (peu de précisions sont apportées sur ces étapes préliminaires). A ces trois cartes s'ajoutent des informations sur les zones d'intérêt

de l'image, correspondant à des ensembles de régions de même type, et à des ensembles de contours. Ensuite, une zone d'intérêt est choisie, où chaque élément va être étudié.

En ce qui concerne les régions, il existe des règles dédiées dont la conclusion est la fusion ou la division de deux régions et dont la prémisse est constituée d'attributs sur les régions et sur les contours. La fusion de deux régions dépend, selon la règle, de la grosseur des deux régions, de la différence de moyenne dans chaque couleur, de la variance des deux régions dans chaque couleur, du nombre de pixels à la frontière, de la présence de pixel-contours à la frontière ainsi que de la forme de l'histogramme.

Les règles dédiées aux contours consistent à prolonger, à joindre ou à supprimer ceux-ci. Les prémisses prennent en compte la présence d'autres contours dans le voisinage, leur alignement, le gradient moyen sur le contour, la longueur de celui-ci ainsi que l'unicité des régions adjacentes.

La comparaison des attributs à valeur numérique s'effectue qualitativement avec les qualificatifs suivant : "very high", "high", "medium", "low" and "very low". Bien que les résultats soient relativement sensibles à ces qualificatifs réducteurs, les règles qu'ils proposent sont complexes et expriment des critères relativement sophistiqués. S'ils perdent en précision sur un attribut, ils gagnent en revanche en adaptation grâce à l'exploitation de plusieurs attributs dans la même prémisse.

En ce qui concerne les zones d'intérêt, elles sont classées en deux grandes catégories, textures ou régions homogènes. Une texture est identifiée grâce à un ensemble de petites régions de forte variance. Les zones d'intérêt sont mises à jour dynamiquement et permettent une focalisation sur un ensemble restreint de régions et de contours. Lorsque toutes les zones d'intérêt ont été traitées, les régions et les contours résiduels sont passés en revue pour terminer la segmentation.

Nous donnons ci-dessous trois règles de leur système expert :

```
IF    (1) The REGION SIZE is NOT LOW
      (2) The REGION AVERAGE GRADIENT is HIGH
      (3) The REGION HISTOGRAM is BIMODAL
THEN (1) SPLIT the REGION according to the HISTOGRAM;
```

```
IF    (1) The REGION SIZE is NOT LOW
      (2) REGION is BISECTED BY LINE
      (3) The LINE LENGTH is NOT LOW
      (4) The LINE AVERAGE GRADIENT is HIGH
THEN (1) SPLIT the REGION at LINES
```

IF (1) The LINE END point is OPEN
 (2) The LINE GRADIENT is NOT VERY LOW
 (3) The DISTANCE to the LINE IN FRONT is NOT VERY HIGH
 (4) The two LINES have the SAME REGION to the LEFT
 (5) The two LINES have the SAME REGION to the RIGHT
 THEN (1) JOIN the LINES by FORWARD expansion;

La coopération entre les régions et les contours apparaît ici dans les deux sens, pour diviser correctement les régions traversées par un contour, mais aussi pour aider le prolongement des contours. Toutefois, là encore, le processus de détection des contours appliqué en début de programme s'est fait sans l'aide du processus de croissance de région. Les lignes n'étant remises en cause que sur des critères de longueur par rapport aux régions, le résultat final devrait dépendre fortement de la détection des contours. Il est dommage que peu d'indications aient été données sur ce détecteur.

De façon générale, le système expert qu'ils présentent est difficile à juger, car nous ne sommes pas capables d'évaluer l'expertise contenue dans chaque règle, ce qui constitue un paradoxe pour un système expert. Il aurait sans doute fallu qu'ils expliquent la méthodologie de création des règles, ainsi que les moyens mis en œuvre pour les évaluer et les modifier. Nous retiendrons que ce système expert est une tentative d'explicitation des connaissances en segmentation d'images, avec émergence des informations dans les règles de production.

2.2.3 L'approche multi-agents

L'approche multi-agents pour effectuer une coopération régions-contours semble naturelle (chaque agent est un petit système expert autonome lié aux autres agents). Baujard et Garbay proposent un système de vision complet pour l'identification de cellules biologiques, fondé sur la coopération entre le détecteur de contours de Deriche [Deri 87] et la technique de classification appliquée aux régions de Fisher [Fish 58], précédée d'un filtrage de l'image [BaGa 93]. Un agent est dédié à chacun des processus. Après détection des régions et des contours, les résultats sont analysés et comparés dans un agent spécifique. Lorsqu'il y a compatibilité entre la frontière des régions et les contours, le résultat est validé et envoyé à un agent qui tente de classer la région à partir de la forme de celle-ci. Si la compatibilité n'est pas vérifiée, les régions sont découpées ou fusionnées.

Cette approche est particulièrement intéressante car elle pose clairement le problème de la gestion de la coopération. Il ne s'agit pas ici de juger cette méthode sur la qualité des

résultats mais sur la manière d'aborder le problème. Il semble en effet qu'il y ait de grandes difficultés à faire coopérer deux méthodes qui ont peu de points communs et qui ne sont pas spécialement préparées pour cette coopération. Si le principe du cumul des informations est important, le principe de complémentarité des informations doit également être pris en compte pour que l'efficacité soit réelle. Ainsi, à partir d'un détecteur tel que celui de Deriche (en fait à partir de la plupart des détecteurs de contours), il est difficile de trouver un processus de détection des régions qui soit complémentaire. La distribution des rôles dans un univers multi-agents n'est donc pas remise en cause, mais elle n'est pas très efficace si les processus coopérants sont trop indépendants. Cette remarque reste valable pour les coopérations non fondées sur l'approche multi-agents.

Il est difficile de cerner les avantages théoriques de l'approche multi-agents pour la coopération région-contours. Si chaque agent est distribué sur un processeur différent, il y a bien sûr un gain de temps. En ce qui concerne la qualité de la segmentation, il nous semble que les informations sur les régions et les contours doivent être exploitées simultanément au niveau local. Peut-être faudrait-il donc distribuer spatialement les agents. De manière générale, l'approche multi-agents et les algorithmes parallèles méritent une plus grande attention car la façon d'aborder les problèmes incite à une réflexion profonde sur la gestion des informations et l'enchaînement des étapes.

2.3 Discussion

2.3.1 Les petites phrases

Comme pour le chapitre consacré à la détection des contours, nous effectuons ici un petit tour d'horizon de la façon d'aborder le problème de la segmentation, en rappelant quelques petites phrases de plusieurs chercheurs.

Haralick et Shapiro, d'abord en discutant d'une technique de segmentation, puis dans leur conclusion de leur "survey" sur la segmentation [HaSh 85] :

<< ... although the segmentation procedure did exactly as it should have done, the results are not what we desired. Segmentation into homogeneous regions is not necessarily a good solution to a segmentation problem. >>

<< Not discussed as part of segmentation is the fact that it might be appropriate for some segments to be merged not on the basis of the gray tone distributions, but on the basis of the object sections which they represent. >>

Tenenbaum et Barrow, en présentant leur technique de segmentation [TeBa 77] :

<< The conventional approach of performing segmentation followed serially by interpretation is frequently ineffective because semantically important pictorial features can be visually indistinct while prominent features can be artifacts in the context of the current task. Consider, for example, partitioning an image of a black and white cat on a black and white rug. >>

Riseman et Arbib, discutant de l'utilisation des informations sémantiques pour aider la segmentation [RiAr 77] :

<< There has been some debate over the degree to which semantic information should be employed in the partitioning of an image. ... It seems to us that there is a large number of nonsemantic patterns of sensory visual data which can allow effective, although not perfect, initial segmentation without recourse to semantics. ... For these reasons, we view the problem of image understanding as one of performing initial segmentation via general procedures, feeding this low-level output to a high-level system, and then allowing feedback loops so that the interpretation processes can influence refined segmentation. ... From this point of view, the segmentation processes provide a compact description of the location and characteristics of visually distinct areas of the image. >>

Jain et Binford, en soulignant l'importance de la segmentation [DIAL 91] :

<< After a very active early period in segmentation research, the problems in segmentation have usually been ignored. ... There have been very few efforts to use three-dimensional models of objects and knowledge of image formation in refining segmentation. >>

Aloimonos et Rosenfeld, en répondant à Jain et Binford [DIAL 91] :

<< There is no doubt that segmentation is an important problem in general recovery. Unfortunately, it is an ill-defined problem: there is no straightforward way to define subdivisions of an image into meaningful parts, unless these parts have simple definitions in terms of pixel intensity. >>

Lee et Chung en comparant cinq méthodes de seuillages fondées sur l'histogramme des niveaux de gris [LeCh 90].

<< ... all algorithms break down for a certain ratio of population of objects and background pixels in an image, which in practice may arise quite frequently. Also, our experiments show that the thresholding algorithms discussed in this paper are data-dependent. >>

Fu et Mui, en commentant les techniques de segmentation en régions [FuMu 80] :

<< Up to the present moment, there is not published information on successfully applying region extraction techniques to a practical cytology pattern recognition system. >>

Pavlidis, en expliquant pourquoi les progrès sont lents en vision [Pavl 92] :

<< Segmentation algorithms usually assume a smoothly varying brightness for each object which is true for simple images but untrue in general. >>

2.3.2 Les informations sémantiques

Comme nous l'avons exprimé dans l'introduction de ce chapitre, il faut différencier deux types d'approches pour la segmentation : celles qui exploitent des informations sémantiques et celles qui n'en tiennent pas compte. Il existe en fait deux buts possibles pour les processus de segmentation. Soit il s'agit d'obtenir une partition de l'image en respectant au mieux les frontières des objets, soit il s'agit d'obtenir une description de l'image selon ses principales caractéristiques, comme première étape d'un processus de vision. Dans le premier cas, la segmentation est pratiquement un système complet de vision, certaines étapes intermédiaires étant "court-circuitées". Des informations sémantiques dépendantes de l'application peuvent être exploitées pour contraindre la fusion des régions et aboutir à des résultats satisfaisants. Dans le deuxième cas, après l'étape de segmentation, il est nécessaire de procéder à des associations d'indices visuels, puis de tenter une interprétation de la scène. Si l'interprétation est réussie, il est toujours possible de proposer une nouvelle segmentation dont les régions correspondent aux objets.

Tenenbaum et Barrow prennent l'exemple pertinent d'un chat noir et blanc sur un tapis noir et blanc pour montrer que la segmentation sans information sémantique peut conduire à un échec. De même, une technique d'extraction de régions à partir de l'image noir et blanc du dalmatien (image bien connue, présentée par exemple dans [SpLe 87]) fournirait des régions qui ne correspondraient pas à la forme du chien. Toutefois, en considérant que la segmentation a un but uniquement descriptif, il doit être possible d'associer les contours de ces régions, par exemple selon les principes de la théorie Gestaltiste [RoPa 91], puis d'interpréter une partie de ces contours comme étant une tête ou une patte et finalement de reconnaître le dalmatien. En particulier, un des grands principes de cette théorie est que les indices visuels, pris isolément, ne signifient rien, alors que l'association de certains indices devient une information pertinente. Par conséquent, la décomposition 1) détection d'indices visuels primaires 2) interprétation, critiquée à juste titre par Tenenbaum et Barrow, n'est pas la plus appropriée. Nous proposons plutôt :

- 1) détection des indices visuels primaires
- 2) association d'indices visuels
- 3) interprétation

Notons que nous abordons de nouveau ces problèmes dans la section 2.3.3 qui suit celle-ci.

Comme Riseman et Arbib (Cf 2.3.1), nous considérons donc qu'il est possible de procéder à une première étape de segmentation **descriptive**, et que la segmentation idéale est alors trouvée après identification des objets de la scène. Notre approche qui consiste à détecter les indices visuels primaires sans nous préoccuper de leur sens sémantique est donc justifiée.

2.3.3 Le but de la segmentation

Comme en témoignent "les petites phrases ", de nombreux chercheurs considèrent que le problème de la segmentation est très complexe.

La segmentation est souvent la première étape d'un système de vision. Si les résultats de celle-ci sont insuffisants, les performances globales du système ne sont pas satisfaisantes. Or, justement, dans le cas général, les résultats de la segmentation sont difficilement exploitables.

Mais quels sont les reproches qui sont faits aux méthodes de segmentation ?

Le reproche fondamental est que les régions ne correspondent pas à des objets ou des parties bien définies d'objets de la scène. Ce reproche est paradoxal puisque en théorie, comme nous l'avons déjà souligné, aucune information sémantique ne doit être utilisée dans un processus de segmentation descriptive. Nous ferons deux remarques importantes pour mieux comprendre ce qui est reproché à la segmentation :

- REPROCHE DE L'INSUFFISANCE SEMANTIQUE :

Le but d'un processus de segmentation descriptive ne peut être d'effectuer une mise en correspondance optimale entre objets et régions. Un processus de description doit avoir pour but de relever les indices visuels essentiels existant dans l'image. En l'occurrence, ces indices sont des informations sur des régions de l'image. Si ensuite il n'est pas possible d'effectuer une interprétation de la scène, c'est que ces indices sont insuffisants et qu'il manque des informations **complémentaires** pour faire **émerger** les caractéristiques des objets à reconnaître. Il ne s'agit donc pas de blâmer la méthode de segmentation, mais de rajouter les outils nécessaires à la détection d'indices plus pertinents, et de préparer l'exploitation de ces indices. Par exemple, après détection des contours et des régions, il est possible de procéder à des **associations** de courbes, selon

des critères de ressemblance, de proximité, de symétrie par rapport aux régions voisines etc.... Bien entendu, la méthode de reconnaissance des objets doit être modifiée et adaptée en conséquence.

• **REPROCHE DE LA SIMPLICITE DE LA MODELISATION :**

Comme l'a souligné entre autres Pavlidis, les hypothèses sur lesquelles sont fondés de nombreux algorithmes sont trop réductrices. Nous présentons ci-dessous la définition classique de la segmentation afin d'en cerner les faiblesses.

Définition classique de la segmentation :

Etant donné un prédicat P pouvant s'appliquer aux attributs des régions, une segmentation SEG est un partitionnement de l'image en N régions tel que :

- 1) $SEG = \{ R_1, R_2, \dots, R_n \}$
- 2) R_i est une région connexe quel que soit i .
- 3) Chaque pixel appartient à une région unique.
- 4) $P(R_i) = \text{vrai}$ quel que soit R_i
- 5) $P(R_i \cup R_j) = \text{faux}$ quelles que soient les régions connexes R_i et R_j

En rajoutant de plus un critère C à minimiser, comme par exemple la somme des variances des régions pondérées par le nombre de celles-ci, le problème de la segmentation admet alors une solution optimale en fonction de P et C . A partir de ces hypothèses de travail, des algorithmes optimaux de segmentation ont été proposés. Il est toutefois paradoxal qu'un algorithme optimal, comme tous les autres algorithmes de segmentation, fournisse des résultats difficilement exploitables. Les hypothèses de travail sont en fait fortement réductrices à plusieurs niveaux :

- **Le prédicat choisi est souvent un critère d'homogénéité trop simple.**

Par exemple, l'écart entre la valeur minimale et la valeur maximale de niveau de gris d'une région doit être inférieur à un seuil. S'il existe une région légèrement texturée, il est probable qu'il n'existe aucun seuil satisfaisant qui préserverait à la fois la totalité de la région texturée et les frontières entre deux zones homogènes d'intensité voisine. Le choix du prédicat P qui conditionne la fusion des régions est selon nous un problème très important d'une grande complexité. Alors que de nombreuses recherches ont été faites pour trouver la meilleure technique de croissance de régions, le choix du critère de fusion, de division ou d'agrégation est souvent passé au second plan (voir notre discussion de l'approche de Gagalowicz et Monga au 2.1.2). Pourtant, le but de la segmentation est de décrire l'image. Il s'agit donc d'identifier les zones de texture, les

dégradés, les "tâches", les régions homogènes bien sûr, et les régions aux formes simples. Or, toutes ces régions respectent des propriétés très différentes, difficiles à exprimer sous la forme d'un simple prédicat.

- De manière générale, il nous semble que **le problème de la segmentation a été trop simplifié et même sans doute dénaturé**. Nous ne voyons pas la nécessité d'imposer dans le cadre général de la segmentation que tous les pixels appartiennent à une région, et que les régions ne se recouvrent pas. Il serait même intéressant de proposer des groupes de régions, ou une décomposition hiérarchique de celles-ci. De la même façon que l'association d'indices visuels de type contour est importante, l'association de certaines régions a parfois un sens alors qu'aucune d'entre elles, prise isolément, ne présente de caractéristiques intéressantes.

LES DIFFICULTES DE LA SEGMENTATION :

Quelles sont plus précisément les difficultés de la segmentation ?

La première grande difficulté est relative aux problèmes locaux résultant de toutes les configurations particulières des pixels.

- Par exemple, les pixels près des discontinuités ont des niveaux de gris différents de la moyenne des régions voisines. Faut-il les prendre en compte dans la moyenne et la variance de celles-ci ?

- Ne faudrait-il pas traiter simultanément les textures simples (sans motif interne visible), les dégradés et les petites tâches (quelques pixels) qui sont différents types de régions ?

- Enfin, les problèmes de forme et de distance relative ont été selon nous largement sous-évalués, alors qu'ils font partie des critères les plus importants. En effet, comme nous l'avons souligné en définissant les indices visuels primaires au premier chapitre, l'existence de corrélations entre la position des pixels et leur niveau de gris doit nécessairement intervenir très tôt dans l'émergence des indices visuels.

A ces problèmes relatifs au contenu des images s'ajoutent les problèmes d'ordre algorithmique. Toutes les techniques de segmentation posent des problèmes de contrôle au niveau de l'organisation et l'enchaînement des étapes, ainsi que des problèmes de choix des structures de données. Ces problèmes sont liés à la gestion des informations, dont nous avons donné quelques principes au premier chapitre.

Tous ces problèmes sont bien sûr très complexes et rendent difficile la segmentation. S'ils ne sont pas correctement pris en compte, les régions trouvées par les algorithmes perdent automatiquement beaucoup d'objectivité, dans le sens où elles ne permettront pas

d'aider efficacement les étapes suivantes et notamment l'interprétation de la scène. Nous ne critiquons d'ailleurs pas tel ou tel algorithme mais plutôt le fait que les difficultés réelles n'aient pas été clairement établies, avec une évaluation souvent insuffisante. Jain et Binford illustrent leurs propos avec l'histoire de celui qui cherchait ses clés sous la lumière d'un lampadaire, loin de l'endroit sombre où il les avait réellement perdues [DIAL 91]. Voudraient-ils exprimer le fait qu'une inertie non négligeable tendrait à unifier le cadre général de la segmentation descriptive, avec simplification implicite du but à travers la modélisation classique qui en est faite, ce qui limiterait la critique à un contexte bien précis et conduirait à un manque de clairvoyance des difficultés réelles ?

Nous concluons donc que le but d'un détecteur de régions est de décrire l'image en identifiant les zones qui ont des propriétés intéressantes. Toute la difficulté est de trouver ce qui se cache derrière ces "propriétés intéressantes". Il s'agit sans doute d'informations sur les niveaux de gris, sur la position des pixels ou sur une corrélation entre les deux. A cela s'ajoute des informations sur les relations avec les autres régions, ce qu'il ne faut pas négliger car un ensemble d'indices visuels a parfois plus de sens que la somme de ses parties.

3. Une coopération région-contours intégrale

Nous présentons dans cette partie une nouvelle technique de coopération entre un détecteur de contours (celui du chapitre 2) et une méthode de croissance de régions ([SaGa 92] et [SaGa 93a]). Nous commençons par décrire les grandes lignes de la méthode de croissance de régions en expliquant nos choix, puis nous détaillons la structure de contrôle qui permet la coopération et l'aide entre les deux processus.

3.1 Agrégation de pixels adaptative

Comme pour la détection des contours, nous justifions d'abord notre approche. Nous décrivons ensuite notre algorithme et nous essayons de formaliser le critère d'agrégation des pixels. Nous terminons par une évaluation du détecteur de régions.

3.1.1 Spécifications générales

Quel est le but de notre détecteur de régions ?

- Nous désirons fournir une description pertinente de l'image avec des indices visuels primaires de type région. Pour obtenir une description riche, nous proposons de différencier plusieurs types de régions : le type homogène, le type texture (texture simple sans motif visible) et le type dégradé. Ce typage nous paraît essentiel pour deux raisons :
 - Lors du processus de détection, il est important de s'adapter à l'indice visuel présent. En typant les régions, nous pouvons ainsi sélectionner dynamiquement le critère de détection le plus approprié.
 - Lorsqu'il s'agira ensuite d'exploiter la carte des régions, ces types seront autant d'informations pertinentes qui peuvent faciliter la tâche des processus de plus haut niveau. En considérant les principes que nous avons énoncés au premier chapitre, nous pouvons dire que le typage permet de rendre les informations plus objectives.
- Nous ne désirons pas effectuer une segmentation de l'image, mais bien une détection des indices visuels de type région. Pour cette raison, nous n'imposons pas que chaque pixel appartienne à une région. Le fait que l'image ne soit pas complètement segmentée, ne nous paraît pas gênant, bien au contraire. En effet, il existe souvent des groupes de pixels qui ne vérifient aucune propriété pertinente. Quel intérêt y a-t-il à regrouper ces pixels en régions ? Quels critères utiliser pour les regrouper ? Peut-on en faire une

description intéressante ? De telles régions ne pourraient sans doute qu'embarrasser les processus de plus haut niveau. Le fait même qu'aucune région ne soit détectée à un certain endroit est de toutes façons une autre information qui peut être exploitée.

De plus, les pixels situés sur une discontinuité, ou même en bordure de celle-ci, ont souvent des caractéristiques locales qui ne sont plus celles de la région voisine. L'annexion de tels pixels entraînerait alors une variation arbitraire de la moyenne et de l'écart-type des niveaux de gris de la région, ce qui n'est pas souhaitable.

Pourquoi un processus d'agrégation plutôt qu'une technique de fusion des régions ?

- Nous avons choisi un processus d'agrégation pour plusieurs raisons. Avant tout, il nous semble que l'appartenance d'un pixel à une région nécessite la prise en compte d'informations sur la région et d'informations locales relatives au pixel. Or, dans un processus de fusion, puisque l'étude se fait au niveau des régions, on perd l'information locale. Il est possible d'étudier le gradient à la frontière pour améliorer la fiabilité mais la décision reste difficile à contrôler. D'un autre côté, il est vrai que le processus d'agrégation est parfois trop local et que l'expansion produit des formes parfois très découpées peu intéressantes. Nous opposons à ces critiques deux arguments :

- Premièrement, pour éviter d'être "trop local", il suffit de prendre en compte un voisinage assez important autour du pixel étudié. Nous pouvons par exemple calculer le gradient avec des masques 5x5 comme pour notre détecteur ainsi qu'un écart-type et une moyenne des niveaux de gris dans une fenêtre 3x3. Nous supposons ici aussi que le facteur d'échelle est le même pour toutes les images étudiées.

- Deuxièmement, l'avantage du processus d'agrégation réside dans les capacités de contrôle de chaque étape. Etant un processus incrémental, le choix du germe peut être imposé et le processus peut être interrompu pour demande d'informations complémentaires ce qui facilite la coopération avec un autre processus. Ce dernier point va être illustré par la technique de coopération que nous présentons dans la partie 3.2.

Notons enfin que nous nous inspirons des travaux de Chassery et Garbay qui ont développé une technique d'agrégation de pixels pour une application en cytologie [ChGa 84].

3.1.2 Description de l'algorithme

En considérant les spécifications générales faites à la section précédente, nous proposons les étapes suivantes : (voir figure 1)

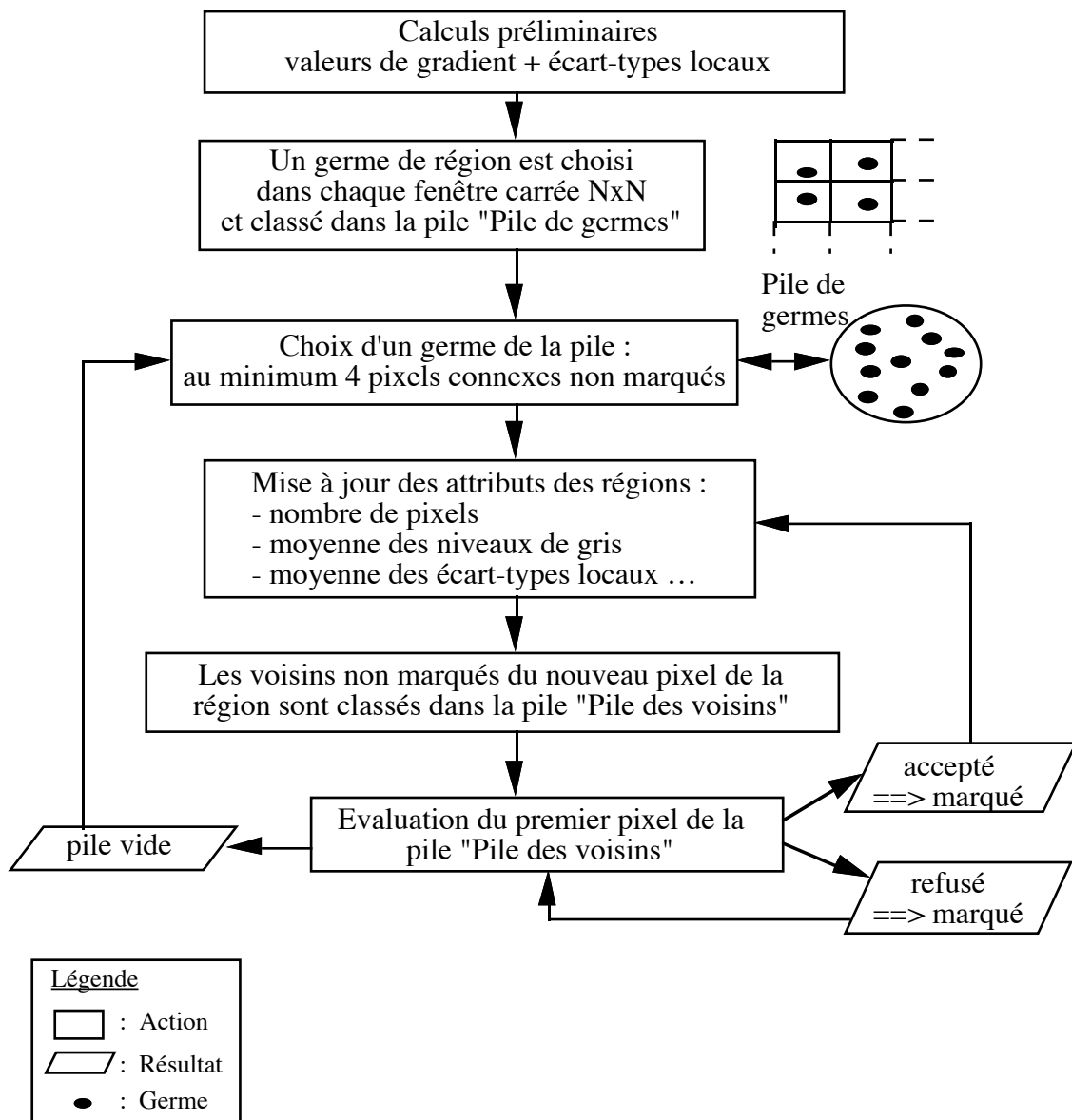


Figure 1 : Notre technique de croissance de régions par agrégation de pixels.

- Premièrement, pour tous les pixels de l'image, nous calculons le gradient comme pour notre détecteur de contours, et nous y ajoutons le calcul de l'écart-type et de la moyenne des niveaux de gris dans une fenêtre 3×3 .

- Deuxièmement, l'image est divisée en fenêtres $N \times N$ disjointes. Nous laissons le choix de N à l'utilisateur (nous proposons par défaut $N=10$). Dans chaque fenêtre, nous choisissons le meilleur germe possible de région en minimisant la valeur de l'écart-type local et nous le classons dans une pile.

Les deux étapes précédentes sont effectuées une seule fois au début de l'algorithme. En revanche, les étapes suivantes s'imbriquent dans un processus incrémental qui s'apparente beaucoup à celui de notre détecteur de contours.

- Troisièmement, nous choisissons un germe dans la pile en vérifiant qu'il existe bien 4 pixels non marqués, afin de commencer avec une région de taille 4 (ceci est nécessaire pour uniformiser un traitement présenté plus loin). Une région est créée.

- Quatrièmement, les attributs de la région sont mis à jour. Le type de la région est déterminé en fonction de la valeur de ses attributs. Nous proposons de caractériser les types de la façon suivante :

1) Une région en dégradé possède une valeur de gradient moyen non nulle et une orientation de celui-ci constante. La valeur du gradient ne doit pas être trop forte pour qu'il n'y ait pas confusion avec un contour.

2) Une région de type texture (texture simple sans motif visible) possède une moyenne d'écart-type local forte.

3) Une région homogène possède une moyenne d'écart-type local faible.

Nous introduisons également un quatrième type pour faciliter les débuts du processus d'agrégation :

4) Une région de petite taille a bien sûr peu de pixels. Les contraintes d'agrégation sont relâchés pour ces petites régions.

Nous proposons finalement la décomposition arbitraire suivante, qui pourrait être modifiée suivant l'application envisagée :

```

si (nombre de pixels < 10)
    alors type = "Petite région"
sinon si (
    et (4 ≤ gradient_moyen ≤ 20)
    et (80% des pixels ont même orientation de gradient à 45 degrés près)
    )
    alors type = "Dégradé"
sinon si (moyenne des écart-type locaux > 3)
    alors type = "Texture"
sinon
    alors type = "Homogène"

```

- Cinquièmement, les pixels voisins (en 8 voisinage) du nouveau pixel de la région sont examinés. S'ils ne sont pas marqués, c'est à dire s'ils n'appartiennent pas déjà à une région ou aux voisins d'une région, et s'ils ont au moins 2 voisins consécutifs en 8

voisinage qui appartiennent déjà à la région, ils sont placés dans une pile "Pile des voisins" et classés en fonction d'une fonction d'évaluation présentée plus loin. Les pixels de cette pile correspondent à la frontière extérieure de la région. La condition des "2 voisins consécutifs appartenant à la région", illustrée figure 2, permet de conserver une compacité locale de la région et évite les expansions à partir d'un seul pixel. Cette condition est à l'origine de la taille minimale imposée de 4 pixels pour commencer le processus d'agrégation.

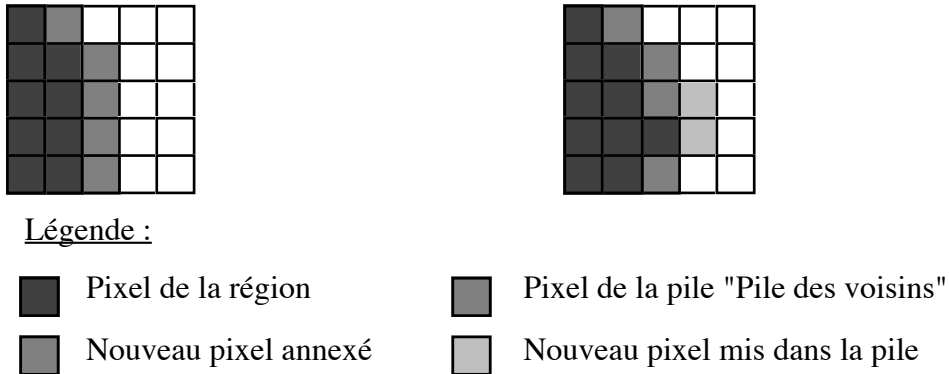


Figure 2 : Après annexion du pixel central, à gauche aucun nouveau pixel ne va être rajouté à la pile alors qu'à droite 2 vont l'être.

- Sixièmement, le premier pixel de la pile "Pile des voisins" est évalué selon un critère de similarité (voir section suivante le critère exact). S'il est accepté, il est agrégé et l'algorithme continue à l'étape 4. S'il est refusé, il est marqué et l'algorithme reprend à l'étape 6. Si la pile est vide, la région est terminée, retour à l'étape 3, tant qu'il reste des germes. Le typage de la région permet d'effectuer une évaluation adaptative du nouveau pixel. Par exemple si le type est texture, nous pouvons accepter un écart-type plus fort.

Notons que les pixels frontières de chaque région n'appartiennent à aucune région et qu'il existe de nombreux pixels non affectés. Comme nous l'avons déjà expliqué, notre partitionnement de l'image n'est généralement pas complet.

DESCRIPTION DE LA STRUCTURE DE DONNEES :

- Nous utilisons quatre tableaux d'octets de la taille de l'image pour stocker la valeur du gradient, sa direction et l'écart-type et la moyenne locale.
- Pour stocker les germes de région, nous avons une pile ordonnée où chaque élément est défini par les coordonnées x et y du pixel germe et une note qui est égale à l'écart-type local lié à ce pixel.

- Lors d'une croissance de régions, la "pile des voisins" stocke les pixels qui sont situés autour de la région. Les nouveaux pixels sont classés dynamiquement en fonction d'une fonction d'évaluation présentée dans les sections suivantes. Les pixels acceptés pour l'agrégation à la région et ceux qui sont refusés sont retirés dynamiquement de la pile.

- Une région est une classe C++ définie avec les attributs suivants :

- Type (petite, homogène, texture ou dégradé).
- Pile des voisins.
- Nombre de pixels.
- Moyenne de gris.
- Moyenne des écart-types locaux.
- Moyenne de la norme des gradients.
- Nombre de pixels ayant la même direction de gradient à 45 degrés près, pour chacune des 8 directions possibles.

3.1.3 Définition du critère de similarité

Pour décider de l'annexion d'un pixel à la région, il faut déterminer un critère de similarité et évaluer les attributs de ce pixel en fonction de celui-ci. Comme nous l'avons fait remarquer dans la partie 2 de ce chapitre, il s'agit là d'un problème extrêmement important qui n'a pas de solution simple. Monga propose par exemple une décomposition hiérarchique de ce critère, mais cela nous semble insuffisant. Il faudrait tenir compte de façon simultanée d'un grand nombre de paramètres pour adopter la meilleure décision ce qui est très difficile. Nous pouvons néanmoins essayer de caractériser approximativement le critère de similarité. Nous proposons intuitivement trois aspects :

(1) Il faut que les variations locales au niveau du pixel examiné soient inférieures ou du même ordre que les variations locales de tous les autres pixels déjà annexés à la région.

(2) Il faut également que le niveau de gris local soit du même ordre que celui de la moyenne des pixels de la région. Une grande différence entre les deux valeurs est acceptable si la région est bruitée ou texturée, alors que peu de différence est accordée si la région a un niveau de gris uniforme.

Notons que pour une région de type dégradé, ce critère n'a aucun intérêt, puisqu'il est normal que le niveau de gris croisse ou décroisse dans une direction donnée. Il faut en revanche vérifier la conservation des informations liées à la norme et à la direction du gradient.

(3) Il faut tenir compte de la forme générée par l'annexion du nouveau pixel. En particulier, les critères 1) et 2) doivent être moins stricts si la forme de la région devient simple après annexions de plusieurs pixels d'une même zone de l'image. Inversement, si la forme de la région est régulière et que l'annexion de nouveaux pixels brise cette régularité, alors il faut être plus sévère au niveau des critères 1) et 2).

En tenant compte seulement de ces trois aspects, nous rencontrons déjà de grandes difficultés. En effet, ceux-ci sont fortement corrélés et posent des problèmes algorithmiques complexes. En particulier, le troisième critère implique une gestion délicate des informations liées à la forme de la région, ce qui ne peut se résoudre facilement.

Pour simplifier les problèmes et constituer une première solution algorithmique, nous proposons de construire une heuristique qui tienne compte uniquement des deux premiers critères, mais de façon simultanée. L'étude d'une solution algorithmique permettant de caractériser le troisième critère pourra faire l'objet de recherches futures. La prise en compte des deux premiers critères de façon simultanée constitue une solution partielle, dont il est intéressant de connaître la validité. L'analyse des résultats expérimentaux liés à cette solution nous permettra de mieux comprendre les relations existant entre les informations manipulées, ce qui développera notre expertise de la détection des indices visuels de type région.

3.1.4 Spécifications des critères

Nous essayons ici de caractériser les deux critères cités section précédente.

- Pour exprimer le premier critère caractérisant la conservation des variations locales, nous proposons d'examiner des valeurs expérimentales. Pour chaque pixel, nous calculons le gradient de la même façon que pour notre détecteur de contours présenté au chapitre 2, ainsi que l'écart-type local des niveaux de gris dans une fenêtre 3x3 centré sur ce pixel. Ces deux valeurs nous renseignent sur les variations locales existant dans un voisinage centré sur le pixel étudié. Le gradient donne une information sur des variations corrélées à la position des pixels, tandis que l'écart-type donne une information sur le désordre local. Ces deux valeurs sont grossièrement complémentaires et de même dimension car normalisées par rapport au nombre de pixels. Pour caractériser la variation locale d'un pixel, nous proposons une solution simple qui consiste à choisir la valeur maximale entre le gradient et l'écart-type.

Etant donnée une région R de moyenne de variation locale M , pour qu'un pixel de cette région respecte le critère de similarité des variations locales, il ne faut pas qu'il paraisse singulier par rapport aux autres. Ainsi, en présence de bruit impulsionnel, les pixels dont la variation locale est singulièrement forte ne doivent pas être annexés à la région, ou du moins pas selon ce critère. Pour exprimer ce critère de similarité, il nous semble donc préférable d'évaluer une variation locale d'un pixel par rapport à leur moyenne sur une région, sans tenir compte de leur variance. En effet, si la variance des variations locales est forte, cela implique que des pixels de cette région sont singuliers et qu'il n'y a donc pas véritable similarité. Nous posons donc comme hypothèse que la distribution des variations locales d'une région respectant ce premier critère de similarité doit être une Gaussienne de faible largeur.

Etant donné un pixel P de variation locale V_p et une région R avec une moyenne des variations locales V_r , le problème est finalement de savoir quelle est la valeur limite de V_p en dessous de laquelle nous considérons que le critère de conservation des variations locales est satisfait.

Pour cela, nous proposons une démarche expérimentale. A partir d'un ensemble de régions sélectionnées dans des images de synthèse et des images réelles, nous établissons pour chacune le seuil V_p en fonction de V_r . Pour les images de synthèse, nous choisissons un bruit blanc Gaussien d'écart-type compris entre 0 et 10, échantillonné tous les 0.5. Pour chaque région, nous déterminons le seuil V_p visuellement en examinant les zones que nous jugeons critiques et en modifiant parfois dynamiquement les niveaux de gris pour mieux apprécier la pertinence de notre choix. Nous obtenons finalement le graphe présenté figure 5.

Pour les moyennes de variations locales supérieures à 7, aucune région d'image réelle n'ayant été trouvée, les points expérimentaux proviennent essentiellement d'images de synthèse bruitées. Signalons une caractéristique importante des images de synthèse : leurs points expérimentaux sont tous situés à droite des autres, pour une variation locale donnée. Il semble que cette particularité soit due à la distribution gaussienne des valeurs qui permet d'identifier immédiatement une variation singulière. Dans les images réelles, il arrive en effet que l'augmentation de la variation locale soit répartie dans une zone assez grande. Comment dans ce cas décider quelle est la variation limite acceptable ? Pour simplifier les problèmes, nous avons évité dans la mesure du possible de choisir des points expérimentaux dans de telles régions. Ces restrictions importantes sont provisoirement nécessaires pour pouvoir caractériser notre premier critère indépendamment des autres. Notons que notre jugement visuel est subjectif et approximatif, mais qu'il constitue néanmoins une des références les plus sûres.

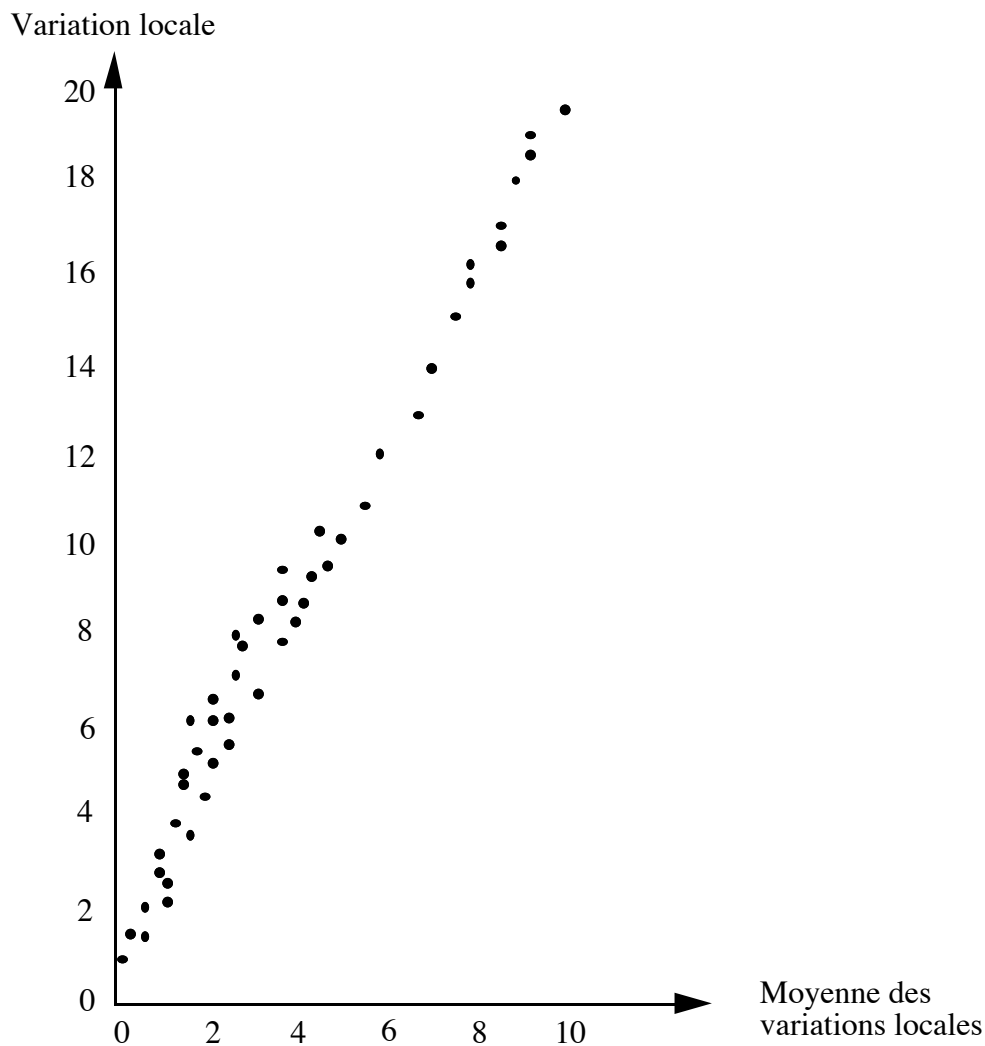


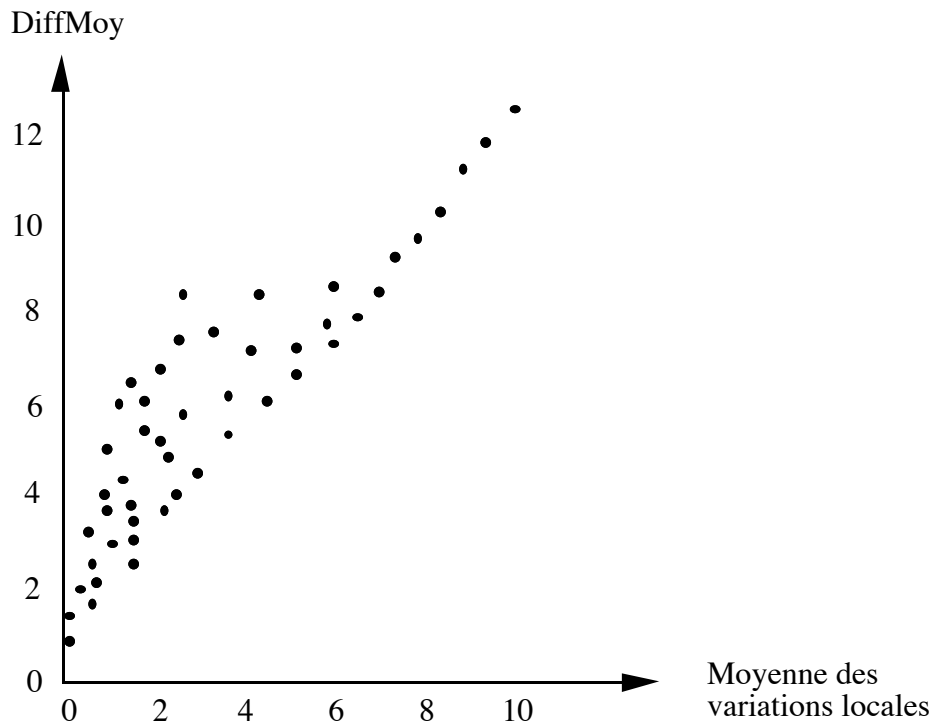
Figure 5 : Points expérimentaux pour le premier critère.

- Pour caractériser le deuxième critère, nous proposons une démarche similaire. Nous calculons la moyenne des niveaux de gris dans une fenêtre 3x3 centrée sur le pixel étudié, puis nous calculons la valeur absolue de la différence avec la moyenne des niveaux de gris de la région. Nous appelons cette différence DiffMoy.

Etant donné un pixel P et une région R de moyenne de variation locale V_r , le problème est alors de déterminer quelle est la valeur limite de $\text{DiffMoy}(P,R)$ en dessous de laquelle nous considérons que le critère de conservation du niveau de gris moyen est satisfait.

Pour cela, comme pour le premier critère, nous proposons de chercher les valeurs limites d'un ensemble de points expérimentaux. Pour un grand nombre de régions, nous calculons V_r et nous cherchons visuellement la valeur limite de DiffMoy. Le choix des régions n'est pas tout à fait le même que pour le premier critère. En effet, le traitement des régions présentant un faible dégradé local est délicat : localement il y a peu de différence

avec la moyenne des pixels voisins, mais par rapport à la région, la différence est forte relativement à V_r . Dans un tel cas, quelle réponse adopter ? Pour éviter ce problème, il est nécessaire d'éviter les régions qui ont un dégradé local. Nous présentons nos points expérimentaux figure 6, obtenus à partir d'images de synthèse et d'images réelles.



Nos résultats expérimentaux montrent que la valeur limite de DiffMoy est difficile à déterminer en fonction de la moyenne des variations locales. La raison essentielle est que malgré tous nos efforts, les régions examinées dans les images réelles comportent toutes plus ou moins des dégradés locaux. Visuellement, aucune différence ne semble significative par rapport aux pixels voisins mais si nous replaçons la configuration locale au cœur de la région, le contraste est évident. Comme pour le critère 1, en ce qui concerne les images de synthèse, la distribution Gaussienne des valeurs permet d'identifier facilement une configuration locale s'écartant sensiblement de la moyenne de la région.

Cette analyse expérimentale montre en fait toute la difficulté de la détection des régions. Il existe toujours des dégradés locaux, des écarts de texture et des configurations particulières qui ne peuvent être facilement traités. Ainsi, les paramètres que nous proposons pour décrire une région ne capturent pas toutes les informations importantes et ne sont donc pas assez objectifs. Outre le fait que la forme de la région devrait être prise

en compte, il faudrait également des informations complémentaires, par exemple sur la moyenne des niveaux de gris dans des zones internes à la région. Cela pose toutefois des problèmes algorithmiques difficiles et nous nous en tiendrons donc là dans le cadre de cette étude.

3.1.5 Formulation de l'heuristique

Grâce aux deux diagrammes expérimentaux des figures 5 et 6, nous pouvons maintenant exprimer nos deux critères. Etant données les incertitudes et la subjectivité de nos critères, nous proposons de les traduire simplement par une droite en utilisant la technique des moindres carrés appliquée aux points expérimentaux.

Nous rappelons que nous distinguons quatre types de région (Cf section 3.1.1) :

- 1) Les petites régions
- 2) Les régions en dégradé
- 3) Les régions homogènes
- 4) Les régions texturées

Pour annexer un pixel à une région en dégradé, nous vérifions uniquement la norme et le type du gradient par rapport à ceux de la région. Les critères 1 et 2 ne concernant pas ce type, nous proposons un critère simple d'agrégation :

Soit G et D la norme et la direction du gradient sur le pixel examiné, et soit D_{moyen} la direction moyenne de gradient des pixels de la région.

type = "Dégradé" :

si $(4 \leq G \leq 20)$
 et $(|D - D_{moyen}| < 45 \text{ degrés})$
 alors pixel annexé à la région
 sinon pixel refusé

En ce qui concerne les petites régions, elles peuvent être considérées comme homogènes ou texturées, ce qui ramène donc notre problème à ces deux derniers types.

La limite entre les deux types est donnée par un seuil arbitraire sur V_r , qui est, rappelons le, la moyenne des variations locales des pixels d'une région. En fixant ce seuil à 3 par défaut, nous obtenons donc :

Si $V_r < 3$, nous avons une région de type homogène.

Si $V_r \geq 3$, nous avons une région de type texture.

En appliquant la technique des moindres carrés aux deux critères et aux deux types de région, nous obtenons finalement les inéquations suivantes :

Etant donné les définitions suivantes,

R région traitée, P pixel à évaluer pour l'annexion à R

Variation_Moyenne : moyenne des variations locales de tous les pixels de R

Variation_Locale : variation des niveaux de gris dans un voisinage centré sur P

Diff_Moyenne : différence de niveaux de gris entre R et un voisinage centré sur P

• Type homogène :

Critère 1 : $Variation_locale < 2.3 * Variation_Moyenne + 0.5$

Critère 2 : $Diff_Moyenne < 2.7 * Variation_Moyenne + 1.1$

• Type texture :

Critère 1 : $Variation_locale < 1.9 * Variation_Moyenne + 0.5$

Critère 2 : $Diff_Moyenne < 1.2 * Variation_Moyenne + 0.3$

Les inéquations trouvées n'expriment pas directement la force du critère. Nous proposons donc de caractériser cette "force" par les formules suivantes :

• Type homogène :

Critère1 = $2.3 * Variation_Moyenne + 0.5 - Variation_Locale$

Critère2 = $2.7 * Variation_Moyenne + 1.1 - Diff_Moyenne$

• Type texture :

Critère1 = $1.9 * Variation_Moyenne + 0.5 - Variation_Locale$

Critère2 = $1.2 * Variation_Moyenne + 0.3 - Diff_Moyenne$

De cette manière, la qualité du pixel est d'autant plus grande que la valeur du critère est forte, et une qualité insuffisante est caractérisée par une valeur négative.

Il reste à trouver la meilleure combinaison des deux critères pour exprimer complètement notre heuristique. Nous proposons pour cela la formule générale suivante, indépendamment du type texture ou homogène de la région :

k étant un réel compris entre 0 et 1 et Seuil un réel quelconque,

Si $(k * Critère1 + (1-k) * Critère2) > Seuil$

Alors P annexé à R
 Sinon P appartient à la frontière extérieure de R

Cette heuristique est relativement simple. Pour $k = 0$, seul le critère 2 est pris en compte, et pour $k = 1$, c'est le critère 1. Pour déterminer les meilleurs paramètres k et Seuil, nous ne proposons pas de démarche expérimentale précise. Les imprécisions et les approximations qui ont été faites ne permettent pas un calcul pertinent de ces valeurs. Nous laissons donc à l'utilisateur le choix de ces paramètres, qu'il fixera en fonction des images à traiter et des résultats qu'il désire obtenir. Par défaut, nous proposons une valeur de k égale à 0.7 et une valeur de Seuil égale à 2. La valeur 0.7 privilégie la conservation de la variation locale, qui nous semble plus robuste et plus adaptée, notamment en présence de dégradés locaux. La valeur 2 de Seuil peut paraître paradoxale car des pixels ne satisfaisant aucun des deux critères (valeurs < 0) peuvent ainsi être annexés. Un tel choix se justifie par la présence quasi-systématique de dégradés locaux et de configurations particulières dans les images réelles. La valeur 2 peut entraîner un découpage insuffisant des régions, mais elle permet un relâchement des contraintes qui améliore dans la plupart des cas l'objectivité des résultats.

Notons que pour le type "petite région" (moins de 10 pixels), nous prenons un Seuil égal à 4 pour faciliter les débuts de la croissance de région et éviter les arrêts prématurés, dus à une configuration locale relativement homogène par rapport à son voisinage.

3.1.6 Evaluation de notre détecteur de régions

Nous examinons ici les heuristiques du processus de croissance de régions.

Nous proposons une évaluation en plusieurs parties. Pour commencer, nous examinons l'apport du type dégradé, puis nous nous intéressons aux formules exactes d'agrégation pour les deux autres types, et enfin nous effectuons une évaluation globale du détecteur.

- Pour évaluer l'apport du type dégradé, nous proposons simplement d'observer les résultats lorsque ce type de régions n'est pas traité.

CHOIX 0 : Version complète du processus de croissance de régions.

CHOIX 1 : Nous choisissons l'option qui ne traite pas le type "dégradé".

RESULTATS :

Nous présentons des résultats en annexe 3 avec l'image de la souris qui présente quelques régions en dégradé. Parce que le problème se situe uniquement au niveau de l'apport du dégradé, nous avons en fait appliqué notre technique de coopération régions-contours. Les résultats sont malgré tout identiques au niveau de la carte des régions qui nous intéresse plus particulièrement.

EVALUATION :

En conservant le type dégradé, la description gagne beaucoup en qualité. Le dessus de la souris est bien mieux décrit avec de telles régions qu'avec des régions de type texture. Notons qu'une grande région en dégradé régulier ne doit pas être détecté en tant que région de type texture. En effet, les informations disponibles dans ce type ne permettent pas de distinguer un dégradé d'une région véritablement texturée. La description serait alors ambiguë et l'indice visuel peu pertinent. De manière générale, les dégradés sont des indices visuels souvent caractéristiques des objets, ils méritent donc d'être détectés en tant que tels..

- Pour vérifier la caractérisation de nos deux critères, nous proposons d'évaluer notre détecteur sur une image de synthèse avec bruit Gaussien et bruit additif impulsionnel. Théoriquement, en choisissant une valeur de Seuil égale à 0 et une valeur de k égale à 0.5, nous devons obtenir toutes les régions correctement, car l'image de synthèse respecte les hypothèses de construction de nos deux critères : pas de dégradé local et pas d'augmentation répartie de la variation locale.

CHOIX : $k_T = k_H = 0.5$, $\text{Seuil}_H = \text{Seuil}_T = 0$ (T pour Texture et H pour Homogène)

RESULTATS :

Nous avons appliqué notre croissance de régions à une image de synthèse connue. Les résultats sont présentés en annexe 3.

EVALUATION :

Nous obtenons les résultats attendus. Toutes les régions ont été correctement détectées. Il existe de nombreux petits trous qui correspondent à la présence d'un bruit impulsionnel local. Comme nous l'avons déjà précisé, il est nécessaire de ne pas agréger de tels pixels, car ils n'ont pas une similarité suffisante et pourraient faire diverger les propriétés des régions. Il est toutefois certain que ces pixels mériteraient d'être annexés. Cependant, cela ne pourrait se faire qu'en tenant compte de leur position à l'intérieur de la

région, ce qui relève d'un autre critère que nous n'avons pas exploité. Ce point nous semble particulièrement important : si les informations prises en compte ne permettent pas de décider correctement de l'annexion d'un pixel, alors ce serait une erreur que d'effectuer tout de même cette annexion. Par analogie, c'est comme si nous arrivions à trouver la solution d'un problème avec une démonstration fautive. Il s'agit donc de bien comprendre les limites de nos critères et d'évaluer les résultats en connaissance de cause.

Notons que la région en dégradé de l'image de synthèse a été correctement détectée, ceci grâce à notre deuxième critère qui ne permet pas un écart des niveaux de gris "visible" entre la région et le pixel étudié.

- Nous proposons d'examiner l'influence des coefficients de nos fonctions heuristiques d'agrégation des pixels. Nous rappelons ces formules.

* type = "Texture"

si $(kT * \text{Critère1_Texture} + (1-kT) * \text{Critère2_Texture} > \text{SeuilT})$

alors pixel annexé

sinon pixel refusé

* type = "Homogène"

si $(kH * \text{Critère1_Homogène} + (1-kH) * \text{Critère2_Homogène} > \text{SeuilH})$

alors pixel annexé

sinon pixel refusé

Pour tous les choix présentés, c'est l'utilisateur qui ajuste les paramètres.

CHOIX 0 : $kT = kH = 0.7$ $\text{SeuilT} = \text{SeuilH} = 2$ (valeurs par défaut)

CHOIX 1 : $kT = kH = 0.7$ $\text{SeuilT} = \text{SeuilH} = 3$

CHOIX 2 : $kT = kH = 0.5$ $\text{SeuilT} = \text{SeuilH} = 2$

CHOIX 3 : $kT = kH = 1$ $\text{SeuilT} = \text{SeuilH} = 2$

CHOIX 4 : $kT = kH = 0$ $\text{SeuilT} = \text{SeuilH} = 2$

RESULTATS :

Nous présentons les résultats de la croissance de régions avec l'image du muscle en annexe 3.

EVALUATION :

Les résultats de la segmentation en régions varient beaucoup selon les valeurs de coefficient utilisées.

Entre les deux premiers choix (0 et 1), le relâchement du seuil permet de combler un peu mieux les trous présents dans les dégradés locaux. Cependant, il y a parfois des débordements et certaines régions englobent plusieurs cellules ce qui n'est pas le cas avec le seuil plus strict.

Pour les choix 2 à 4, les résultats montrent que le critère 1 de conservation de la variation locale est plus robuste que le critère 2 de conservation de la moyenne de la région. Les cellules de cette image comportent en effet de nombreux dégradés locaux qui rendent peu pertinent le critère 2. Toutefois, en ne prenant en compte que le critère 1, il y a de nombreux trous et quelques erreurs qui sont dus à une augmentation brusque des variations locales, un peu comme du bruit impulsionnel, mais réparti au niveau d'une petite zone. En prenant en compte le critère 2 à 30%, cela revient à relâcher le seuil pour les pixels restant à peu près au même niveau de gris que celui de la région, et au contraire à élever le seuil pour ceux qui s'en écartent. De cette façon, nous obtenons un compromis permettant une meilleure adaptation du critère global d'agrégation.

- Pour effectuer une évaluation globale, nous proposons des types de difficultés liés au partitionnement en régions. Comme pour les contours, nous distinguons les difficultés locales des difficultés globales.

- Dans la première catégorie, nous trouvons le problème de "l'attribution d'un pixel" en zone texturée ou en zone uniforme. Il y a le problème inverse des pixels à ne pas attribuer car correspondant à une discontinuité dans la texture ou entre deux régions d'intensité à peine différente. Ces différents types de problèmes sont locaux.

- Dans la deuxième catégorie, un des problèmes essentiels est la "conservation de la propriété" pertinente d'une région. En effet, il ne faut pas que l'annexion de pixels en bordure de région entraîne une variation des propriétés de celle-ci et que le processus d'agrégation diverge. Il ne faut pas non plus qu'une région de texture assez homogène soit découpée en un grand nombre de petites régions. Enfin, et c'est un point que nous voudrions souligner, les formes simples conduisent à une description plus facilement exploitable de la scène (les formes simples sont plus objectives). Il s'agit donc dans la

mesure du possible de proposer des régions avec contours réguliers et des formes simples à décrire.

RESULTATS :

Nous présentons tableau 1 une évaluation qualitative des résultats en fonction du type de difficulté rencontré dans les images.

Régions homogènes	Satisfaisant
Textures régulières	Satisfaisant
Dégradés réguliers	Satisfaisant
Attribution des pixels	Insuffisant
Conservation de la propriété	Passable
Objectivité des formes	Insuffisant
BILAN	Passable

Tableau 1 : Evaluation des régions en fonction du type de problème.

EVALUATION :

Au niveau de l'évaluation en fonction du type de difficulté, nous gardons un point de vue très critique sur la qualité des régions proposées. Si l'image contient des régions simples de type très homogène, de texture régulière ou de dégradé régulier, nos résultats sont intéressants. Si l'image est plus complexe, les régions restent difficilement exploitables. Pour l'image du muscle par exemple, l'attribution des pixels est parfois insuffisante avec des frontières de régions quelques peu illusoires sans aucune discontinuité dans le voisinage immédiat. Ces erreurs sont dues à des dégradés locaux de texture qui font diverger les propriétés globales de la région et font perdre à notre croissance de régions son caractère adaptatif. Comme nous l'avons déjà indiqué, aucune information sur la forme ou la position relative des pixels n'est exploitée pour contraindre ou aider ce processus, ce qui est sans doute à la base des principales erreurs constatées et du manque d'"objectivité" de nos régions.

En ce qui concerne l'intervention de l'utilisateur, la marge de manœuvre est très grande, ce qui permet une adaptation à l'image traitée. Une forte valeur de Seuil permet de relâcher les contraintes et conduit à une décomposition plus simple ou plus grossière de l'image, tandis qu'une valeur nulle est adaptée pour une détection plus stricte des régions.

3.1.7 Discussion

Deux points nous semblent importants :

- Nos critères d'évaluation d'un pixel à agréger sont sans doute insuffisants, mais ils ont le mérite d'incorporer dans la même formule de nombreuses informations, et ils constituent une réponse partielle à un problème qui reste irrésolu. Notre démarche expérimentale nous permet de plus d'identifier plus clairement ces insuffisances. La prise en compte de la forme de la région est ainsi selon nous un des paramètres essentiels qui doit être combiné aux deux autres critères pour parvenir à une détection plus pertinente des indices régions.

Si les régions détectées sont parfois satisfaisantes, il faut donc rester conscient que de nombreuses imperfections existent et que nos critères restent incomplets. Un des intérêts majeurs de cette étude réside finalement dans l'identification des difficultés, ce qui est utile pour poursuivre d'éventuelles recherches.

- Une autre faiblesse de notre détecteur est d'obtenir des limites de régions découpées et loin de leurs véritables frontières. Ceci est dû aux effets de bord du calcul du gradient. En effet, le gradient calculé en 5×5 commence à être significatif à trois ou quatre pixels de la discontinuité. Ces effets de bords ont malgré tout l'avantage d'empêcher la région de s'étendre dans un passage étroit entre deux discontinuités très proches l'une de l'autre.

Pour remédier au problème du découpage des frontières et étendre un peu plus nos régions, nous pourrions envisager l'application d'un opérateur de dilatation. Une autre possibilité est de lancer un processus de détection des contours dans le voisinage pour localiser exactement la frontière.

La coopération avec un détecteur de contours est précisément la solution que nous proposons d'étudier dans la suite de ce chapitre.

3.2 Gestion de la coopération

Pour réaliser une coopération efficace, nous proposons d'utiliser les deux méthodes de détection de contours et d'agrégation de pixels présentées précédemment. Elles permettent un contrôle incrémental et une coopération qui s'intègre naturellement dans les deux sens.

3.2.1 Contrôle incrémental

L'incrémentalité consiste à exploiter immédiatement les résultats intermédiaires jusqu'au traitement de la dernière étape avant de revenir se focaliser dans un autre endroit de l'image et recommencer les traitements. L'intérêt d'un tel contrôle est de guider au mieux chaque étape en fonction des résultats qui viennent d'être obtenus. Par exemple, si une agrégation est problématique, une détection des contours peut commencer exactement là où se situe le problème, exploiter les résultats en cours de l'agrégation de points et déterminer si oui ou non un contour est présent.

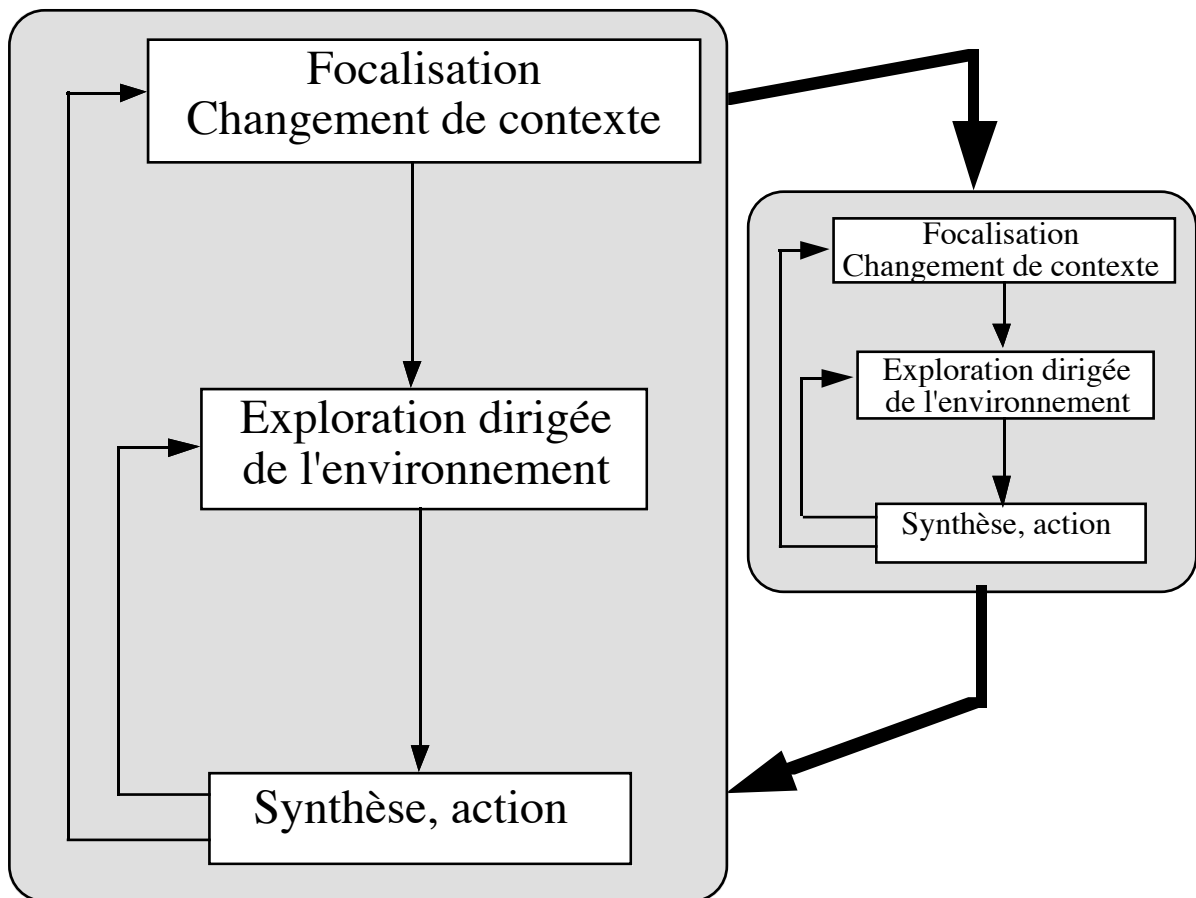


Figure 3 : Cycle de contrôle focalisation, exploration, action.

Pour qu'un contrôle incrémental soit efficace, il est nécessaire que chaque processus puisse s'interrompre de lui-même pour demander une coopération avec un autre processus (facilité de changement de contexte, liberté de l'information), de telle sorte que le lieu et la manière soient imposés et que des informations complémentaires soient obtenues. Nous proposons ainsi un cycle de contrôle "focalisation, exploration, synthèse et action" illustré figure 3 qui est indépendant de nos algorithmes (voir [SaGa 93a]).

- (1) La focalisation permet de préciser le contexte et d'analyser la situation.
- (2) L'exploration se justifie par la demande d'informations complémentaires sur le voisinage.
- (3) La synthèse est nécessaire pour faire émerger les résultats intermédiaires et préparer les étapes suivantes, ce qui implique donc une action.

A chacune de ces étapes, il est possible de commencer un autre cycle et d'imbriquer récursivement chaque niveau, ce qui permet une grande liberté d'informations.

Chacune des deux méthodes respectant le contrôle incrémental, il y a finalement deux cycles de traitement. Nous présentons figure 4 le schéma algorithmique général de notre coopération.

Légende : Objet
 Méthode

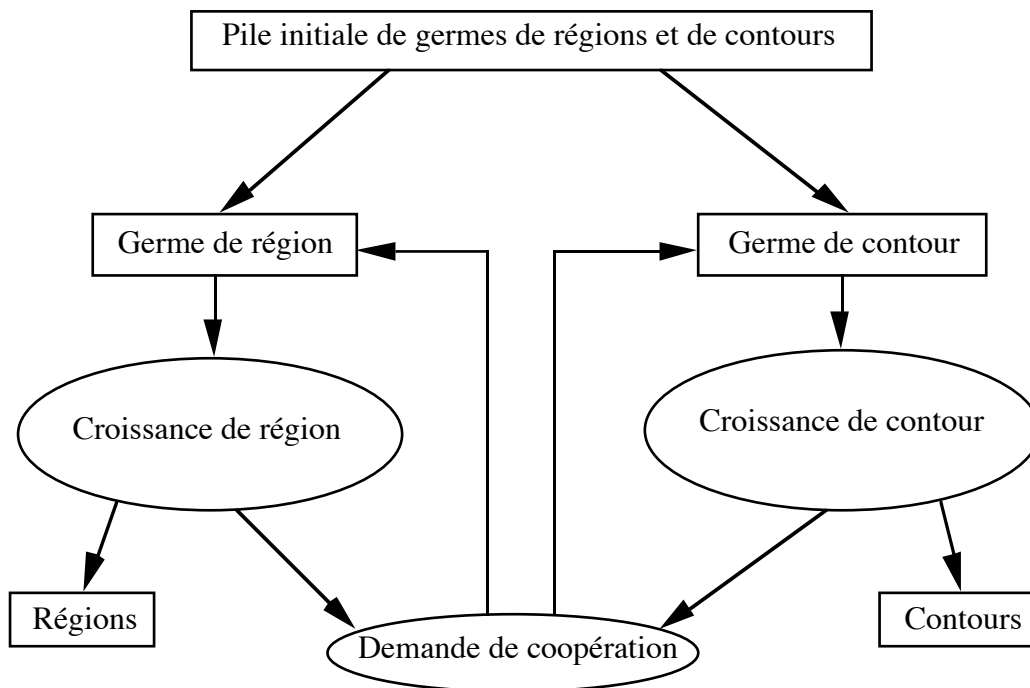


Figure 4 : Contrôle incrémental de la coopération

3.2.2 Enchaînement des méthodes

Nous proposons de gérer les événements à l'aide d'une pile de processus. Lorsqu'un processus demande une coopération avec un processus complémentaire, il doit spécifier le type de processus demandé ainsi que la position approximative du pixel germe. Ensuite, un germe du bon type est recherché dans un voisinage $N \times N$ centré sur ce pixel. C'est ce que nous appelons l'étape de focalisation (voir les sections suivantes).

Si un germe est accepté, les informations sur le processus en cours sont stockées dans la pile des processus en attente et un nouveau processus est immédiatement activé. Lorsque celui-ci se termine, les informations acquises sont délivrés au processus père qui est réactivé, sauf si d'autres germes sont demandés. La version qui vient d'être décrite a été implantée dans notre laboratoire par Fabrice Bellet [Bell 93]. La version originale que nous avons développée est moins sophistiquée : le stockage des informations du processus père est implicitement réalisé grâce à des appels direct aux procédures de détection complémentaires, avec empilement récursif. Notre contrôle est moins fort car nous ne pouvons donc pas choisir à tout moment le processus qui doit être activé. Nous imposons de plus un nombre limité d'empilements récursifs (et donc de coopération) pour des raisons de place mémoire.

Il arrive qu'un processus de croissance de régions soit bloqué par une autre région en cours de formation, dont le processus est en attente de réactivation. Dans d'autres cas, deux régions peuvent se retrouver voisines à 1 ou 2 pixels de distance sans aucune discontinuité entre les deux. Si aucune discontinuité n'a été validée entre les deux régions, celles-ci pourraient être fusionnées dans certaines conditions. Nous n'avons pas développé de procédure de fusion, mais il est certain que la description pourrait parfois en bénéficier. De même, si deux extrémités de contours se rejoignent, ils devraient pouvoir être fusionnés pour ne former qu'une seule chaîne. Ce traitement n'a pas été implanté dans la version actuelle.

3.2.3 Création des premiers germes

Au début de la coopération, l'image est échantillonnée en fenêtres de $N \times N$ pixels. N est laissé au choix de l'utilisateur (10 par défaut). Dans chaque fenêtre, un germe de contour et un germe de région sont recherchés.

Pour un germe de croissance de contour, le pixel choisi est celui qui n'est pas marqué et qui a la valeur de gradient la plus forte. Si ce gradient n'est pas un maximum local dans la direction de celui-ci, le germe est refusé.

Pour un germe de croissance de région, le pixel choisi a la plus faible valeur d'écart-type local, calculé dans un voisinage 3x3.

Tous les germes acceptés sont classés dans deux piles par ordre de qualité. Les germes de contours sont classés proportionnellement au gradient et les germes de région sont classés inversement proportionnellement à l'écart-type local.

Chaque germe est un processus potentiel qui est activé en fonction de son classement dans la pile, à condition que la zone ne soit pas marquée. Après activation, aucun autre processus de la pile des germes n'est activé tant que celui qui est en cours n'est pas terminé, avec toutes les demandes de coopération qu'il aura pu occasionner. Notons qu'il y a donc deux sortes de germes, ceux qui sont choisis au début de l'algorithme et ceux qui sont demandés pour la coopération.

Il reste à définir une stratégie pour choisir le type du germe. Nous pouvons en effet commencer par tous les germes de régions puis tous les germes de contours ou réciproquement. Pour cela, nous proposons une étude expérimentale de l'étape de sélection des germes :

CHOIX 1 : Les germes de contours sont choisis en premier.

CHOIX 2 : Les germes sont sélectionnés dans des fenêtres 5x5, au lieu de la fenêtre 10x10 proposée par défaut.

CHOIX 3 : Les germes de région et de contour sont rangés dans une seule pile en fonction de l'ordre dans lequel il sont rencontrés lors du parcours de l'image (sans tenir compte de la force du gradient et de l'écart-type local). Nous proposons un parcours de haut en bas et de gauche à droite.

CHOIX 4 : Nous proposons de noter chaque germe de la façon suivante :

- La note d'un germe de contour est sa valeur de gradient.
- La note d'un germe de région est donnée par la formule :

$$\text{note} = 256 / (\text{ecart-type local} + 1)$$

De cette manière, nous obtenons un interclassement des deux types de germe. En choisissant celui qui a la meilleur note, on ne peut savoir à l'avance quel est le processus qui va être lancé.

RESULTATS :

Nous présentons ci-dessous les résultats pour l'image du muscle.

EVALUATION :

- Le fait remarquable de cette étude est sans doute la faible variation des résultats en fonction de la technique de sélection des germes. Les principales régions sont les mêmes pour toutes les versions. Il existe quelques différences au niveau de la frontière exacte des régions, mais ceci n'est pas important puisque nous savons que celle-ci est mal localisée et que la véritable frontière doit se trouver au niveau du contour.

- Le choix 2 permet une segmentation plus complète de l'image. Toutefois, de nombreuses petites régions peu pertinentes sont obtenues ce qui n'est pas d'un grand intérêt pour la description de l'image.

- Les résultats des choix 1 et 3 sont très semblables à ceux du choix 4. Le tri des germes ne semble donc pas présenter d'améliorations sensibles.

BILAN :

De manière générale, même si les résultats sont parfois un peu différents, la qualité de la description ne dépend pratiquement pas de l'étape de sélection des germes. Toutefois, nous préférons le choix numéro 4, ceci afin de traiter l'image du plus simple vers le plus compliqué. Si ce choix n'est pas clairement justifié pour cette image, nous pensons que cela pourrait s'avérer intéressant pour d'autres images, surtout si nous envisageons un contrôle plus important de l'enchaînement des processus.

3.2.4 Aide des contours vers les régions

Nous expliquons ici comment le processus de croissance de régions peut bénéficier de l'aide du détecteur de contours.

Le processus de croissance de régions par agrégation de points est très robuste. En mode autonome, il existe pratiquement une frontière de région pour chaque contour visible. Le problème n'est donc pas d'arrêter le processus. En revanche, il y a parfois une sursegmentation et une très mauvaise localisation de la frontière. Ceci s'explique aisément par deux raisons :

- Une divergence faible mais régulière de la nature de la région entraîne un arrêt prématuré du processus.
- Les forts gradients proches d'une frontière s'étendent souvent un peu, ce qui a pour effet de décaler la frontière de la région à quelques pixels de sa véritable position.

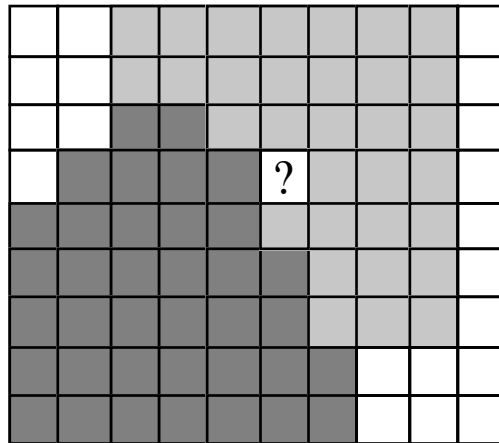


Figure 5 : Zone de recherche d'un germe de contour (quadrillage diagonal) après refus d'agrégation d'un pixel (le point d'interrogation) à la région (en gris foncé).

Connaissant les faiblesses de la croissance de régions, il s'agit donc d'exploiter au mieux les informations sur les contours pour aider le processus dans les cas difficiles. Etant donnée la robustesse de ce processus, aucune interruption ne se justifie vraiment, tant qu'il y a des pixels à agréger à la région. Lorsqu'un pixel de la pile "Pile des voisins" est rejeté, il s'agit de s'assurer qu'il existe bien un contour dans le voisinage, qui suive approximativement la frontière de la région.

Pour cela, nous proposons de lancer un processus de croissance de contour dans le voisinage extérieur de la région, pour obtenir la véritable frontière de la région, sauf bien sûr si ce contour existe déjà. Le germe de contour est recherché dans un voisinage $N \times N$ centré sur le pixel refusé (voir figure 5). Le germe éventuellement retenu ne doit pas être marqué et le gradient doit être maximal localement. Si le contour est confirmé, la région devrait être propagée jusqu'au contour, mais ce traitement parfois délicat n'a pas été implanté. En revanche, chaque pixel-contour pointe vers la structure région qui lui est associée. Ensuite, le processus d'agrégation reprend avec les pixels restant dans la pile "Pile des voisins". Pour simplifier la coopération et ne pas provoquer des demandes de contours à répétition, seules les régions de taille raisonnable (nous proposons supérieur à 30 pixels) sont autorisées à demander une recherche de contours.

Il reste à déterminer N qui caractérise la taille de la fenêtre de focalisation. Pour cela, nous proposons une étude expérimentale à la fin de la section suivante.

3.2.5 Aide des régions vers les contours

Le processus de croissance de contours exploite déjà des informations sur les régions pour calculer le "gradient relatif". Toutefois, la recherche d'informations sur les régions voisines avec des fenêtres 3x3 peut avantageusement être remplacée par le détecteur de régions :

- Si les régions situées de part et d'autre du pixel-contour examiné sont de type différent, le pixel-contour est systématiquement accepté.
- Sinon, pour calculer le gradient relatif, il suffit de récupérer les informations d'écart-type de ces régions. Les informations étant plus globales, les critères peuvent être affinés et la décision devient plus fiable.

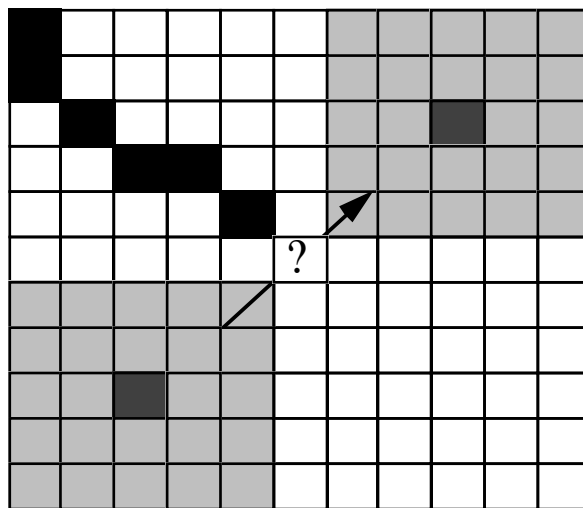


Figure 6 : Zones de recherche d'un germe de région lorsqu'un pixel-contour n'est pas évident (le ?). La flèche indique la direction du gradient sur le contour marqué en noir.

Si le gradient est supérieur au "seuil de fort gradient" (nous proposons une valeur de 30 par défaut), le contour peut aisément être détecté sans avoir recours aux informations sur les régions voisines. En revanche, en dessous de ce seuil, les informations sur les régions voisines sont automatiquement requises. Si une région existe déjà, les informations sur cette région sont récupérées à l'aide d'un tableau de pointeurs de la taille de l'image. Chaque élément de ce tableau pointe vers une structure de région ou de contour, ce qui permet d'éviter une recherche coûteuse. Si aucune région n'existe, deux processus d'agrégation de points de part et d'autre du pixel-contour en question sont demandés (voir figure 6).

Pour cela, il y a une recherche de germes dans une fenêtre $N \times N$ à une distance de K pixels du pixel-contour problématique dans la direction du gradient.

En ce qui concerne la valeur de K , nous proposons qu'elle soit fonction de N et du type du contour. Si le contour est de type "marche", nous suggérons de prendre $(N \div 2 + 1)$ pour coller la zone de recherche au contour. Si le contour est de type "trait", il faut franchir en général un pixel de plus (dont le gradient est de type marche), nous suggérons donc une valeur de k égale à $(N \div 2 + 2)$. Nous proposons d'étudier la valeur de N à la fin de cette section.

Si la demande de croissance de régions est refusée (nombre de processus trop important), les informations sur les régions voisines sont tout de même obtenues grâce à des fenêtres 3×3 pour préserver le calcul du gradient avec les formules du gradient relatif.

Les heuristiques d'évaluation du détecteur de contours restent inchangées, les régions représentées par les fenêtres 3×3 étant donc remplacées dans la mesure du possible par les régions globales. Notons que pour simplifier, le traitement du dégradé sur le pixel-contour est supprimé (des régions locales étaient nécessaires).

Si un contour n'est pas fermé, ou s'il est inclus dans une même région, celui-ci est conservé s'il respecte le critère de visibilité globale présenté au chapitre 2. Il peut donc arriver qu'un contour reste présent au milieu d'une région. Ceci ne nous semble pas gênant, car le but est d'effectuer une segmentation descriptive. Si un contour a été validé, c'est qu'il doit exister une discontinuité locale, il est donc normal que celui-ci subsiste.

Il nous reste à choisir une valeur de N pour déterminer la fenêtre de focalisation. Pour cela, nous proposons une étude expérimentale très simple. Nous observons les résultats pour deux valeurs de N et nous décidons de la meilleure stratégie. Par la même occasion, nous présentons une évaluation de la taille de la fenêtre de focalisation pour le processus complémentaire étudié section précédente.

CHOIX 1 :

- fenêtre $N \times N$ de focalisation pour demande de contours : $N = 7$
 - fenêtre $N \times N$ de focalisation pour demande de régions : $N = 5$
- (C'est ce choix qui est illustré figures 5 et 6).

CHOIX 2 :

- fenêtre $N \times N$ de focalisation pour demande de contours : $N = 9$
- fenêtre $N \times N$ de focalisation pour demande de régions : $N = 7$

RESULTATS :

Nous présentons ci-dessous les résultats pour les images IRM et muscle. Les paramètres des deux détecteurs sont choisis par défaut.

EVALUATION :

Peu de différences sont constatées. Alors que des discontinuités absentes avec le premier choix sont présentes dans le deuxième cas, quelques oublis sont aussi à noter pour cette dernière. L'évaluation est sans doute biaisée par le nombre limité d'empilements récursifs. Certaines demandes de coopération sont refusées, ce qui nuit à la qualité des résultats dans certaines parties de l'image, que ce soit pour l'une ou pour l'autre version.

BILAN :

Les différences étant peu significatives, nous conservons donc le choix 1 qui est de moindre coût, avec donc des petites tailles de fenêtre.

4. Evaluation

Nous proposons une évaluation en deux parties. Dans un premier temps, nous étudions les résultats obtenus sur quelques images, puis nous effectuons une évaluation globale pour conclure cette étude.

4.1 Analyse de quelques résultats

RESULTATS :

Nous proposons d'examiner les résultats pour les images suivantes :

- 1) Image du muscle.
- 2) Image de la femme.
- 3) Image de la souris.
- 4) Image IRM

Pour traiter chaque image, nous choisissons tous les paramètres par défaut. Les résultats sont présentés en annexe 3, sauf pour l'image IRM qui a déjà servi pour déterminer un choix algorithmique section 3.2.5.

4.1.1 Image du muscle

Les contours obtenus correspondent assez bien aux discontinuités visibles dans l'image. En ce qui concerne le gain de la coopération, il n'est pas évident. La détection des contours est en fait très robuste en mode autonome (elle exploite déjà des informations sur les régions voisines avec les fenêtres 3x3). Pour certains contours difficiles entre cellules claires texturées, la détection est plus complète. Ceci est dû à la prise en compte des régions globales au lieu des fenêtres 3x3. Réciproquement, certains contours manquent car le critère d'agrégation choisi par défaut conduit parfois à des régions un peu trop étendues avec une conservation imparfaite des propriétés locales.

Les régions obtenues ne sont pas toujours pertinentes. Si notre détecteur région reste adaptatif dans la texture, il découpe un peu trop certaines zones où sont présents des dégradés locaux. Inversement, certaines régions sont un peu trop étendues, les paramètres par défaut n'étant pas assez stricts. Le gain dû à la coopération n'est pas "visible". Comme nous l'avons expliqué, nous n'interrompons pas le processus de croissance de régions, il est donc normal d'obtenir des résultats du même ordre que ceux

présentés lors de l'évaluation de cette méthode. Cependant, le gain existe, puisque l'intérêt de la coopération est ici de trouver la véritable frontière de la région.

Les résultats obtenus sont finalement d'assez bonne qualité, ce qui nous laisse envisager une éventuelle poursuite de nos travaux par une tentative de reconnaissance des cellules.

4.1.2 L'image de la femme

Il est très difficile d'évaluer les résultats obtenus pour cette image.

En ce qui concerne les contours, les résultats correspondent aux discontinuités les plus visibles. Le contour délimitant la droite du visage n'a pas été bien détecté, mais nous avons vu au chapitre 2 que sa visibilité était difficile et même partielle.

En ce qui concerne les régions, les résultats sont intéressants :

- Au niveau du visage, de nombreuses régions en dégradé ont été détectées, ce qui permet une description pertinente. Très peu de contours existent entre ces régions, ce qui est normal car aucune discontinuité n'y est visible.
- Les différentes régions correspondant au fond ont été correctement détectées. Toutefois, au niveau de la joue de la femme sur la droite de l'image, il existe un léger empiètement de la région du fond, ce qui aurait pu être évité avec des critères plus stricts.

Si les résultats sont relativement satisfaisants, ils restent insuffisants pour envisager la reconnaissance, même si nous ajoutons une étape d'association d'indices. Le problème est en fait trop complexe, il faudrait notamment proposer d'autres indices visuels.

4.1.3 L'image de la souris

La qualité de détection des contours reste du même ordre que celle présentée au chapitre 2. Là encore, peu importe l'échelle où on se place, l'essentiel est donc de tenir compte des informations sur les régions voisines en calculant le gradient relatif. Notons qu'aucun contour n'a été validé entre les régions en dégradé du dessus de la souris, ce qui est satisfaisant.

Une des caractéristiques intéressantes est la détection de nombreux petits indices visuels régions ou contours dans la texture du bois. Ces résultats sont satisfaisants, car il

correspondent effectivement à ce que nous voyons. Pour identifier cette texture, il faudrait alors associer ces indices visuels qui ont des propriétés similaires.

En ce qui concerne la forme arrondie de la souris, la présence de dégradés est une information qui peut être exploitée pour une éventuelle discrimination de cet objet. La somme des indices visuels détectés fournit finalement une description relativement riche de l'image.

4.1.4 L'image IRM

Les indices contours et régions détectés sont satisfaisants, ils capturent les informations essentielles de l'image. En effet, il existe de nombreuses petites zones qui ne correspondent à aucun indice visuel détecté. Ces zones ne présentant aucune caractéristique intéressante, la description est ainsi simplifiée tout en restant pertinente. Suivant l'application, il est de toutes façons possible de choisir un nombre de germes plus important au départ pour obtenir plus de détails.

Une des difficultés majeures est la représentation du fond. Une seule région ne capturerait pas l'information de texture (ou est-ce du bruit) qui traverse horizontalement l'image. En revanche, plusieurs régions conduiraient à une décomposition arbitraire peu significative. Notre algorithme de coopération fournit plusieurs régions qui sont donc peu significatives. Il n'existe en fait aucune solution pleinement satisfaisante. Il faudrait sans doute passer par une décomposition hiérarchique des régions. De cette façon, les régions trouvées pourraient être regroupées sans qu'il y ait perte d'informations.

4.2 Evaluation globale

4.2.1 Utilité de nos principes

Un des aspects positifs mais peu apparent de notre coopération réside à la fois dans le report de la décision délicate, l'accumulation des informations et leur "liberté de mouvement". En effet, l'application d'un processus complémentaire dès que la décision est difficile a pour conséquence une accumulation du nombre d'informations pertinentes, ce qui permet dans certains cas d'éviter des erreurs, grâce à ce que nous pourrions appeler des effets de bords.

Ceci est illustré figure 7 à l'aide d'un dessin qui est inspiré d'un exemple réel :

Une croissance de région s'interrompt lorsque le premier pixel (appelons le P1) de la pile "Pile des voisins" est rejeté. Une croissance de contour est appliquée à l'endroit problématique et un peu plus loin, une autre croissance de région est demandée, ce qui permet d'obtenir 2 régions. Le résultat est relativement satisfaisant (même si le contour séparant les deux régions est un peu court).

En revanche, une croissance de région sans coopération conduit à une région unique avec un amincissement local. Cela est dû au fait que des pixels classés après P1 dans la pile "Pile des voisins", donc *a priori* de qualité inférieure, sont réévalués positivement en raison d'un léger changement des attributs de la région. Si la coopération est absente, il y a alors une expansion mal contrôlée de cette région, justement à partir de l'amincissement local et une seule région est trouvée.

L'accumulation des informations et la "liberté de mouvement" de celles-ci conduise donc par effets de bords à de meilleurs résultats.

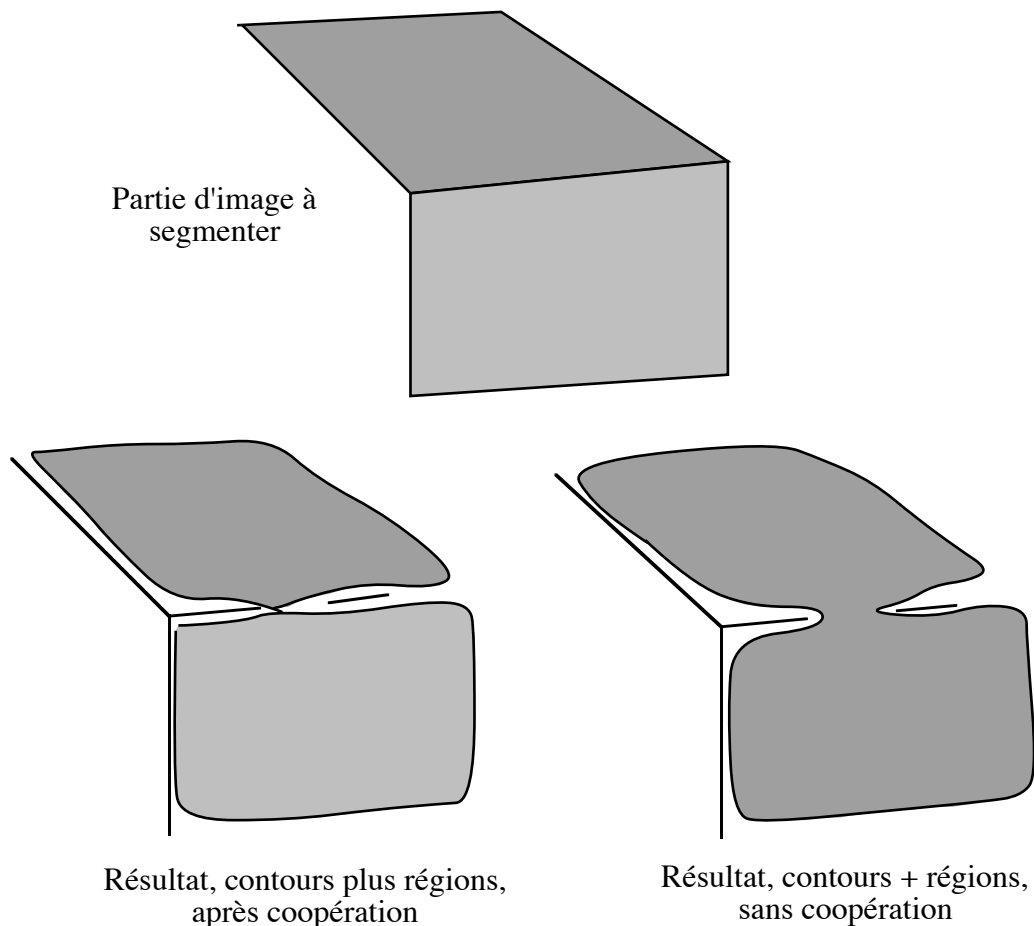


Figure 7 : Apport de la coopération grâce au report de la décision délicate.

4.2.2 Evaluation qualitative des résultats

Si nous juxtaposons le résultat du détecteur de contours et du détecteur de régions sans effectuer de coopération, nous obtiendrions une carte des régions-contours à peu près semblable. Là s'arrêterait toute comparaison, car le gain est important au niveau des relations entre les contours et les régions, ce qui n'est pas directement visible.

En ce qui concerne la détection des contours, nous conservons donc la même qualité de détection. La prise en compte des régions globales ne permet pas d'améliorer de façon sensible les résultats. L'essentiel est finalement de tenir compte des régions voisines, peu importe l'échelle à laquelle on se place.

Notons qu'il manque parfois des contours, mais l'utilisateur peut imposer un plus grand nombre de germes initiaux pour obtenir le résultat désiré.

En ce qui concerne la détection des régions, nous présentons tableau 2 une nouvelle évaluation qualitative. L'apport de la coopération est ici évident. En effet, les frontières découpées et mal localisées des régions sont remplacées par les contours. Le problème de l'attribution des pixels en bordure de région est donc partiellement résolu et les formes sont plus objectives.

Nous considérons finalement que nos résultats sont assez satisfaisants dans l'ensemble.

Régions homogènes	Satisfaisant
Textures régulières	Satisfaisant
Dégradés réguliers	Satisfaisant
Attribution des pixels	Passable
Conservation de la propriété	Passable
Objectivité des formes	Passable
BILAN	assez satisfaisant

Tableau 2 : Evaluation des régions en fonction du type de problème.

- Notons qu'au niveau des temps de calculs, il faut environ 2 minutes sur un Sun Sparc Station 2 pour traiter l'image 256x256 du muscle (avec visualisation en temps réel de tous les résultats intermédiaires, régions et contours). Nous n'avons toutefois pas examiné en détail les problèmes de temps.

5 Conclusion sur la coopération

5.1 Aspects positifs

- La coopération que nous proposons est très riche. Les processus qui coopèrent ont une structure de contrôle très souple, permettant une grande liberté de mouvement de l'information, avec changement de contexte rapide et efficace, chaque processus pouvant s'interrompre à tout moment pour demander une coopération avec un autre processus. Une structure de contrôle incrémentale permet non seulement de guider les processus coopérants mais également une focalisation très forte là où se situent précisément les problèmes. La coopération est ainsi pratiquement intégrale au niveau de la décision interne à chaque processus, contrairement à l'approche classique où la coopération consiste plutôt à trouver le meilleur compromis entre les résultats de chaque méthode.

- Le tri des premiers germes acquis après échantillonnage de l'image permet un traitement du plus simple vers le plus compliqué. Le traitement des cas difficiles arrive donc en dernier, ce qui entraîne souvent une délimitation de la zone problématique et évite des débordements mal contrôlés du processus de croissance de régions dans les zones à texture variable.

- Le typage des régions apporte un plus descriptif non négligeable. Pour certaines applications, les régions en dégradé sont des indices visuels très pertinents permettant une reconnaissance rapide. Le type texture est également intéressant pour des applications comme l'identification des cellules de l'image du muscle.

- Nous voudrions souligner à nouveau le caractère expérimental de notre approche. De nombreuses décisions sont prises à l'aide d'heuristiques. Le nombre d'informations disponibles est très grand, ce qui permet une décision fiable, à condition toutefois que les heuristiques soient judicieusement choisies. Pour trouver les meilleures heuristiques, il est nécessaire de se fier à l'expertise humaine, à l'évaluation et à l'analyse des résultats expérimentaux. Il nous semble que cette méthodologie est difficile à mettre en œuvre mais qu'elle permet une richesse et un potentiel algorithmique de grande qualité. Pour l'avoir expérimentée, nous pensons qu'elle constitue une voie prometteuse vers des résultats honorables en segmentation descriptive. Le fait même de devoir évaluer de façon pragmatique conduit à une profonde implication du programmeur dans le problème et une véritable découverte des difficultés mais aussi des solutions possibles. Ceci constitue donc un aspect très positif de notre approche.

5.2 Aspects négatifs

- En premier lieu, si notre coopération est potentiellement très puissante, les résultats ne montrent pas clairement le gain de chaque méthode au niveau de la carte régions-contours. Il se trouve en effet que notre détecteur est déjà à lui tout seul un système coopérant et que le gain des régions est "dissimulé" dans les relations entre contours et régions. Le bénéfice de la coopération peut tout de même sembler insuffisant, en regard des moyens déployés. Nous pensons néanmoins que ce jugement est provisoire car nous sommes loin d'avoir exploité tout le potentiel de cette coopération (voir la section perspectives).

- Si nos résultats constituent des indices visuels primaires d'une certaine qualité, ils restent difficile à exploiter. Il manque bien sûr toutes les étapes intermédiaires de la vision qui ont pour but de construire des indices visuels de plus haut niveau, mais les régions obtenues n'ont pas toujours une forme simple, les contours frontières étant parfois incomplets ou même absents.

- Il faudrait rajouter une procédure à notre coopération pour étendre les régions jusqu'aux contours qui les délimitent. Ce problème n'est toutefois pas trivial, en raison du grand nombre de cas particuliers.

- Nos régions présentent de nombreux "trous" correspondant au rejet de certains pixels jugés de qualité médiocre. Ce problème pourrait être résolu par une fermeture morphologique, mais nous perdrons alors l'information de dissimilarité de ces pixels. Ce traitement peut en fait être décidé en fonction de l'application.

5.3 Perspectives

Deux aspects essentiels méritent une attention toute particulière : les problèmes liés aux critères d'agrégation qui sont pour l'instant insuffisants et les problèmes liés à la manière d'aider le processus complémentaire qui pourrait être bien plus efficace.

5.3.1 Les critères d'agrégation

Les critères liés à la forme nous semblent d'une importance capitale. Nous avons considéré uniquement les propriétés concernant les niveaux de gris des régions, cela nous

a conduit à deux types d'erreur qui sont classiques pour la plupart des algorithmes de segmentation en régions :

(1) Il y a une sursegmentation due à une divergence des propriétés globales de la région. Pour l'image IRM par exemple, le fond présente des zones de texture plus ou moins forte, ce qui a conduit à plusieurs régions de forme très découpée qui ne s'appuient sur aucune frontière véritable. Ce problème existe aussi pour l'image des cellules avec des refus d'agrégation pour des pixels qui ne sont pas à une véritable frontière et qui participent à des légers dégradés tout juste perceptibles. De manière générale, le changement de propriétés des niveaux de gris d'une zone de l'image peut se faire de façon assez régulière, ce qui interdit toute discontinuité à l'intérieur de cette zone, mais ce qui entraîne aussi une sursegmentation en régions. Cette difficulté ne peut être résolue simplement. La solution pourrait passer par une prise en compte de la forme et par une décomposition hiérarchique de régions. Par exemple, en cas de changement régulier des propriétés de la région, plutôt que d'arrêter celle-ci en fonction de critères relatifs aux niveaux de gris (avec implicitement des formes quelques peu aléatoires), il doit être possible de découper la région en sous-régions de formes simples et de continuer la croissance par rapport à celles-ci afin de conserver toute l'adaptabilité du processus. Ce découpage pourrait également se faire lorsque la forme de la région est trop complexe, afin de fournir une description qui soit exploitable.

(2) Il y a une segmentation trop tolérante dans certains cas particuliers. Certains pixels "limites" sont agrégés à la région alors qu'ils constituent un alignement avec une véritable discontinuité. Même si les propriétés locales respectent tout juste celles de la région, l'alignement devrait être pris en compte pour éviter ce type d'erreur (corrélation entre niveau de gris et position des pixels). En présence de texture, il est difficile de faire la part entre discontinuité de faible amplitude et variations locales irrégulières plus importantes. Les informations sur la forme sont alors des indices essentiels pour déterminer la frontière la plus pertinente de la région. La forme est cependant véhiculée par les contours, il faudrait donc que la croissance de régions soit contrainte par la position relative des discontinuités du voisinage, ce que nous n'avons malheureusement pas fait.

5.3.2 Une aide plus efficace entre les processus

- Nous avons imposé dès le départ que la croissance de régions ne devait pas être interrompue tant que des pixels pouvaient être agrégés. Cette contrainte est trop forte et limite implicitement l'aide éventuelle apportée par le détecteur de contours.

Une des façons d'envisager une aide plus efficace est de relâcher petit à petit le critère d'agrégation, en demandant périodiquement une aide du détecteur de contours. En analysant la disposition des contours autour de la région, il doit ensuite être possible de relâcher le critère pour permettre à la région de coller au contour, ou au contraire d'être plus strict pour conserver une forme régulière.

- En ce qui concerne le détecteur de contours, nous n'avons pas exploité la moyenne des niveaux de gris des deux régions, ni la présence d'autres contours délimitant chaque région, ni même leur alignement ou leur disposition particulière. Les informations disponibles n'ayant pas été toutes exploitées, il serait intéressant de poursuivre notre étude.

- Enfin, nous n'avons pas envisagé la coopération contour-contour ou région-région, mais cela est tout à fait envisageable.

La coopération que nous proposons offre donc de nombreuses perspectives qui méritent d'être étudiées.

CONCLUSION GENERALE

Nous avons présenté une analyse de la gestion des informations pour les algorithmes de détection d'indices visuels primaires, ce qui nous a permis de dégager quelques principes importants. Nous proposons le cumul des informations avant la prise de décision, la nécessité de guider les méthodes afin d'obtenir des informations complémentaires, l'exploitation d'informations "objectives" pour résoudre les problèmes d'émergence des indices visuels et enfin la "liberté" de l'information pour que la structure de contrôle soit efficace et permette par exemple le changement de contexte ou le report des décisions difficiles.

Des principes que nous avons dégagés au premier chapitre, nous avons tiré une méthodologie fondée sur une approche pragmatique des problèmes, avec une forte évaluation expérimentale qualitative et l'exploitation d'heuristiques. Ceci nous permet d'analyser et de comprendre les difficultés pour finalement essayer d'approcher la solution optimale au moyen de stratégies heuristiques.

En appliquant notre méthodologie, nous avons construit un nouveau détecteur de contours fondé sur l'heuristique du gradient relatif. Les résultats de ce détecteur adaptatif nous semblent très honorables, ce qui montre toutes les qualités de notre approche.

Nous avons également proposé une technique de coopération entre notre détecteur de contours et un processus de croissance de régions par agrégation de pixels. Si notre structure de contrôle est particulièrement souple, nos résultats ne montrent pas clairement l'apport de la coopération entre les deux processus. Notre croissance de régions est originale avec détection de plusieurs types de régions, homogène, texture ou dégradé, et segmentation partielle de l'image. Cette originalité, dûe encore une fois à notre méthodologie, montre avant tout qu'il est possible d'aborder différemment les problèmes et que de nombreux progrès sont encore possibles dans le domaine de la segmentation descriptive. Même si nos résultats sont intéressants, nous avons exprimé des critiques sévères sur notre algorithme, car nous avons sans doute trop simplifié les problèmes, notamment en ce qui concerne les informations sur la forme.

La conclusion la plus importante, concernant à la fois la détection des contours et des régions dans le cadre de la détection d'indices visuels primaires, est qu'il est possible d'aborder différemment les problèmes. Trop souvent le but a été trop simplifié, voire

dénaturé, ce qui a conduit pratiquement au "mythe de la segmentation", au "problème mal posé" ou à la solution impossible. Nous sommes d'avis que ce problème est extrêmement complexe mais qu'il est tout à fait abordable. Il faut selon nous analyser profondément les difficultés et éviter de simplifier le but et les moyens.

Finalement, après plus de trente ans de travaux dans ce domaine, nous pensons qu'il reste encore beaucoup à faire, que les solutions existent, mais que leur complexité demande des efforts considérables, tant au niveau algorithmique qu'au niveau méthodologique.

Il s'agit là sans doute d'un des plus grands challenges de l'intelligence artificielle.

Références

- [BaBr 82] D.H. Ballard and C. Brown, "Computer vision", Prentice Hall Inc., Englewood Cliffs, New Jersey, USA, 1982.
- [BaGa 93] O. Baujard and C. Garbay, "KISS : A multi-agent segmentation system", SPIE Optical Engineering, vol. 32, n°6, p. 1235-1249, june 1993.
- [Bell 93] F. Bellet, "Approche pragmatique pour la détection des contours dans une image", Rapport de DEA d'Informatique de l'ENSIMAG, INPG, juin 1993.
- [BeLe 93] R. Bergevin and M. D. Levine, "Generic object recognition: Building and matching coarse descriptions from line drawings", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 15, n°1, janvier 1993.
- [Berg 87] F. Bergholm, "Edge focusing", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 9, n°6, p. 726-741, novembre 1987.
- [Bied 85] I. Biederman, "Human image understanding: recent research and a theory", CVGIP, vol. 32, p. 29-73, 1985.
- [Bonn 91] P. Bonnin, "Méthode systématique de conception et de réalisation d'applications en vision par ordinateur", Thèse de Doctorat de l'Université de Paris VII, 1991.
- [BrFe 70] C.R. Brice and C.L. Fennema, "Scene analysis using regions", Artificial Intelligence, vol. 1, p. 205-226, 1970.
- [Broo 81] R.A. Brooks, "Symbolic reasoning among 3d objects and 2D models", Artificial Intelligence, vol. 17, p. 285-348, 1981.
- [Cann 86] J. Canny, "A computational approach to edge detection", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 8, n°6, p. 679-698, novembre 1986.

- [ChGa 84] J.M. Chassery and C. Garbay, "An iterative segmentation method based on a contextual color and shape criterion", *IEEE Trans. PAMI*, vol. 6, p. 794-800, 1984.
- [Chen 92] L. Chen, "A two-phase area-level line/edge detector", *Pattern Recognition*, vol. 25, n°1, p. 55-63, 1992.
- [ChAg 90] C. Chu and J. K. Aggarwal, "The integration of region and edge-based segmentation", *IEEE Proceedings of the 3rd International Conference on Computer Vision, Osaka (Japon)*, p. 117-120, 1990.
- [Clem 90] V. Clement, "Raisonnements cognitifs appliqués au pilotage d'algorithmes de traitement d'images", *Thèse INRIA, Nice-Sophia-Antipolis*, 1990.
- [Cocq 85] J.P. Cocquerez et J. Devars, "Détection de contours dans les images aériennes : nouveaux opérateurs", *Traitement du signal*, vol. 2, n°1, 1985.
- [CoGP 92] J.P. Cocquerez, P. Gaussier et S. Philipp, "Système d'interprétation mixte : réseau de neurones / système-expert appliqué aux images aériennes", *Traitement du Signal*, vol. 8, n°6, numéro spécial IA, 1991.
- [CoBW 90] I.J. Cox, R. A. Boie and D.A. Wallach, "Line recognition", *IEEE Proceedings of the 10th International Conference on Pattern Recognition, Atlantic City (New Jersey)*, vol.2, p. 639-645, 1990.
- [DCOT 89] E. De Micheli, B. Caprile, P. Ottonello and V. Torre, "Localization and noise in edge detection", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 11, n°10, p. 1106-1117, octobre 1989.
- [Dema 86] Y. Demazeau, "Niveaux de représentation pour la vision par ordinateur. Indices d'images et indices de scène", *Thèse de doctorat de l'Institut National Polytechnique de Grenoble*, 1986.
- [Deri 87] R. Deriche, "Using Canny's criteria to derive a recursively implemented optimal edge detector", *International Journal of Computer Vision*, p. 167-187, 1987.

- [DIAL 91] DIALOGUE, "Ignorance, Myopia and Naiveté in Computer Vision Systems", CVGIP: Image Understanding, vol. 53, n° 1, 112-117, 1991.
- [FMAD 92] O. Faugeras, J. Mundy, N. Ahuja, C. Dyer, A. Pentland, R. Jain, K. Ikeuchi, "Workshop Panel Report: Why aspects graphs are not (yet) practical for computer vision", CVGIP: Image Understanding, vol. 55, n°2, p. 212-218, mars 1992.
- [Fish 58] W.D. Fisher, "On grouping for maximum homogeneity", JASA, vol. 53, p. 789-798, 1958.
- [Flec 90] M.M. Fleck, "Multiple widths yield reliable finite differences", IEEE Proceedings of the 3rd International Conference on Computer Vision, Osaka (Japon), p. 58-61, 1990.
- [Flec 92] M.M. Fleck, "Some defects in finite-difference edge finders", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 14, n°3, p. 337-345, 1992.
- [Free 91] W. Freeman, "La physiologie de la perception", Pour La Science, n°162, p. 70-78, avril 1991.
- [FuMu 80] K.S. Fu and J.K. Mui, "A survey on image segmentation", Pattern Recognition, vol. 13, p. 3-16, 1980.
- [GaCo 91] P. Gaussier et J.P. Cocquerez, "Utilisation des réseaux de neurones pour la reconnaissance de scènes complexes: simulation d'un système visuel comprenant plusieurs aires corticales", Traitement du Signal, vol. 8, n°6, p. 441-466, 1991.
- [GaMo 85] A. Gagalowicz et O. Monga, "Un algorithme de segmentation hiérarchique", Actes du 5ème congrès RFIA de l'AFCEP, p. 163-177, 1985.
- [GaGM 89] P. Garnesson, G. Giraudon et P. Montesinos, "MESSIE: un système multi-spécialistes en vision. Application à l'interprétation en imagerie aérienne." Rapport de recherche INRIA, n° 1012, avril 1989.

- [GGGD 90] D. Geman, S. Geman, C. Graffigne and P. Dong, "Boundary detection by constrained optimization", IEEE PAMI, vol. 12, p. 609-628 1990.
- [Gira 87] G. Giraudon, "Chaînage efficace de contours", Rapport interne INRIA n°605, janvier 1987.
- [GrMi 87] S. Grossberg and E. Mingolla, "Neural dynamics of surface perception: boundary webbs, illuminants and shape-from-shading", CVGIP, n°39, p. 116-165, 1987.
- [Gros 80] S. Grossberg, "How does the brain build a cognitive code ?", Psychological Review, vol. 87, n°1, p. 1-51, janvier 1980.
- [Gros 87] S. Grossberg, "Cortical dynamics of three-dimensional form, color, and brightness perception : I. Monocular theory", Perception & Psychophysics, vol. 2, n°41, p. 87-116, 1987.
- [Guzm 68] A. Guzman, "Computer recognition of three-dimensional objects in visual scenes", Technical Report MAC-TR- 59, MIT, Cambridge, Massachussets; 1968.
- [GREC 91] Rapport du Groupe De Recherche 134 du CNRS Traitement du signal et images, "RAPPORT SEGMENTATION première partie : prétraitement et approche frontière", éditeur ENSEA-ETIS, GRECO signal images, 1991.
- [Hadd 88] J.F. Haddon, "Generalised threshold selection for edge detection", Pattern Recognition, vol. 21, n°3, p. 195-203, 1988.
- [HaRi 78] A.R. Hanson and E.M. Riseman, "VISIONS: a computer vision system for analysing scenes", A.R. Hanson and E.M. Riseman editors, Computer Vision Systems, Academic Press NewYork, 1978.
- [Hara 84] R.M. Haralick, "Digital step edges from second directional derivatives", IEEE Trans. PAMI, vol. 6, p. 58-68, 1984.
- [HaLe 88] R.M. Haralick and J. Lee, "Context dependent edge detection", Proceedings of the Int. Conf. on Computer Vision and Pattern Recognition, Ann Harbor, Michigan, p. 223-228, 1988.

- [HaSh 85] R.M. Haralick and L.G. Shapiro, "SURVEY: Image segmentation techniques", *Computer Vision, Graphics and Image Processing*, vol. 29, p. 100-132, 1985.
- [HKLR 81] L.D. Harmon, M.K. Khan, R. Lascha and P.F. Ramig, "Machine identification of human faces", *Pattern Recognition*, vol. 13, n° 2, p. 97-110, 1981.
- [HoPa 76] S.L. Horowitz and T. Pavlidis, "Picture segmentation by a Tree Traversal Algorithm", *Journal of the Association for Computing Machinery*, vol. 23, n°2, p. 368-388, avril 1976.
- [HuWi 79] D.H. Hubel and T.N. Wiesel, "Brain mechanisms of vision", *Scientific American*, p. 150-162, septembre 1979.
- [Ikeu 87] K. Ikeuchi, "Generating an interpretation tree from a CAD Model for 3D-object recognition in bin-picking tasks", *International Journal of Computer Vision*, p. 145-165, 1987.
- [JoMo 89] J.M. Jolion et A. Montanvert, "La pyramide adaptative : construction et utilisation pour l'analyse de scènes 2D", *Actes du 7ème congrès RFIA de l'AFCEC*, Paris, tome 1, p. 197-206, novembre 1989.
- [KJPH 93] T. Kanungo, M.Y. Jaisimha, J. Palmer and R.M. Haralick, "A quantitative methodology for analysing the performance of detection algorithms", *Proc. of the 4th Int. Conf. on Comp. Vis. (ICCV93)*, Berlin (Germany), p. 247-252, may 1993.
- [Kirs 71] R.A. Kirsch, "Computer determination of the constituent structure of biological images", *Computers and Biomedical Research*, vol. 4, n°3, p. 315-328, 1971.
- [Klop 82] A. H. Klopff, "The hedonist neuron: A theory of memory, learning and intelligence", Hemisphere publishing corporation, New-York, 1982.
- [Kund 90] A. Kundu, "Robust edge detection", *Pattern Recognition*, vol. 23, n°5, p. 423-440, 1990.

- [LeCh 90] S.U. Lee and S.Y. Chung, "A comparative study of several global thresholding techniques for segmentation", *Computer Graphics and Image Processing*, vol. 52, 171-190, 1990.
- [Lux 85] A. Lux, "Algorithmique et contrôle en vision par ordinateur", Thèse d'Etat de l'Institut National Polytechnique de Grenoble, 1985.
- [Marr 82] D. Marr, "Vision", Freeman editors, San Francisco, 1982.
- [MaHi 80] D. Marr, H. Hildreth, "Theory of edge detection", *Proceedings of the Royal Society of London B207*, p 187-217, 1980.
- [Mart 76] A. Martelli, "An application of heuristic search methods to edge and contour detection", *Communication of the ACM*, vol. 19, n°2, p. 73-83, février 1976.
- [MaHw 85] T. Matsuyama and V. Hwang, "SIGMA: a framework for image understanding", *Proc. 9th IJCAI*, p. 908-915, 1985.
- [Mats 89] T. Matsuyama, "Expert systems for image processing: Knowledge-based composition of image analysis processes", *CVGIP*, vol. 48, p. 22-49, octobre 1989.
- [McHD 85] D.M. McKeown, W.A. Harvey et J.McDermott, "Rule-based interpretation of aerial imagery", *IEEE PAMI*, vol. 7, n° 5, p. 570-585, septembre 1985.
- [MeCh 92] M. Melkemi and J.M. Chassery, "Edge-region segmentation process based on generalized Voronoi diagram representation", *Proc. 11th Int. Conf. Pattern Recog. (ICPR 92)*, The Hague, Vol. 3, 323-326, 1992.
- [MoMR 91] A. Montanvert, P. Meer and A. Rosenfeld, "Hierarchical image analysis using irregular tessellations", *IEEE Trans. PAMI*, vol. 13, p. 307-316, avril 1991.
- [NaMa 79] M. Nagao, T. Matsuyama, "Edge preserving smoothing", *Computer Graphics and Image Processing*, vol. 9, p. 394-407, 1979.

- [NaMM 91] O. Nakamura, S. Mathur and T. Minami, "Identification of human faces based on isodensity maps", *Pattern Recognition*, vol. 24, n°3, p. 263-272, 1991.
- [NaLe 84] A. Nazif and M. D. Levine, "Low-level image segmentation: An expert system", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 6, n°5, p. 555-577, septembre 1984.
- [OhPR 78] R. Ohlander, K. Price and D.R. Reddy, "Picture Segmentation using a recursive splitting method", *Computer Graphics and Image Processing*, vol. 8, p. 313-333, 1978.
- [PaLi 90] T. Pavlidis and Y. Liow, "Integrating region growing and edge detection", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 12, n°3, p. 225-233, mars 1990.
- [Pavl 92] T. Pavlidis, "Why progress in machine vision is so slow", *Pattern Recognition Letters*, vol. 13, p. 221-225, 1992.
- [Pear 87] J. Pearl, "HEURISTIQUE, stratégies de recherche intelligente pour la résolution de problèmes par ordinateur", Cepadues-éditions, 1987.
- [PeMa 82] T. Peli and D. Malah, "A study of edge detection algorithms", *Computer Graphics and Image Processing*, vol. 20, p. 1-21, 1982.
- [PeMa 90] P. Perona and J. Malik, "Detecting and localizing edges composed of steps, peaks and roofs", *IEEE Proceedings of the 3rd International Conference on Computer Vision, Osaka (Japon)*, p. 52-57, 1990.
- [Prew 70] J.M.S. Prewitt, "Object enhancement and extraction", *Picture Processing and Psychopics*, B.S. Lipkin and A. Rosenfeld editors, New York Academic Press 1970.
- [QuMD 91] T. Quiguer, P. Miché et R. Debrie, "Segmentation by auto-adaptative thresholding", *Progress in Image Analysis and Processing*, V. Cantoni, L.P. Cordella, S. Levialdi, G. Sanniti di Baja editors, *Proceedings of the 6th Int. Conf. on Image Analysis and Processing*, p. 34-41, 1991.

- [RiAr 77] E.M. Riseman and M.A. Arbib, "SURVEY : Computational techniques in the visual segmentation of static scenes", Computer Graphics and Image Processing, vol. 6, p. 221-276, 1977.
- [Robe 65] L.G. Roberts, "Machine perception of three-dimensional solids", Optical and Electro-optical Information Processing, J.P. Tippet et al editors, MIT Press Cambridge, Massachusets, 1965.
- [RoPa 91] I. Rock et S. Palmer, "L'héritage du gestaltisme", Pour la Science, n°160, p. 64-70, février 1991.
- [Salo 92] M. Salotti, "Reflexions sur la segmentation", Rapport de recherche TIM3 / IMAG, n° RR 873-I-, janvier 1992.
- [SaGa 92] M. Salotti and C. Garbay, "A new paradigm for segmentation", Proc. 11th Int. Conf. Pattern Recog. (ICPR 92), The Hague, Vol. 3, 611-614, 1992.
- [Salo 93] M. Salotti, "Progress in edge detection", Rapport de recherche TIMC / IMAG, n° RR 914-I-, Grenoble (France), avril 1993.
- [SaGa 93a] M. Salotti and C. Garbay, "Cooperation between edge detection and region growing : the problem of control", Proc. Int. Conf. on Image Analysis and Processing, San Remo (Italie), p. 95-98, juin 1993.
- [SaGa 93b] M. Salotti and C. Garbay, "Edge detection : Model-based vs heuristic-based approach", Proc. of 5th Univ. New Brunswick Conf. on Artif. Intell., << Are we moving ahead ? Lessons from computer vision >>, 12-14 august, Fredericton (Canada), 1993.
- [SaGa 94] M. Salotti et C. Garbay, "Détection de contours : les heuristiques remplacent avantageusement les modèles", Actes du congrès RFIA de l'AFCEC, Paris, janvier 1994.
- [SSWC 88] P.K. Sahoo, S. Soltani, A.K.C. Wong and Y.C. Chen, "A survey of thresholding techniques", Computer Vision Graphics and Image Processing, vol. 41, p. 233-260, 1988.

- [ShCa 92] J. Shen and S. Castan, "An optimal linear operator for step edge detection", CVGIP: Graphical Models and Image Processing, vol 54, n°2, p. 112-133, mars 1992.
- [Sobe 78] I. Sobel, "Neighbourhood coding of binary images for fast contour following and general binary array processing", Computer Graphics and Image Processing, vol. 8, p. 127-135, 1978.
- [SpLe 87] K.T. Spoehr and S.W. Lehmkuhle, "Visual Information Processing", Freeman and Company Ed., San Francisco, CA, 1987.
- [SpHe 92] L.J. Spreeuwers and F. Van der Heijden, "Evaluation of edge detectors using average risk", Proceedings 11th Int. Conf. Patt. recog. (ICPR 92), The Hague, vol. 3, p. 771-774, 1992.
- [TeBa 77] J.M. Tenenbaum and H.G. Barrow, "Experiments in interpretation-guided segmentation", Artificial Intelligence, vol. 8, p. 241-274, 1977.
- [ToPo 86] V. Torre and T. Poggio, "On edge detection", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 8, n°2, p. 147-163, mars 1986.
- [Wesz 78] J.S. Weszka, "A survey of threshold selection techniques", Computer Graphics and Image Processing, vol. 7, p. 259-265, 1978.
- [WeNR 74] J.S. Weszka, R.N. Nagel and A. Rosenfeld, "A threshold selection technique", IEEE Transactions on Computers, vol. 23, p. 1322-1326, 1974.
- [WeSk 77] H. Weschler and J. Sklansky, "Finding the rib cage in chest radiographs", Pattern Recognition, n°9, p. 21-30, 1977.
- [WrMo 87] B. Wrobel et O. Monga, "Segmentation d'images naturelles : coopération entre un détecteur contour et un détecteur région", actes du 11ème colloque GRETSI, p. 539-542, juin 1987.
- [Zamp 89] P. Zamperoni, "Model-based selective image enhancement by means of adaptive rank order filtering", Progress in Image Analysis and Processing, V. Cantoni, L.P. Cordella, S. Levialdi, G. Sanniti di Baja editors,

Proceedings of the 5th Int. Conf. on Image Analysis and Processing, septembre 1989.

- [Ziou 91] D. Ziou, "La détection des contours dans des images à niveaux de gris : mise en œuvre et sélection de détecteurs", Thèse de doctorat de l'Institut National Polytechnique de Lorraine, 1991.
- [Zuck 76] S.W. Zucker, "Survey Region growing: Childhood and adolescence", Computer graphics and image processing, vol. 5, p. 382-399, 1976.
- [ZuHa 88] O.A. Zuniga and R.M. Haralick, "Gradient threshold selection using the facet model", Pattern Recognition, vol. 21, n°5, p. 493-503, 1988.

ANNEXE 1

Pixels-contours de visibilité ambiguë

Points expérimentaux pour le type marche

(Cf chapitre 2, Ajustement de l'heuristique)

Nombre de points : 85

Abscisse : Valeur de EcartMin Max = 10.75 Min = 0.167

Ordonnée : Valeur du Gradient Max = 37 Min = 4

Points expérimentaux pour le type trait

(Cf chapitre 2, Ajustement de l'heuristique)

Nombre de points : 64

Abscisse : Valeur de EcartMin Max = 11.67 Min = 0.167

Ordonnée : Valeur du Gradient Max = 33 Min = 3.5

ANNEXE 2

Résultats de la détection des contours

Images et échantillons sélectionnés pour l'évaluation

(Cf chapitre 2, partie 5 : évaluation du détecteur de contours)

Evaluation des coefficients du gradient relatif

(Cf chapitre 2, partie 5 : évaluation du détecteur de contours)

Comparaison avec le détecteur de Deriche

(Cf chapitre 2, partie 5 : évaluation du détecteur de contours)

Résultats complets pour les six images tests

(Cf chapitre 2, partie 5, évaluation du détecteur de contours)

Contours de type marche : noir

Contours de type trait : gris

ANNEXE 3

Résultats de la coopération région-contour

Evaluation de la croissance de régions : utilité du type dégradé

(Cf chapitre 3, Evaluation du détecteur de régions, Evaluation de la coopération)

Evaluation des critères de la croissance de régions

(Cf chapitre 3, Evaluation du détecteur de régions : une image bruitée)

Evaluation de l'heuristique de la croissance de régions

(Cf chapitre 3, Evaluation du détecteur de régions)

Entre parenthèses sont indiqués : (seuilT = seuilH, %kT, %1-kT, %kH, %1-kH)

Résultats complets de la coopération pour deux images

(Cf chapitre 3, partie 4, évaluation de la coopération région-contour)