

République Algérienne Démocratique et Populaire  
Ministre de l'enseignement Supérieur et de la Recherche Scientifique  
Université Mentouri - Constantine  
Faculté des Sciences de L'Ingénieur  
Département d'Electronique

**Mémoire**

Présenté pour l'obtention du diplôme de magistère en électronique

**Option : Traitement du signal**

Thème :

*Développement d'une technique de maillage :  
application à la segmentation et au suivi d'objet*

***Dirigé par :***

*Pr. Mohammed KHAMADJA*

***Présenté par :***

*Amina BOUNAS*

***Devant le jury :***

***Président : Abdelhak BENNIA, Professeur UMC***

***Rapporteur : Mohammed KHAMADJA, Professeur UMC***

***Examineurs : Abelfatah CHAREF, Professeur UMC***

***Fella HACHOUF, MC UMC***

## Résumé

Dans ce travail, nous utilisons un algorithme de maillage actif qui place les arêtes du maillage sur les discontinuités de mouvement. On commençant avec un maillage grossier et régulier, notre algorithme le déforme pour créer la carte de segmentation. Le mouvement intérieur à chaque cellule est estimé par un algorithme de recherche exhaustive utilisant des blocs triangulaires de tailles différentes.

La déformation du maillage se fera principalement par la minimisation d'un terme d'énergie globale, ce dernier composé d'un terme d'énergie relatif au mouvement et un terme spatial.

## ملخص

في هذا العمل استخدمنا خوارزم شبكية نشيط. الوظيفة الأساسية لهذا الخوارزم هي تحريك أضلاع الشبكية بغرض الحصول على أضلاع الشبكية متزاوجة مع أضلاع الجسم المراد تجزئته و تتبع حركته. الخوارزم يبدأ بشبكية منتظمة ذات خلايا مثلثية ثم يقوم بتحريكها الى غاية الحصول على بطاقة تجزئة. الحركة الداخلية للخلايا المثلثية تحسب باستخدام خوارزم تقدير الحركة يعتمد على خلايا مثلثية متغيرة الشكل و الحجم.

تشويه و تحريك الشبكية يعتمد أساسا على تقليل الطاقة الإجمالية التي تتكون في الأصل من مجموع طاقتين: طاقة خاصة بالحركة و طاقة فضائية  
النتائج المتحصل عليها من خلال التطبيق على سلاسل حقيقية تثبت نجاعة الخوارزم الى حد كبير

## **Abstract**

In this work, we use a mesh-based video segmentation and object tracking algorithm. The algorithm moves the mesh nodes along motion or luminance discontinuities in order to fit the objects boundaries. The motion inside adjacent triangular cells is estimated by exhaustive matching algorithm using triangular bloc.

The mesh deformation is applied by minimization of an objective function, falling into two energy terms: a motion energy term and spatial boundary energy.

Experimental results with both synthetic and real-world sequences prove the efficiency of the proposed approach and the cooperative action of the two energy terms.

## *Remerciements*

*Je tiens à remercier vivement et respectueusement mon directeur de mémoire Professeur Mohammed KHAMADJA pour le choix du sujet, pour son temps précieux, son soutien moral et ses conseils qui m'ont aidé à étoffer ce travail*

*Je remercie sincèrement Dr. Saïd BEN YERBEH pour l'aide incommensurable qu'il m'a apporté et pour l'intéressante documentation qu'il m'a mise à ma disposition.*

*Sans omettre de remercier les membres du jury qui ont accepté de juger ce travail.*

- Prof. Abdelhak BENNIA Université de Constantine*
- Prof. Abelfatah CHAREF, Université de Constantine.*
- MC. Fella HACHOUF Université de Constantine.*

*A mes parents*

## Sommaire

<b>Introduction générale</b> .....	<b>1</b>
<b>Chapitre Introduction aux maillages</b> .....	<b>3</b>
1.1 Introduction.....	3
1.2 Définition générale d'un maillage .....	3
1.2.1 Définition au sens topologique .....	4
1.2.2 Définition au sens géométrique .....	5
1.3 Typologie proposée pour les différentes classes de maillages.....	5
1.3.1 Maillages réguliers .....	6
1.3.2 Maillages irréguliers .....	6
1.3.3 Maillages hiérarchiques .....	8
1.3.4 Maillages adaptatifs .....	9
1.4 Etat de l'art des maillages .....	9
1.5 Estimation de mouvement .....	10
1.5.1 Approche par mise en correspondance .....	11
1.5.2 Block-matching : mise en correspondance de blocs.....	13
1.5.3 Maillage dynamique .....	16
a) Grilles quadrangulaires .....	16
b) Maillages triangulaires .....	17
c) Maillages adaptés au contenu ou « maillages actifs » .....	17
d) Maillage objet.....	21
e) Approche par ligne de rupture .....	22
1.6 Conclusion .....	24
<b>Chapitre II Implémentation d'un algorithme de segmentation et de suivi d'objet</b> .....	<b>25</b>
2.1 Introduction.....	25
2.2 Stratégie de construction.....	25
2.3 Constructions de maillage.....	27
2.4 Déformation du maillage .....	29
2.4.1 Critère de mouvement.....	30
2.4.2 Critère spatial .....	35
2.5 Minimisation du critère d'énergie.....	38
2.6 Prédiction du maillage d'une image a l'autre .....	41

2.7 Conclusion .....	41
<b>Chapitre III Applications .....</b>	<b>42</b>
3.1 Introduction.....	42
3.2 Résultats .....	42
3.2.1 Estimation du mouvement .....	42
3.2.2 Segmentation.....	45
3.2.2.1 Images synthétiques .....	45
3.2.2.2 Images réelles.....	47
3.2.3 Suivi d'objet.....	49
3.3 Conclusion .....	55
<b>Conclusion générale .....</b>	<b>55</b>
<b>Bibliographie .....</b>	<b>56</b>

# Liste des figures

Figure 1.1 : a: grille quadrilatère; b, c, d: grilles triangulaire ; e: grille hexagonale.....	5
Figure 1.2 : diagramme de Voronï .....	7
Figure 1.3: triangulation de Delaunay associée et diagramme de Voronï .....	8
Figure 1.4: les deux méthodes pour obtenir une représentation hiérarchique.....	9
Figure 1.5: Block-matching avec : a. prédiction forward, b : prédiction backward.....	12
Figure 1.6 : illustration d'un bloc de taille $8 \times 8$ pixels.....	13
Figure 1.7 : Algorithme de recherche exhaustive .....	15
Figure 1.8 : exemple de grille active (quadrangles).....	16
Figure 1.9 : Mise en correspondance par maillage triangulaire .....	17
Figure 1.10 : connectivité régulière/connectivité irrégulière .....	18
Figure 1.11 : géométrie rigide/géométrie adaptative .....	19
Figure 1.12 : approche « quadtree » .....	19
Figure 1.13 : approche « quadtree » par maillage triangulaire.....	20
Figure 1.14 : approche « quadtree » + déformation .....	20
Figure 1.15 : approche par insertion de nœuds .....	21
Figure 1.16 : Exemple de maillage objet.....	21
Figure 1.17 : Trous et collisions engendrés par les mouvements différents des objets présents dans la scène.....	22
Figure 1.18 : La notion de lignes de rupture sur un maillage.....	23
Figure 2.1 : l'approche par insertion de nœuds : un pixel a fort gradient spatial situé dans une lacune .....	26
Figure 2.2 : maillage à base de quadrangles Les nœuds du maillage initial (a) sont déplacés ; Une fois la déformation effectuée (b) le maillage n'est plus quadrangulaire (c) .....	26
Figure 2.3 : Les nœuds du maillage initial (a) sont déplacés. Une fois la déformation effectuée, des cellules sont subdivisées et des nouveaux nœuds sont créés (c). Les nœuds marqués en noir (e) sont alors figés, car la zone grisée ne serait plus quadrangulaire s'ils sont déplacés (d) .....	27
Figure 2.4: Echange de diagonale d'un quadrilatère convexe selon le critère de la boule vide... ..	28
Figure 2.5 : Exemple de maillage de Delaunay ne respectant pas la frontière .....	28
Figure 2.6 : minimisation du critère d'énergie .....	29

Figure 2.7 : Illustration du concept de fond couvert et découvert.....	31
Figure 2.8 : Modèle de translation du mouvement.....	32
Figure 2.9 : choix de la taille du bloc : a) bloc de grand taille, b) bloc de petite taille .....	33
Figure 2.10 : parcours dans un bloc dans l'image courant Les points à l'extérieur du triangle n'entrent pas dans les calculs .....	34
Figure 2.11: maillage déformé uniquement par prise en compte d'un critère de mouvement.....	35
Figure 2.12 : détermination de $r(e_j)$ pour chaque arête $e_j$ .....	36
Figure 2.13 : filtrage et carte de distance .....	37
Figure 2.14 : filtrage de l'image « Lena ».....	38
Figure 2.15 : Operation de marquage sur un plan alpha .....	40
Figure 3.1 : résultat de l'estimation du mouvement .....	43
Figure 3.2 : comparaison entre l'estimation du mouvement avec maillage régulier et non régulier .....	45
Figure 3.3 : segmentation d'un rectangle avec maillage actif .....	46
Figure 3.4 : segmentation d'un cercle et d'un triangle avec maillage actif .....	46
Figure 3.5 : résultat de segmentation sur « foreman » .....	47
Figure 3.6 : résultat de segmentation sur « coastguard » .....	48
Figure 3.7: Évolution du critère d'énergie en fonction des itérations.....	49
Figure 3.8: suivi d'objets sur la séquence « foreman » .....	50
Figure 3.9: segmentation automatique et suivi sur la séquence « foreman » .....	51
Figure 3.10 : suivi d'objets sur la séquence « Coastgard » .....	52
Figure 3.11 : suivi de deux objet sur la séquence « Coastgard » .....	53

## Introduction générale

Le concept de représentation par maillage est relativement récent en codage d'image fixe ou vidéo, alors que ce type de représentation est exploité depuis bien longtemps pour la modélisation et l'analyse de structures par l'intermédiaire en particulier de la méthode des éléments finis. La méthode des éléments finis qui consiste à diviser un système physique au comportement complexe en composantes simples au nombre de paramètres limité pour simuler, par leur juxtaposition le comportement d'un système complet. De là est née l'idée de représenter une image en procédant à une approximation par morceaux par l'assemblage de facettes, chacune remplie d'une texture appropriée. Ainsi, même si ces petites briques prises indépendamment ne permettent pas d'expliquer ou de rendre un comportement général, leur assemblage dans une structure globale permet de reproduire une structure au comportement beaucoup plus complexe.

Dans le domaine du traitement d'image, une grande branche s'occupe de l'*analyse d'image* ou *analyse de scène*. La segmentation n'est généralement pas simple et les procédés de segmentation sont fort nombreux.

D'une manière générale, l'utilisateur se voit contraint de construire un procédé de segmentation le plus adapté au problème qu'il traite. Il faut qu'il garde à l'esprit qu'une même image peut subir plusieurs segmentations toutes aussi valables les unes que les autres.

D'autre part, Parmi les problèmes qu'il est nécessaire de résoudre lors de l'analyse des scènes, le suivi d'objet occupe une place prépondérante. Lorsque les objets à suivre sont de nature non rigide, l'utilisation de modèles déformables est préconisée.

Nous développons dans ce travail une approche maillage 2D pour la segmentation et le suivi d'objets dans une séquence vidéo ;

Le document est structuré comme suit :

Un état de l'art est exposé dans le premier chapitre. Une présentation des différents type de maillage et des notions théorique détaillés concernant la triangulation de Delaunay et le maillage actif sujet de notre recherche.

Le chapitre deux présente les approches choisies et les choix effectués pour la construction du maillage actif, ainsi que l'algorithme proposé pour la segmentation et la poursuite d'objet vidéo. Cependant le problème de segmentation automatique reste un problème ouvert, nous proposons dans cette partie l'utilisation du maillage triangulaire dynamique pour la segmentation et le suivi d'objet en réalisant une segmentation automatique ou semi-automatique.

L'algorithme proposé construit un maillage actif, dont les arêtes sont placées sur les discontinuités de mouvement et les discontinuités spatiales, pour séparer au mieux les objets présents dans la scène. Une estimation de mouvement utilisant des blocs triangulaires de différentes tailles réduit l'erreur de prédiction.

Le troisième et dernier chapitre est consacré à la représentation des résultats obtenus par l'implémentation de l'algorithme, ce dernier est évalué par des séquences réelles.

## **Chapitre I**

# **Introduction aux maillages**

## 1.1 Introduction

Le concept de représentation par maillage est relativement récent en codage d'image fixe ou vidéo, alors que ce type de représentation est exploité depuis bien longtemps pour la modélisation et l'analyse de structure par l'intermédiaire en particulier de la méthode des éléments finis. La méthode des éléments finis, qui consiste à diviser un système physique au comportement complexe en composantes simples au nombre de paramètres limité pour simuler, par leur juxtaposition le comportement d'un système complet. De là est née l'idée de représenter une image en procédant à une approximation par morceaux par l'assemblage de facettes, chacune remplie d'une texture appropriée. Ainsi, même si ces petites briques prises indépendamment ne permettent pas d'expliquer ou de rendre un comportement général, leur assemblage dans une structure globale permet de reproduire une structure au comportement beaucoup plus complexe.

## 1.2 Définition générale d'un maillage

Le maillage est une opération de génération d'une grille 2D de l'image par définition d'une partition, en éléments polygonaux, basée sur des caractéristiques homogènes pour la représentation réduite des données de cette image. Le but de cette opération est que l'image approchée par cette représentation soit la plus fidèle possible à l'image originale. Les sommets des polygones sont appelés nœuds et ces nœuds sont reliés entre eux par des arcs.

En général les difficultés rencontrées au cours de la génération d'une grille sont de deux types :

- la création de l'ensemble des nœuds (position et nombre).
- La création de la partition (le type de découpage).

Ce dernier point peut être de nature purement géométrique ou alors dépendre de données initiales de manière à minimiser certains critères.

Etant donné un nuage de points d'un plan, un maillage polygonal de ce domaine est un recouvrement de ce dernier par des éléments de nature géométrique élémentaire (triangle, quadrilatère, ...), sous-ensembles des points de ce plan et tel que certaines propriétés soient vérifiées. Diverses topologies de maillages de référence sont généralement utilisées : le maillage peut être uniforme, quadrangulaire, de Delaunay, etc. D'une manière plus générale, la géométrie fait référence aux positions 2D des

points nœuds du maillage, tandis que la topologie indique les connections entre les nœuds, c'est-à-dire la structure des éléments (ou "patches"). Ces définitions topologiques et géométriques coexistent et le choix de l'une ou l'autre dépend du domaine d'application : géométrie algorithmique, image, élément finis, etc. Puisque on fait du traitement d'image, nous utiliserons indifféremment dans la suite du manuscrit les termes triangulation, maillage ou grille. Les maillages seront par conséquent définis sur une seule image ou alors suivis et déformés au cours du temps. On sous-entendra par maillage une représentation continue.

### 1.2.1 Définition au sens topologique

Du point de vue géométrie algorithmique, il existe une distinction entre maillage et triangulation [1]. Une triangulation est un recouvrement de l'enveloppe convexe d'un nuage de points (supposés connus), alors qu'un maillage est un recouvrement d'un domaine donné, en général via une discrétisation de sa frontière. Afin de le définir d'une manière simple, celui-ci est décrit par l'ensemble de points-nœuds reliés entre eux par des arêtes.

La représentation par maillage d'un domaine 2D est une partition du domaine avec un nombre fini d'éléments polygonaux disjoints. Chaque élément est défini par un certain nombre de sommets ou nœuds.

Suivant les méthodes de division adoptées, différentes sortes de maillages apparaissent : triangulaire, hexagonal (figure 1.1.e), quadrilatère, etc. Nous ne considérerons ici que des maillages pour lesquels tous les éléments ont le même nombre de nœuds. Notons par  $K$  ce nombre (Alors,  $K=3$  pour le maillage triangulaire,  $K=4$  pour le maillage quadrilatère)

- *maillage quadrilatère* : Pour un nombre donné de nœuds dans les directions verticales et horizontales il n'y a topologiquement qu'une seule manière de créer un maillage quadrilatère et tous les nœuds à l'exception de ceux se trouvant sur les bords, seront 4-connexes avec d'autres nœuds (figure 1.1.a).

- *maillage triangulaire* : du fait de l'asymétrie inhérente des éléments triangulaire, il existe plusieurs façons pour construire un tel maillage. Il peut simplement être généré à partir d'un maillage quadrilatère en divisant chaque quadrangle en deux triangles.

En plus, suivant le découpage il existe de multiples solutions pour effectuer une triangulation :

- dans une seule direction (figure 1.1.b), dans ce cas tous les nœuds sont 6-connexes)

- en alternant deux orientations possibles (figure 1.1.c), ici même si les nœuds sont 4-connexes ou 8-connexes, ce maillage est généralement plus approprié que le précédent pour des images avec des caractéristiques dans des directions variées.

- en quinconces (figure 1.1.d) : compromis entre les deux maillages précédents puisqu'il est bidirectionnel et 6-connexe.

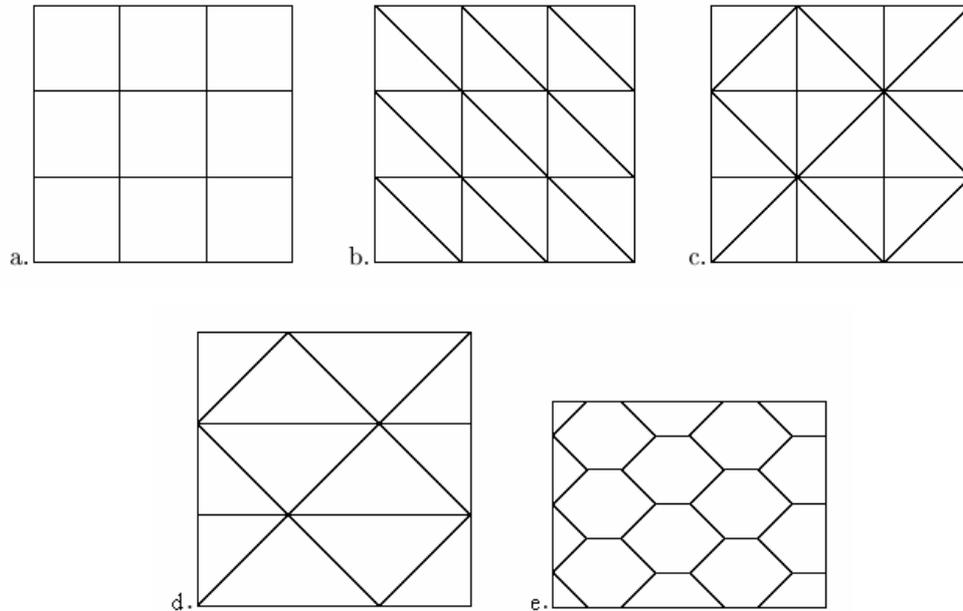


Figure 1.1 : a: grille quadrilatère; b, c, d: grilles triangulaire ; e: grille hexagonale

### 1.2.2 Définition au sens géométrique

La définition au sens géométrique consiste à définir un "ensemble de règles de création" c'est à dire à définir le type de grille que l'on veut : grille régulière ou irrégulière, forme géométrique des facettes, etc.

### 1.3 Typologie proposée pour les différentes classes de maillages

Cette section, présente de façon condensée les principaux découpages utilisés actuellement en tentant de les regrouper sous différentes grandes appellations.

Afin de conserver la continuité entre les éléments ; un nœud d'un élément ne peut pas se trouver sur l'arête d'un autre élément sur un même maillage. En satisfaisant cette contrainte le maillage est dit conforme (les maillage présentés dans les figures précédentes vérifient cette propriété), dans le cas contraire le maillage est dit non-conforme (maillage quadtree par exemple).

### 1.3.1 Maillages réguliers

L'avantage d'un tel maillage réside dans la simplicité de sa définition, et dans le fait que les positions des points d'une cellule (triangle ou quadrilatère) sont identiques d'une cellule à une autre, à un facteur de translation ou de symétrie près. Par conséquent, ces positions peuvent être mémorisées et permettre un balayage rapide des points de l'image.

L'inconvénient de ce maillage est sa non-flexibilité : par exemple, les arêtes du découpage ont fort peu de chances de coïncider avec les contours (gradients spatiaux ou temporels respectivement pour une image fixe ou une séquence vidéo), ce qui rends ces maillages uniformes souvent inappropriés pour refléter des zones voisines de ces contours, compliquant par conséquent l'exactitude du rendu des scènes.

D'un point de vue suivi d'objets vidéo un seul patch peut contenir des mouvements multiples mais cette unique "facette" ne pourra pas rendre fidèlement ces différents mouvements. Ce problème peut se résoudre par un découpage plus fin de cette partie ou par un repositionnement ce qui résulte en un maillage irrégulier.

### 1.3.2 Maillages irréguliers

Une partition irrégulière permet un choix non contraint du nombre de sommets et leur concentration dans des zones d'intérêt mais aussi un placement libre de façon à être plus précis ou plus adapté au contenu.

Dans ce cas, différentes solutions de découpages apparaissent : triangulation de Delaunay, quadrilatère, hexagonale, panaché de polygones, par pavage de Voronoï,...

La triangulation de Delaunay présente d'attrayants critères géométriques. La triangulation de Delaunay est la plus appropriée pour les approximations de surface, car elle produit des triangles aussi équiangulés que possible [2].

#### Triangulation irrégulière de Delaunay

La triangulation de Delaunay est une représentation graphique pour une fonction de  $\mathbf{R}^2$  irrégulièrement échantillonnée. Elle donne une décomposition de l'enveloppe convexe d'une surface  $S$  en polygones convexes, inscriptibles dans des cercles, qui sont tels que le cercle circonscrit à un polygone ne contient aucun autre sommet de  $S$ .

Pour construire un maillage, nous avons besoin de connaître [3] :

- Un ensemble de points

$$S = \overset{\text{def}}{\{x^i \in \mathbb{R}^2 / i \in \{1, \dots, N_p\}\}}$$

- Un ensemble d'arêtes (couples de numéros de points) définissant la maillage de la frontière  $\Gamma_h$  des sous domaines.

$$A = \overset{\text{def}}{\{(sa_1^j, sa_2^j) \in 1, \dots, N_p^2 / j \in \{1, \dots, N_a\}\}}$$

- Un ensemble de sous domaines (composantes connexes de  $\mathbb{R}^2 \setminus \Gamma_h$ ) à mailler, avec l'option par défaut suivante : mailler tous les sous domaines bornés. Les sous domaines peuvent être définis par une arête frontière et un sens (le sous domaine est à droite (-1) ou à gauche (+1) de l'arête orientée). Formellement, nous disposons donc de l'ensemble

$$SD = \overset{\text{def}}{\{(a^i, sens^i) \in \{1, \dots, N_a\} \times \{-1, 1\} / i = 1, N_{sd}\}}$$

Qui peut être vide (cas par défaut).

La méthode est basée sur les diagrammes de Voronoï :

Les diagrammes de Voronoï sont les polygones convexes  $V^i, i=1, N_p$  formés par l'ensemble des points de  $\mathbb{R}^2$  plus proche de  $x^i$  que des autres points  $x^j$  (figure 1.2)

$$V^i = \overset{\text{def}}{\{x \in \mathbb{R}^2 / \|x - x^i\| \leq \|x - x^j\|, \forall j \in \{1, \dots, N_p\}\}}$$

Les arêtes d'un tel diagramme sont en fait des portions de médiatrices entre deux points de  $S$ , les sommets sont des centres de cercle passant par trois points.

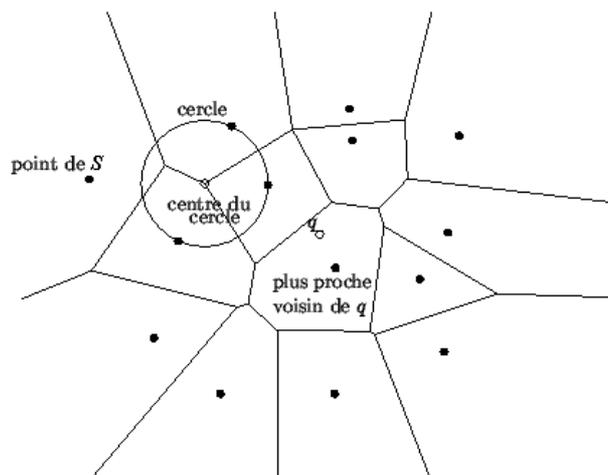
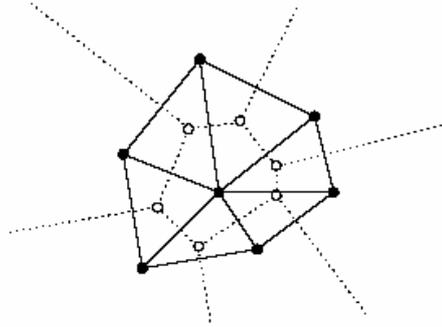


Figure 1.2 : diagramme de Voronoï

Le dual du diagramme de Voronoï est la triangulation de Delaunay (figure 1.3). On l'obtient en reliant les points dont les cellules de Voronoï sont voisines. On obtient ainsi une triangulation de  $S$ , c'est-à-dire un maillage triangulaire ayant les points de  $S$  pour sommets. La triangulation de Delaunay de  $S$  est telle qu'aucune de ses faces ne contient un autre point de  $S$  à l'intérieur de son cercle circonscrit.



**Figure 1.3: triangulation de Delaunay associée (traits pleins) et diagramme de Voronoï (en pointillés)**

En effet, l'utilisation d'un maillage irrégulier donnera des résultats comparables, pour une gestion plus aisée, car le maillage ainsi construit possède moins de cellules que le maillage régulier.

### 1.3.3 Maillages hiérarchiques

Les maillages hiérarchiques ont une structure de données établie de telle manière qu'il est possible de reconstruire des images à différentes résolutions. Ils sont utilisés aussi bien pour représenter des images fixes et vidéo, que dans des applications de synthèse d'image ou de vision par ordinateur (reconnaissance d'objet, représentation à différents niveaux de détail...).

Cette hiérarchie, source de l'emboîtement des partitions a plusieurs intérêts :

- elle rend possible une représentation et reconstruction progressives,
- la structure générale n'est pas totalement remise en cause au moindre ajout ou retrait de nœud(s).
- elle autorise quelques fonctionnalités nouvelles au niveau de la manipulation des images telles que les zooms avec niveaux de détails, etc.

La notion d'emboîtement peut faire référence à l'emboîtement des patches comme à l'emboîtement des nœuds par leur réutilisation.

Il existe deux méthodes (figure 1.4) pour effectuer une pyramide hiérarchique sur un maillage :

- par division/fusion régulière des facettes,
- par ajout/retrait de noeuds et retriangulation de Delaunay.

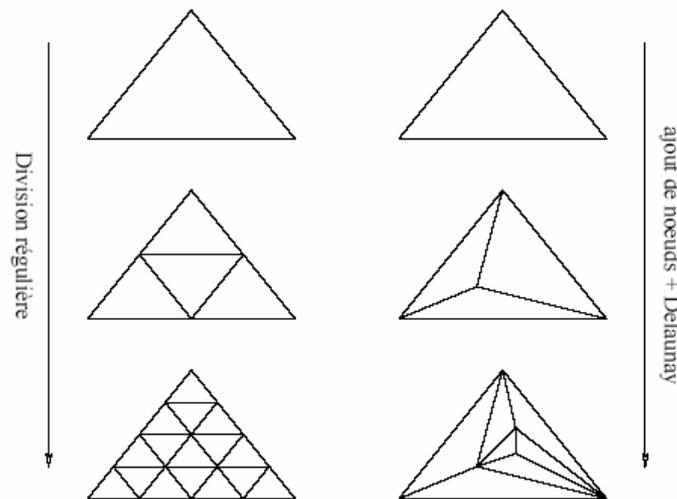


Figure 1.4: les deux méthodes pour obtenir une représentation hiérarchique: par division/fusion régulière des facettes (à gauche), par ajout/retrait de nœuds et retriangulation (à droite)

### 1.3.4 Maillages adaptatifs

Un maillage est adaptatif lorsqu'il tient compte du contenu de l'image représentée. Ceci intervient par une densité de nœuds, et par conséquent de facettes, plus ou moins importante suivant les zones de l'image à mailler.

Un maillage adaptatif peut être régulier adaptatif ou irrégulier adaptatif : cela dépend de la façon dont il est construit c'est à dire de la forme des facettes qui le constituent.

## 1.4 Etat de l'art des maillages

Les techniques de génération de maillages sont employées de manière intensive dans de nombreux domaines, de ce fait, de nombreuses études ont montrées que les techniques utilisant les maillages offrent de bonnes performances ainsi que des fonctionnalités intéressantes.

Citant quelques domaines d'utilisation :

- La compression [4] sans pertes des maillages, l'exploitation des algorithmes de subdivision qui permettent la simplification des maillages triangulaires quelconques à l'aide de la transformée en ondelettes.

- Le mappage de texture [5]. La segmentation pourra isoler certaines parties de l'objet qui pourront être traitées avec des outils spécifiques d'application de texture.
- La paramétrisation de maillage [6], [7] : Paramétrer un maillage consiste à le décomposer en disques topologiques (c'est-à-dire en parties dépliables sans recouvrement et sans trou) et à mettre ces parties en correspondance avec un plan (ou une autre forme).
- Le remaillage et la simplification [8], [9]. Il s'agit de la reconstruction ou de la simplification d'un maillage dans le but d'obtenir un nouveau maillage, de complexité moins importante par exemple.
- La détection de collisions [10]. Les jeux vidéo et bien d'autres applications nécessitent de détecter les collisions entre les objets 3D. Les contraintes de temps réel impliquent une réaction très rapide et la comparaison de l'ensemble des faces des objets 3D représente un coût relativement élevé. Il est envisageable de détecter les collisions, non pas à partir des objets originaux, mais plutôt à partir de modèles simplifiés issus de la segmentation.

Cependant, notre étude visera le suivi d'objets vidéo par maillage, et pour obtenir des contours précis, une segmentation doit être effectuée.

Pour que l'extraction et le suivi des objets en mouvement soit efficace, des techniques performantes d'estimation de mouvement doivent être mise en œuvre associée à un choix d'une méthode de maillage qui sera le maillage déformable, et nous reviendrons à l'explication de ce choix dans les sections qui suivent.

## **1.5 Estimation de mouvement**

L'estimation du mouvement est un problème incontournable dans le domaine du traitement des séquences d'images. Le but est d'estimer le mouvement des objets contenus dans une vidéo sous forme d'un champ dense de vecteurs.

L'estimation de mouvement est excessivement coûteuse en temps de calcul. Cela peut prendre jusqu'à 80% des ressources matérielles de la machine. Une estimation de faible qualité est souvent source de problèmes importants comme par exemple des effets de blocs visuellement gênants. Ainsi, depuis plus de vingt ans, son étude reste un sujet de recherche important.

### 1.5.1 Approche par mise en correspondance

L'estimation de mouvement par mise en correspondance est l'approche la plus généralement employée. Elle est basée sur l'utilisation directe de la contrainte de conservation de l'intensité:

$$I(x, y, t) = I(x + dx, y + dy, t \pm 1) \quad (1.1)$$

L'estimation du champ de vecteurs mouvements dans l'image au temps  $t$  revient ici à essayer d'estimer l'image au temps  $t$  à partir de l'image au temps  $t-1$ . On parle alors d'estimation arrière (backward matching). Ou à partir de l'image au temps  $t+1$  pour l'estimation avant (forward matching). L'image au temps  $t$  est découpée en un ensemble de  $n$  régions  $R_i$ , auxquelles on affecte un mouvement propre modélisé par une transformation  $\tau_i$ . L'estimation de mouvement des différentes régions s'effectue en minimisant l'erreur de compensation de mouvement au temps  $t$  (appelé DFD pour Displaced Frame Difference):

$$DFD_i(x, y) = I(x, y, t) - I(x + dx, y + dy, t \pm 1) \quad (1.2)$$

Ce qui se traduit globalement par la minimisation de l'énergie  $E_{comp}$ :

$$E_{comp} = \sum_{i=1}^n \left( \sum_{p(x,y) \in R_i} (I(p, t) - I(\tau_i(p), t \pm 1)) \right) \quad (1.3)$$

Il faut alors définir plusieurs paramètres pour le calcul de cette énergie :

- Le terme d'erreur  $e(u)$ . Les deux formes les plus généralement employées sont l'erreur quadratique  $e_{quad}(u) = u^2$  et l'erreur absolue  $e_{abs}(u) = |u|$ .

L'emploi de l'erreur absolue se justifie souvent par sa complexité calculatoire moins importante.

- Le type de transformation  $\tau_i$  que l'on désire utiliser pour modéliser le mouvement affecté aux différentes régions  $R_i$ . Il peut être entre autres de type constant, affine ou polynomial. Le champ de mouvement alors estimé n'est pas forcément constitué de valeurs entières, et le calcul de  $I(\tau_i(p), t \pm 1)$  nécessite l'utilisation d'algorithmes d'interpolations.

- Le partitionnement de l'image. Chaque région  $R_i$  se voit associé un type de mouvement unique. Le choix du découpage de l'image en régions est donc très important quant à la qualité de l'estimation du mouvement : il faudra veiller à ce que chaque région ne

possède réellement un mouvement représentable par un jeu de paramètres uniques. Les discontinuités du champ réel de mouvement ne seront donc correctement estimées que si elles sont situées aux frontières entre les régions.

- Le choix de la direction de l'estimation (avant ou arrière, «forward» ou «backward» figure 1.1.5). Pour l'estimation de mouvement seule, les deux approches ont autant d'avantages que d'inconvénients. Pour la compression de séquences vidéo, l'estimation arrière est utilisée de préférence, car elle permet de prédire tous les pixels de l'image  $t+1$  à partir de l'image  $t$ . On parlera dans ce cas de compensation de mouvement.

La minimisation de l'énergie de compensation  $E_{comp}$  engendre nécessairement l'estimation du champ de mouvement dans l'image selon un partitionnement en régions  $R_i$ . Si ces régions sont fixes on dira que la méthode est rigide, dans le sens où le partitionnement est défini à l'avance, sans aucune prise en compte de la distribution du mouvement dans l'image.

Si l'on estime le champ de mouvement et on construit un partitionnement adapté au contenu de l'image, ce qui dans ce cas revient à une segmentation de l'image par le mouvement, la méthode sera dite adaptative.

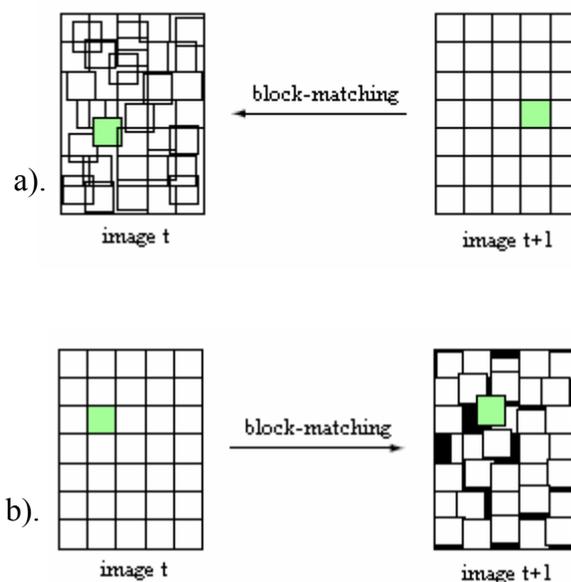


Figure 1.5: Block-matching avec : a. prédiction forward, b : prédiction backward

De nombreuses méthodes de partitionnement de l'image pour la mise en correspondance ont été développées. La plus simple est la mise en correspondance de blocs.

Les sections suivantes détaillent quelques méthodes d'estimation de mouvement par mise en correspondance.

### 1.5.2 Block-matching : mise en correspondance de blocs

#### a. Principe

Le principe général du block-matching est d'exploiter les redondances temporelles existant entre des images consécutives. Pour cela, considérons une séquence vidéo dans laquelle nous voulons estimer le mouvement des différents objets qui y sont contenus. Pour simplifier l'estimation, nous ne considérons que le mouvement présent entre deux images successives : l'image courante et l'image de référence. Chaque image est subdivisée en blocs de taille égale (généralement de  $8 \times 8$  ou  $16 \times 16$  pixels) et chaque bloc est considéré comme étant un objet indépendant (voir figure 1.5). Nous faisons l'hypothèse que le mouvement des pixels est uniforme à l'intérieur d'un bloc.

De manière informelle, l'algorithme consiste, pour un bloc de l'image courante, à choisir un bloc dans l'image de référence (passée ou future) et à calculer un critère comparaison entre ces deux blocs. L'opération est répétée en choisissant un autre bloc jusqu'à ce que tous les blocs d'une zone déterminée de l'image de référence (appelée < fenêtre de recherche >) aient été testés ou jusqu'à un critère arrêt arbitraire. Le bloc le plus semblable est ainsi identifié dans l'image de référence pour chaque bloc de l'image courante.

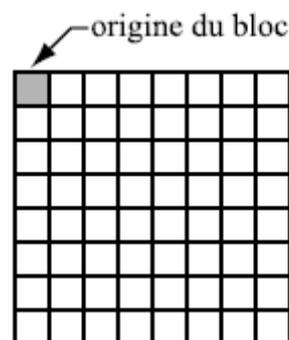


Figure 1.6 : illustration d'un bloc de taille  $8 \times 8$  pixels

Nous obtenons de cette manière pour tout bloc un vecteur déplacement caractéristique du mouvement de ce dernier. L'utilisation d'une fenêtre de recherche permet de limiter

le nombre de blocs de référence que l'algorithme doit tester. La taille de la fenêtre dépend d'un déplacement maximal autorisé fixé par l'utilisateur.

Initialement, on peut vouloir placer l'origine d'un bloc en son centre. Cependant, il est beaucoup plus astucieux de le situer dans le coin supérieur gauche (voir figure 1.6) pour deux raisons : premièrement, les blocs ont en général des cotés de taille paire ce qui implique qu'il n'y a pas de pixel central ; deuxièmement, les tests d'appartenance à la fenêtre sont moins coûteux en temps de calcul avec cette convention.

Le nombre et l'ordre dans lequel les blocs sont testés ainsi que les conditions d'arrêt ont des influences majeures sur l'efficacité de l'algorithme de recherche.

### **b. Critères de comparaison**

Le « critère de comparaison » précédemment mentionné pour trouver le bloc correspondant est une mesure objective de la dissemblance entre les valeurs des pixels contenus dans chaque bloc. Il en existe plusieurs qui prennent en compte ou non la couleur.

De manière générale, la plupart des articles traitant de l'estimation de mouvement considèrent que les blocs sont caractérisés par la luminance. Ceci est dû au fait que l'œil humain est davantage sensible à l'intensité lumineuse qu'à la couleur.

### **c. Parcours du voisinage**

Il existe de nombreuses méthodes de block-matching cherchant à optimiser l'efficacité et la rapidité de l'algorithme [11], [12].

Rappelons qu'étant donné un bloc dans l'image courante, l'objectif est de trouver le bloc dans l'image précédente ou suivante (selon le type de prédiction utilisé) qui lui correspond au mieux.

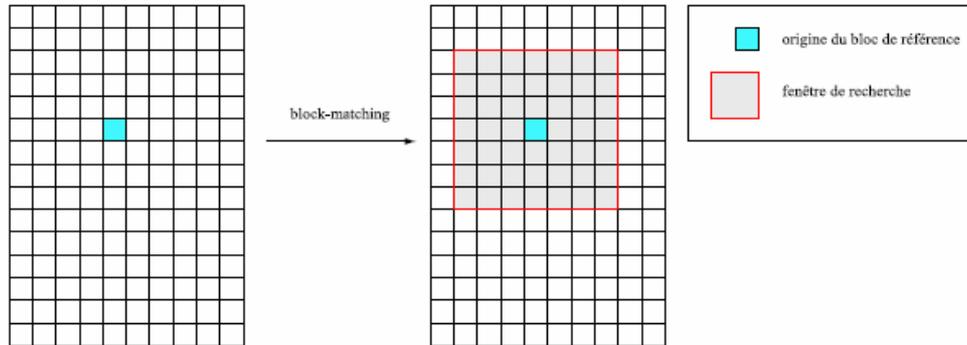
Diverses méthodes sont largement utilisées. Ces méthodes nécessitent l'estimation d'un vecteur à deux dimensions pour chaque bloc. Plusieurs approches sont envisageables pour la détermination du vecteur mouvement d'un bloc.

La plus simple est la recherche exhaustive du vecteur donnant la meilleure compensation parmi un ensemble discret de vecteurs, par exemple l'espace des paires d'entiers  $[-16, 16] \times [-16, 16]$ , nous dirons que cette méthode est optimale [13].

### **Algorithme de recherche exhaustive - Full Search Algorithm**

Le block-matching "Full Search" consiste à parcourir de manière exhaustive l'ensemble de la fenêtre de recherche. En procédant de cette manière, le bloc retourné par

l'algorithme sera celui qui minimise le critère de comparaison. Pour simplifier la compréhension de l'algorithme visible sur la figure suivante, un bloc est identifié par son origine.



**Figure 1.7 : Algorithme de recherche exhaustive**

La fenêtre de recherche est ici définie par un déplacement  $[-3,3] \times [-3,3]$  c'est-à-dire que le déplacement maximal autorisé est de trois dans toutes les directions. Habituellement, le déplacement maximal est de sept mais aucune norme n'est définie. En effet, un déplacement maximal petit pénalise les grands déplacements et augmente naturellement l'erreur de prédiction dans le cadre d'une vidéo présentant de tels déplacements.

Cependant, peu de blocs seront testés et l'algorithme sera alors rapide. A l'inverse, un grand déplacement autorisé permet de donner plus de liberté aux vecteurs mouvement mais augmente de manière quadratique le nombre de blocs testés. Il y a donc un compromis entre qualité et rapidité de l'algorithme. D'autre part, dans certains cas, contraindre le déplacement est une façon d'ajouter un a priori sur le déplacement recherché.

La mise en correspondance de blocs présente de nombreux avantages : son partitionnement est régulier, sa complexité est relativement faible (2 paramètres à estimer par cellule) ; son rapport qualité de compensation/coût de codage l'a rendu très populaire pour la compression de séquences vidéo.

Et si nous classons les différentes méthodes de block-matching selon un ordre décroissant du PSNR (Peak Signal to Noise Ratio), la méthode de recherche exhaustive sera la première ; il est évident que toute méthode rapide ne peut surpasser qualitativement la méthode exhaustive [14].

### 1.5.3 Maillage dynamique

#### a) Grilles quadrangulaires

Le mouvement de l'image est estimé à partir du déplacement des points de contrôle (ou noeuds) formant une grille uniforme. Cette grille partitionne l'image en quadrangles.

La grille est construite sur l'image au temps  $t$  (figure 1.8), le déplacement des noeuds est effectué sur l'image au temps  $t+1$  afin de diminuer l'erreur de compensation de mouvement.

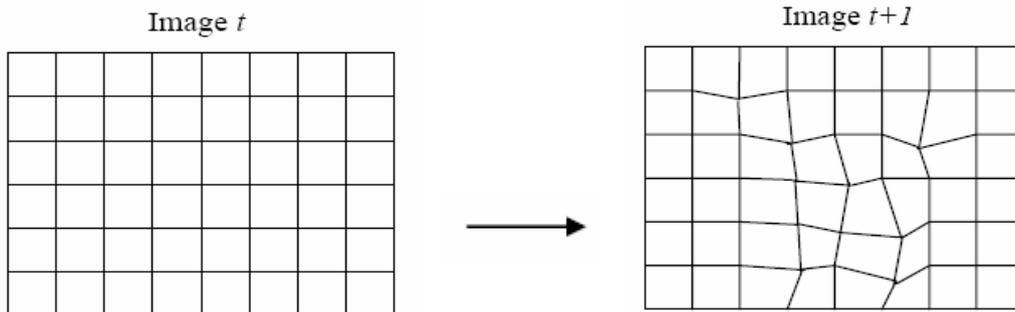


Figure 1.8 : exemple de grille active (quadrangles)

Le mouvement de chaque pixel est déterminé par interpolation du mouvement des noeuds des points de contrôle autour du pixel considéré. Chaque région possède alors son modèle de mouvement, déterminé par le mouvement des points de contrôle. Plusieurs modèles d'interpolation (reflétant le modèle du mouvement) peuvent être envisagés :

Le modèle bilinéaire :

$$p(x, y) \xrightarrow{\tau_i} p'(x', y') / \begin{cases} x' = a_1x + a_2y + a_3xy + a_4 \\ y' = a_5x + a_6y + a_7xy + a_8 \end{cases}$$

Le modèle perspective :

$$p(x, y) \xrightarrow{\tau_i} p'(x', y') / \begin{cases} x' = \frac{a_1x + a_2y + a_3}{a_7x + a_8y + 1} \\ y' = \frac{a_4x + a_5y + a_6}{a_7x + a_8y + 1} \end{cases}$$

Les grilles actives présentent l'avantage de représenter une gamme de mouvements plus riche que l'algorithme de mise en correspondance de blocs, qui ne gère correctement que les translations.

Néanmoins, le mouvement estimé est continu, ce qui pose problème lorsque des discontinuités de mouvement apparaissent dans une séquence (ce qui est le cas dans pratiquement toutes les séquences naturelles). Pour résoudre ce problème, Nosratinia [15] propose un modèle de mouvement adaptatif en fonction de la scène, optimisant la compensation de mouvement par un noyau d'interpolation du mouvement des noeuds variable avec la séquence. Cependant, cette méthode d'optimisation est effectuée sur la globalité de l'image et ne tient donc pas compte du fait que les discontinuités apparaissent localement dans l'image.

### b) Maillages triangulaires

La méthode de mise en correspondance par maillages triangulaire est similaire à l'approche de mise en correspondance par grille, à la différence près que les régions formées sont triangulaires au lieu d'être rectangulaires (figure 1.9).

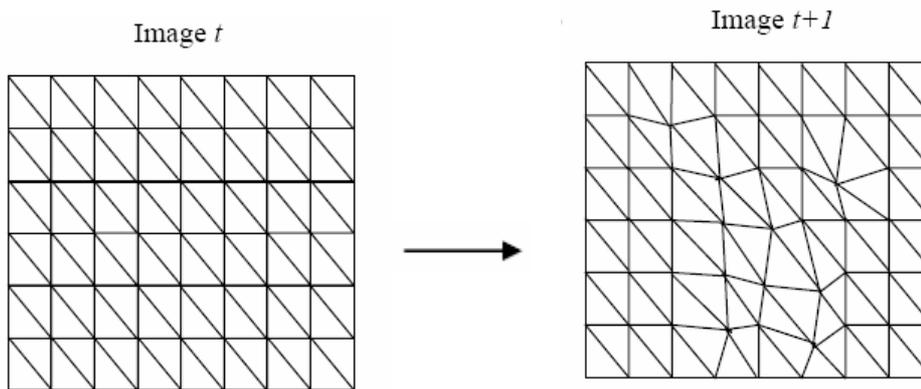


Figure 1.9 : Mise en correspondance par maillage triangulaire

Le modèle d'interpolation à partir du mouvement des noeuds devient alors un modèle affine à 6 paramètres :

$$p(x, y) \xrightarrow{\tau_i} p'(x', y') / \begin{cases} x' = a_1x + a_2y + a_3 \\ y' = a_4x + a_5y + a_6 \end{cases}$$

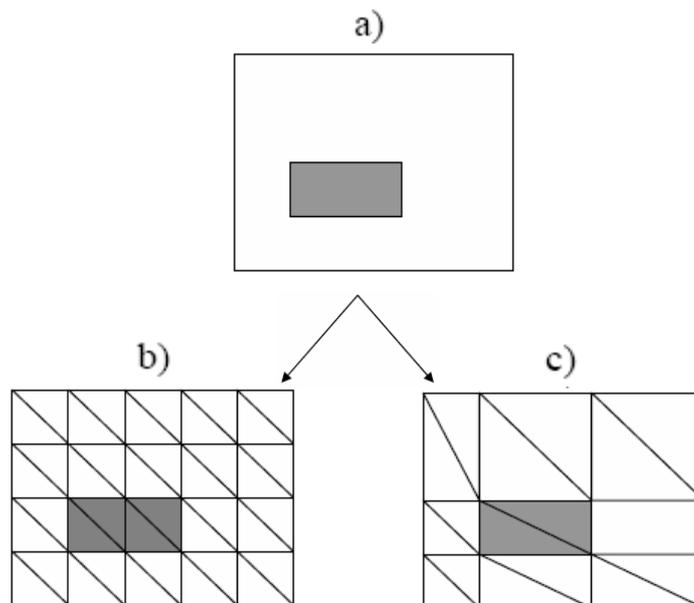
### c) Maillages adaptés au contenu ou « maillages actifs »

Les deux méthodes citées précédemment (maillages triangulaires et quadrangulaires) possèdent l'avantage de représenter un mouvement riche, mais leur structure est rigide sur l'image au temps  $t$ . Ainsi, l'idée d'utiliser un *maillage actif* est apparue, afin d'adapter le maillage à l'image sur laquelle il est plaqué, ceci pour deux raisons :

- Le mouvement au temps  $t$  d'une séquence donnée n'est pas uniformément réparti, certaines zones sont plus riches que d'autres. Il est donc naturel de vouloir construire un maillage dont la précision (c'est-à-dire la taille des mailles) varie avec le mouvement d'une zone donnée. La figure 1.10 illustre cet argument : l'image représentée figure 1.10.a possède deux objets différents : le cube et le fond.

L'utilisation d'un maillage régulier (figure 1.10.b) impose des calculs supplémentaires inutiles dans la zone du fond. L'utilisation d'un maillage irrégulier (figure 1.10.c) donnera des résultats comparables, pour une gestion plus aisée, car le maillage ainsi construit possède moins de cellules que le maillage régulier.

- Pour une plus grande précision dans la compensation du mouvement et donc dans la segmentation des différents objets présents dans la scène, la géométrie du maillage ne doit plus être fixe i.e. il faut adapter la localisation des noeuds dans l'image pour faire correspondre les arêtes du maillage avec les frontières entre les objets présents dans la séquence, comme le montre la figure 1.12 : l'image au temps  $t$  est la même que celle de la figure 1.11.a, à la différence près que le rectangle grisé a été déplacé d'un petit nombre de pixels. Le maillage à géométrie rigide (figure 1.11.b) n'est ici plus adapté à l'image, car ses arêtes ne coïncident plus avec les arêtes du rectangle. Par contre, le maillage à géométrie adaptative (figure 1.11.c) épouse les contours du rectangle.

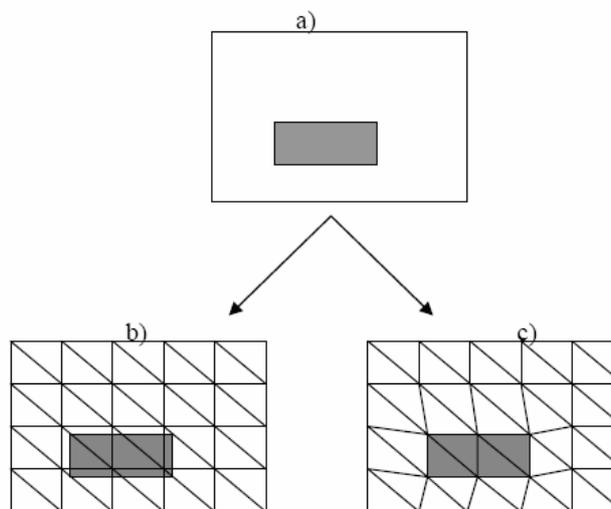


**Figure 1.10 : connectivité régulière/connectivité irrégulière**

**a) image au temps  $t$**

**b) maillage à connectivité régulière**

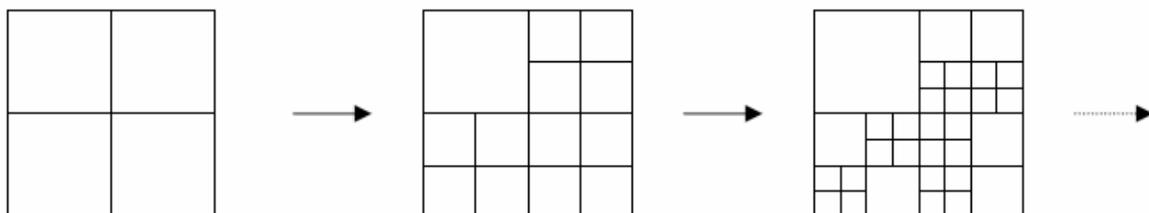
**c) maillage a connectivité irrégulière**



**Figure 1.11 : géométrie rigide/géométrie adaptative**  
 a) image au temps  $t$   
 b) maillage à géométrie rigide  
 c) maillage à géométrie adaptative

Deux approches pour créer un maillage adapté au contenu de l'image ont été développées :

- **approche basée mailles** : Le maillage est formé par découpages (ou subdivisions) successifs des mailles d'un maillage initial régulier et grossier. Un algorithme hiérarchique d'interpolation par grille qui construit une grille par décomposition quadtree a été proposé (figure 1.12). Une méthode similaire est proposée avec un maillage hiérarchique triangulaire (figure 1.13) cette dernière méthode insère une étape d'adaptation du maillage entre les étapes de découpage. Cette adaptation rend le maillage conforme en découpant en 2 ou en 3 les cellules voisines des cellules venant d'être subdivisées. La notion de découpage implique la définition d'un critère de subdivision, qui doit indiquer pour une cellule donnée si son découpage est judicieux ou non.



**Figure 1.12 : approche « quadtree »**

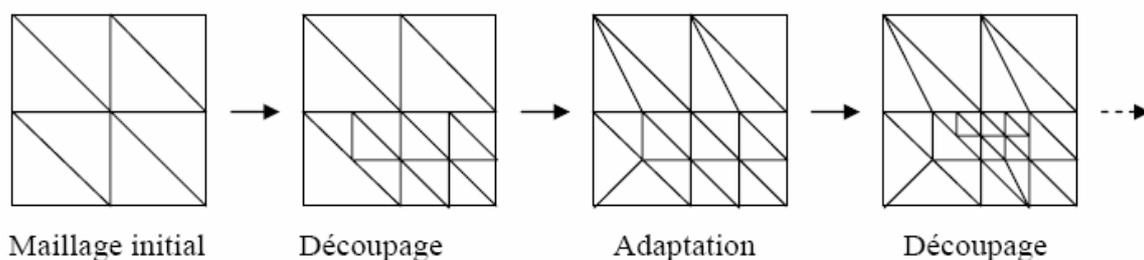


Figure 1.13 : approche « quadtree » par maillage triangulaire

D'autres études utilisent une approche par maillage quadrangle déformable (figure 1.14): entre chaque étape de raffinement, le maillage est déformé pour faire coïncider ses arêtes avec les contours de l'image. La déformation est effectuée par minimisation d'une énergie constitué de trois termes : un terme d'énergie interne (modélisé par un système de masses - ressorts), un terme de mouvement (la DFD) et un terme spatial, tendant à déplacer les arêtes du maillage sur les zones à fort gradient de l'image.

Le critère de découpage proposé est basé sur l'erreur de compensation, qui semble plus approprié que la variance temporelle. Ces méthodes hiérarchiques peuvent engendrer une estimation du mouvement par une approche grossier à fin, offrant une robustesse accrue.

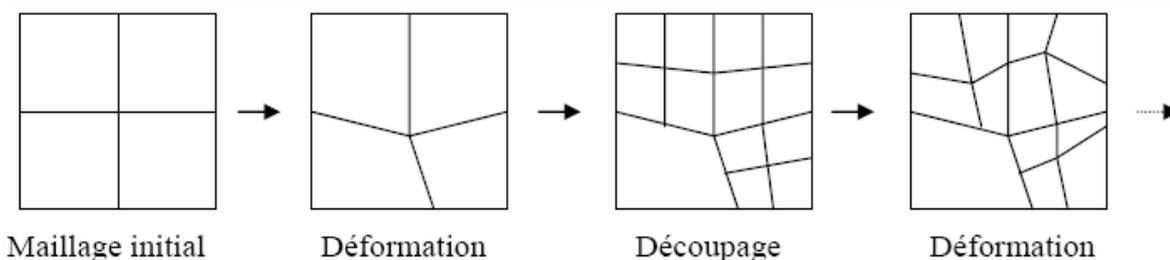


Figure 1.14 : approche « quadtree » + déformation

- **L'approche basée noeuds** : l'algorithme de sélection des points place des noeuds de telle façon que :
  - les contours du maillage soient alignés avec les contours des objets : les noeuds sont placés sur des arêtes c'est-à-dire sur des pixels de gradient spatial élevé.
  - la densité de noeuds soit proportionnelle à l'activité de mouvement locale : les noeuds sont espacés de telle manière que la DFD cumulé ait la même valeur dans chaque disque centré sur ce noeud ; Les pixels à l'intérieur de ce disque ne pourront pas être des noeuds du maillage.

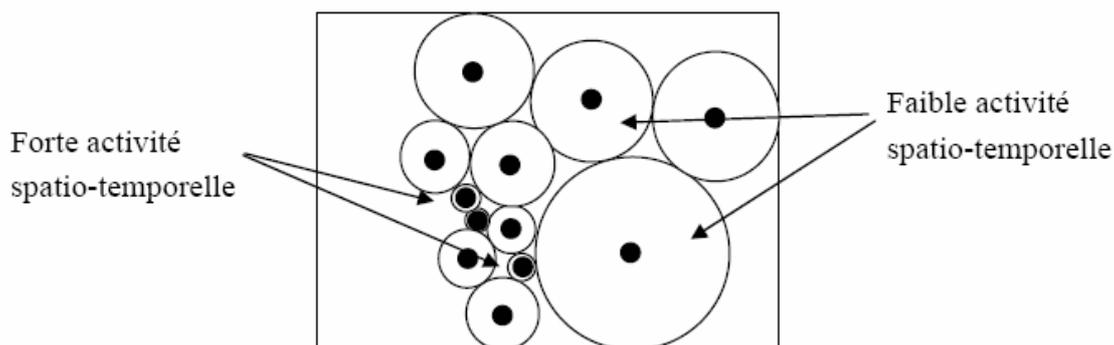


Figure 1.15 : approche par insertion de nœuds.

Un maillage est ensuite construit par triangulation contrainte de Delaunay des noeuds ainsi créés.

#### d) Maillage objet

Plusieurs travaux utilisent la notion de maillage « objet » pour effectuer une poursuite ou un codage des objets vidéo. Un maillage objet se distingue d'un maillage classique par le fait qu'il ne recouvre pas entièrement l'image, mais uniquement l'objet vidéo auquel il est associé (voir figure 1.16).

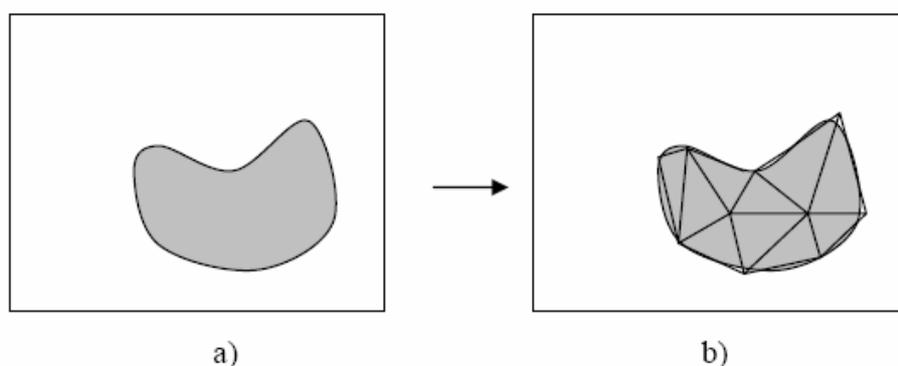


Figure 1.16 : Exemple de maillage objet

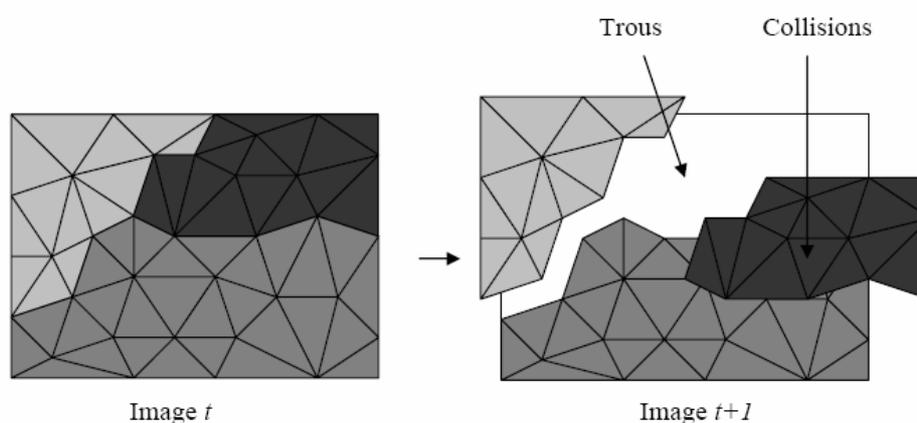
- a) Image contenant deux objets vidéo : le fond et la forme en mouvement
- b) Maillage construit sur l'objet en mouvement

Une scène complète contiendra donc autant de maillages que d'objets vidéo. Une telle représentation permet une approximation des objets vidéo par un contour polygonal dans un but de compression ou pour effectuer le suivi de l'objet vidéo considéré.

Cette approche est apparue car l'estimation de mouvement par maillage n'est réellement efficace que pour les mouvements continus. Or dans une séquence vidéo, des discontinuités apparaissent entre les différents objets en mouvement. L'utilisation de maillages objets permet donc une estimation robuste du mouvement à l'intérieur de chaque objet. Mais une telle approche possède l'inconvénient de ne pas estimer

réellement les discontinuités de mouvement [16], ce qui peut poser des problèmes aux frontières entre les objets (lorsque la forme de l'objet change, par exemple, dans le cas d'auto-occultations).

De plus, du fait des différents déplacements des objets dans une scène, des régions apparaissent au cours du temps (régions découvertes) et des régions disparaissent (régions occultées), comme l'illustre la figure 1.17. Ainsi des trous et des collisions peuvent apparaître entre les différents maillages objets d'une même scène. Des chercheurs proposent de résoudre le problème en détectant à chaque instant les régions découvertes et occultées, et adapte chaque maillage objets en fonction de ces régions.



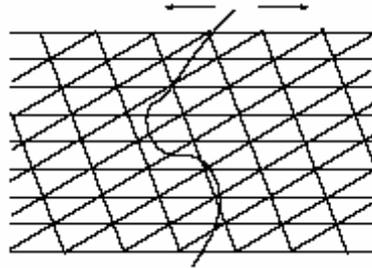
**Figure 1.17 : Trous et collisions engendrés par les mouvements différents des objets (Représentés par des maillages individuels) présents dans la scène**

### e) Approche par ligne de rupture

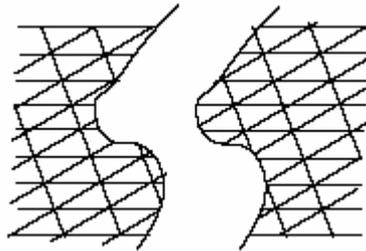
On définit une « ligne de rupture » comme une discontinuité du champ de mouvement, cette notion est illustrée sur la figure 1.18, cette notion a été introduite dans [17]. L'apparition d'une ligne de rupture dans le champ de mouvement est due au mouvement des différents objets. Le maillage est découpé le long de la ligne de rupture et chaque côté du maillage est remaillé indépendamment de l'autre côté, sans les rejoindre. Les deux parties de maillage peuvent alors bouger librement l'une par rapport à l'autre.

Le remaillage est appliqué de chaque côté de la ligne de rupture en créant des recouvrements de mailles qui permettront de prendre en compte de futurs découverts de texture dans cette zone. Cette technique [18] permet d'éviter l'apparition de triangles allongés ou tassés et de mieux estimer le mouvement dans les zones d'occultation.

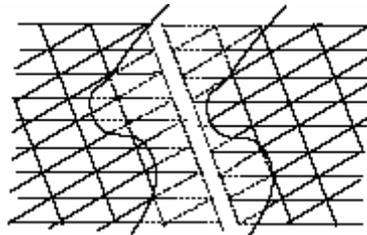
La ligne de rupture est positionnée sur le contour de l'objet créant la discontinuité. Dans le cas d'une ligne de rupture fermée, la découpe du maillage est équivalente à une représentation par objets de la séquence vidéo. Dans le cas d'une ligne de rupture ouverte, les points extrémités de la ligne de rupture nécessitent un traitement particulier.



a) une ligne de rupture apparaît dans le champ de mouvement



b) le maillage est découpé le long de la ligne de rupture



c) chaque côté du maillage est ramillé séparément

**Figure 1.18 : La notion de lignes de rupture sur un maillage**

En effet, la ligne de rupture est détectée et positionnée à l'intérieur d'une zone d'occultation. En dehors de la zone d'occultation, le maillage reste un maillage 2D classique. Autour de la ligne de rupture, la représentation à l'aide des deux portions de maillage est similaire à une représentation objet, cependant les deux portions de maillage appartiennent au même maillage global. Les deux portions de maillage se rejoignent au niveau des points extrémités de la ligne de rupture. A cet endroit, le maillage devient non manifold.

## **1.6 Conclusion**

Ce mémoire aborde deux problèmes parmi les plus importants et les plus complexes dans la vision artificielle, qui sont la segmentation d'images et le suivi d'objets dans les vidéos.

Nous allons consacrer le deuxième chapitre à la description d'un algorithme de segmentation et de suivi d'objet utilisant un maillage actif.

## **Chapitre II**

### ***Implémentation d'un algorithme de segmentation et de suivi d'objet***

## 2.1 Introduction

L'analyse d'images a pour but l'extraction de l'information caractéristique contenue dans une image. Le résultat d'une telle analyse s'appelle très souvent la description structurelle. Celle-ci peut prendre la forme d'une image ou de toute structure de données permettant une description des entités contenues dans l'image. Par opposition avec la phase d'interprétation, l'analyse tente, dans la mesure du possible, de ne pas prendre en compte le contexte (*i.e.* l'application). Essentiellement, l'analyse de l'image fait appel à la segmentation où l'on va tenter d'associer à chaque pixel de l'image un label en s'appuyant sur l'information portée (niveaux de gris ou couleur), sa distribution spatiale sur le support image, des modèles simples (le plus souvent des modèles géométriques).

La segmentation d'images ainsi définie est un domaine vaste où l'on retrouve de très nombreuses approches.

Un autre enjeu majeur est le suivi d'objets dans une séquence vidéo. En l'absence d'une méthode générale, différentes approches ont été proposées, s'appuyant soit sur la notion de région, soit sur celle de contour.

Lorsque les objets à suivre sont de nature non rigide, l'utilisation de modèles déformables est préconisée.

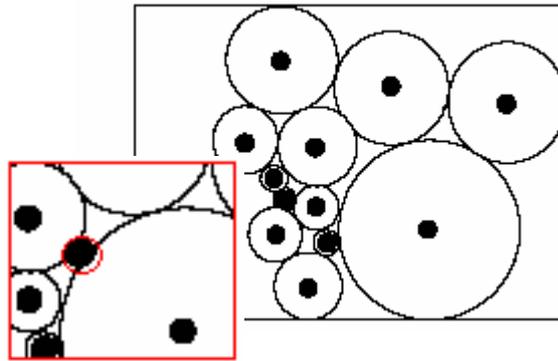
Ce chapitre développera les choix que nous avons effectués pour construire un algorithme de segmentation et suivi d'objet en utilisant un maillage actif.

## 2.2 Stratégie de construction

Pour construire le maillage, il faut d'abord choisir l'approche de construction. Deux choix s'offrent à nous : l'utilisation d'une approche basée mailles ou l'approche basée nœuds.

Dans le cas d'une approche basée nœuds le positionnement des nœuds est effectué sans connaître la topologie du maillage (sa connectivité) et les arêtes ainsi créées ne seraient pas forcément situées sur les frontières des objets. Ainsi, prenons en compte la condition des disques ayant la même DFD, plusieurs lacunes seront formées (figure 2.1), ces dernières peuvent contenir des pixels à fort gradient spatial qui ne seront pas considérés comme des nœuds de maillage puisque la condition de disques à DFD égaux ne sera pas satisfaite.

En plus, dans un souci de qualité de segmentation, l'idée de pouvoir optimiser successivement les nœuds du maillage pour l'adapter au contenu de l'image semble plus intéressante que l'option de placer les nœuds dans les zones à haute activité temporelle et de créer ensuite le maillage par triangulation de Delaunay.



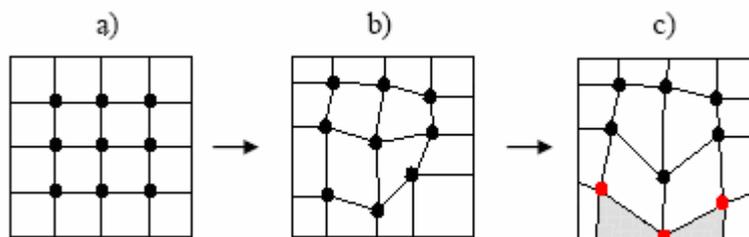
**Figure 2.1 : l'approche par insertion de nœuds :  
un pixel a fort gradient spatial situé dans une lacune**

Ainsi, l'optimisation des positions des nœuds semble être une étape difficilement évitable. De ce fait, la connectivité du maillage sera fixée à l'avance, et où la position des nœuds est optimisée : on opère une déformation intra image.

Il faut par la suite choisir la nature des cellules du maillage : quadrangulaire ou triangulaire.

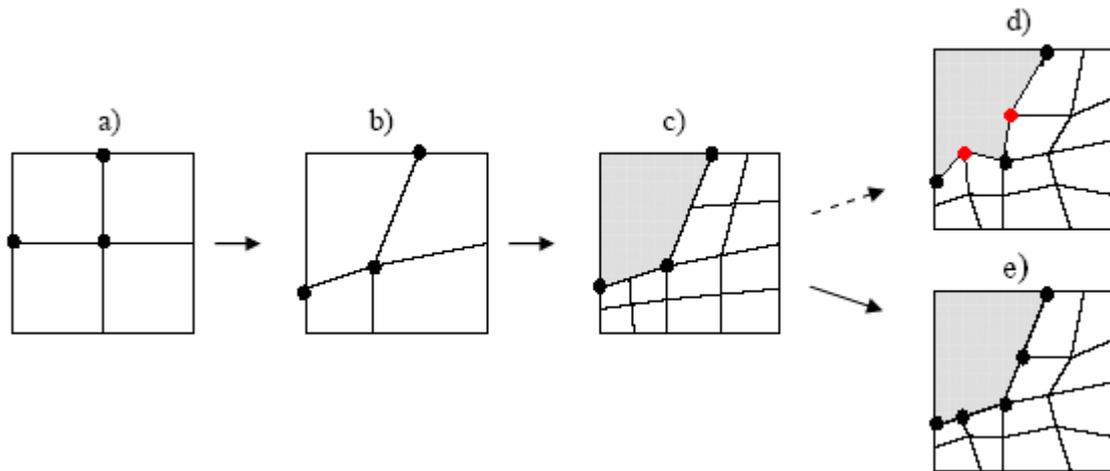
Les maillages à base de quadrangles sont plus difficilement déformables.

Des déplacements des nœuds sont effectués à fur et à mesure, une position optimale d'un pixel ne conduit pas forcément à un maillage quadrangulaire (figure 2.2)



**Figure 2.2 : maillage à base de quadrangles :  
Les nœuds du maillage initial (a) sont déplacés.  
Une fois la déformation effectuée (b) le maillage n'est plus quadrangulaire (c)**

Et si le traitement nécessite une étape d'adaptation, des contraintes de colinéarité sont imposées entre les nœuds créés et les nœuds déjà existant avant la subdivision, afin de garder la forme quadrangulaire des cellules. Ceci a pour conséquence qu'un nœud peut être déplacé au moment de sa création, mais une fois que des nouveaux nœuds sont créés dans un niveau de hiérarchie inférieur, le nœud ne peut plus être déplacé, comme l'illustre la figure 2.3



**Figure 2.3 :** Les nœuds du maillage initial (a) sont déplacés. Une fois la déformation effectuée, des cellules sont subdivisées et des nouveaux nœuds sont créés (c). Les nœuds marqués en noir (e) sont alors figés, car la zone grisée ne serait plus quadrangulaire s'ils sont déplacés (d).

Les maillages triangulaires ne possèdent pas ces inconvénients ; nous avons donc choisi de les utiliser. Le maillage sera donc créé à partir d'un maillage triangulaire régulier, sur lequel on effectue une succession de déformations. Nous opérons dans un schéma d'optimisation donc créer un nouveau maillage sur chaque image est inutile. Un maillage créé à l'image  $t-1$  sera utilisé comme maillage initial sur l'image  $t$ .

### 2.3 Construction de maillage

Nous présentons dans cette partie les notions pratiques nécessaires pour écrire un générateur de maillage (mailleur) bidimensionnel de type Delaunay.

Le maillage de Delaunay s'appuie principalement sur le critère suivant :

*Soit  $T$  un triangle du maillage, alors aucun point du semis (sauf ceux de  $T$ ), n'est contenu dans le cercle circonscrit de  $T$ .*

#### Algorithme de construction du maillage [3]

Pour construire un maillage de Delaunay à partir d'un ensemble donné de points nous allons utiliser un algorithme incrémental tout en suivant les étapes suivantes :

1- prendre pour origine le point de coordonnée minimale  $O = (\min_{i=1, N_p} x^i, \min_{i=1, N_p} y^i)$ , Où les

$(x^i, y^i)$  sont les coordonnées des points  $p^i$ .

2- Trier les  $p^i$  en ordre lexicographique par rapport à la norme  $\|p^i - O\|$  puis par rapport à  $y^i$ .

Cet ordre est tel que le point courant ne soit jamais dans le convexifié des points précédents.

3- Ajouter les points un à un suivant l'ordre prédéfini à l'étape 2. Les points ajoutés sont toujours à l'extérieur du maillage courant. Donc, on peut créer un nouveau maillage en reliant toutes les arêtes frontières du maillage précédent qui voit les points du bon côté.

4. Pour que le maillage soit de Delaunay, il suffit d'appliquer la méthode d'optimisation locale du maillage suivant : autour d'un sommet rendre de Delaunay tous les motifs formés de deux triangles convexes contenant ce sommet, en utilisant le lemme suivant :

Le critère de la boule vide dans un quadrilatère convexe  $s^1, s^a, s^2, s^b$  en  $[s^1, s^2]$  est équivalent à l'inégalité angulaire (propriété des angles inscrits dans un cercle) :

$$\text{angle}(s^1 s^a s^b) < \text{angle}(s^1 s^2 s^b)$$

Nous ferons un échange de diagonale  $[s^a, s^b]$  dans un quadrilatère convexe de coordonnées  $s^1, s^a, s^2, s^b$  (tournant dans le sens trigonométrique) si le critère de la boule vide n'est pas vérifié comme dans la figure 2.4.

Puis, appliquer cette optimisation à chaque nouveau point ajouté pour que le maillage soit toujours de Delaunay.

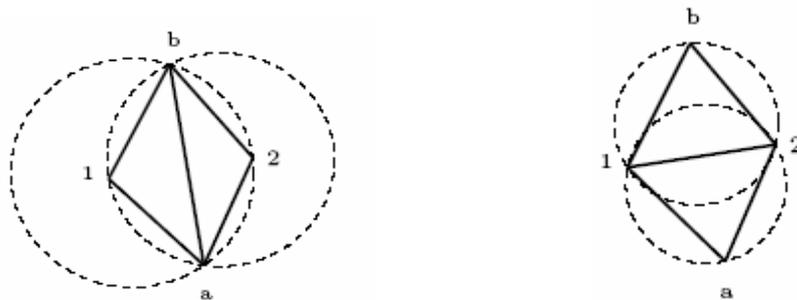


Figure 2.4: Echange de diagonale d'un quadrilatère convexe selon le critère de la boule vide.

5. Il peut arriver que le maillage ne respecte pas la discrétisation de la frontière, comme nous pouvons le remarquer sur la figure (2.5), il faut donc utiliser un algorithme de forçage de frontière.



Figure 2.5 : Exemple de maillage de Delaunay ne respectant pas la frontière.

Nous n'allons pas traiter le cas spécial des objets creux, cependant il est peut probable de fréquenter ce problème.

## 2.4 Déformation du maillage

L'étape de déformation du maillage est l'étape cruciale de notre algorithme ; nous allons décrire dans cette section la méthode employée pour déformer le maillage pour créer la carte de segmentation i.e. déplacer les nœuds du maillage pour faire coïncider ses arêtes avec les frontières entre les objets.

Nous utilisons un critère d'énergie comme critère d'évaluation de cette déformation. Le but recherché étant de trouver la déformation du maillage qui minimise le critère. Ainsi, la définition du critère d'énergie est la tâche fondamentale dans l'algorithme proposé, puisque c'est de celui-ci que va dépendre l'efficacité de l'algorithme.

Nous allons utiliser les deux termes d'énergie suivants :

- Un terme d'erreur de compensation de mouvement.
- Un terme d'énergie visant à faire coïncider les arêtes du maillage avec les forts gradients de l'image.

Donc le maillage sera déformée selon un critère d'énergie  $E(t)$  défini par :

$$E(t) = E_m(t) + \alpha E_s(t) \quad (2.1)$$

Où  $E_m(t)$  est le terme relatif au mouvement (inter images),  $E_s(t)$  le terme spatial (intra image).  $\alpha$  est un paramètre de pondération, servant à équilibrer la contribution entre les deux termes d'énergie, et est fixés par l'utilisateur.

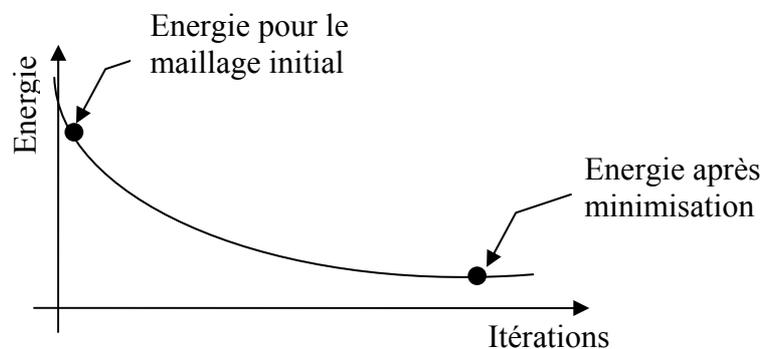


Figure 2.6 : minimisation du critère d'énergie

S.Valette [13] propose un critère décomposable en trois termes, les deux termes cités au dessus plus un terme de régularisation temporelle de la définition des objets.

Notre intention n'est pas d'ajouter une méthode nouvelle à celles qui existent déjà. Nous voulons seulement apporter une solution moins coûteuse en temps de calcul au problème du suivi d'objets.

Nous allons décrire en détail dans les sections suivantes les deux critères d'énergie employés dans notre algorithme : le terme relatif au mouvement et le terme spatial.

### 2.4.1 Critère de mouvement

L'observation d'une séquence vidéo montre que les images successives représentent approximativement les mêmes objets avec quelques différences d'une trame à l'autre. Les différences sont dues au mouvement des objets contenus dans ces images. Un estimateur du mouvement a pour objectif d'attribuer aux composantes d'une image des vecteurs, qui représentent leurs déplacements effectifs entre deux prises de vue consécutives.

La définition d'un critère de mouvement permet de déformer le maillage afin que les cellules soient homogènes au sens du mouvement. Une estimation du mouvement des objets doit donc être effectuée.

L'estimation du mouvement peut être vue comme un problème d'estimation de mouvement direct ou inverse (« avant » ou « arrière »), selon que l'estimation est réalisée entre les instants  $t$  et  $t+1$  ou entre les instants  $t$  et  $t-1$ .

- Dans le cas de l'estimation directe du mouvement, le problème se pose de la manière suivante : si on connaît les échantillons spatio-temporels  $I_t(x,y)$  et  $I_{t+1}(x,y)$ , qui sont liés par la relation (hypothèse de conservation de l'intensité) :

$$I_t(x, y) = I_{t+1}(x + d_x(x, y, t), y + d_y(x, y, t)) \quad (2.2)$$

On doit trouver le vecteur de déplacement

$$d_t(x, y) = (d_{xd}(x, y, t), d_{yd}(x, y, t)) \quad (2.3)$$

Le mouvement des régions découvertes à l'instant  $t$  est correctement estimé, mais le mouvement des zones occultées entre les instant  $t$  et  $t+1$  ne l'est pas.

- Dans le cas de l'estimation inverse du mouvement où les vecteurs de déplacement sont définis entre les instants  $t$  et  $t-1$ , on a la relation

$$I_t(x, y) = I_{t-1}(x - d_x(x, y, t), y - d_y(x, y, t)) \quad (2.4)$$

En arrière : le mouvement des régions occultées entre les instants  $t$  et  $t+1$  est correctement estimé à l'instant  $t$ , mais pour les zones découvertes à l'instant  $t$  (n'existant pas à l'instant  $t-1$ ) le mouvement ne peut pas être estimé (figure 2.7).

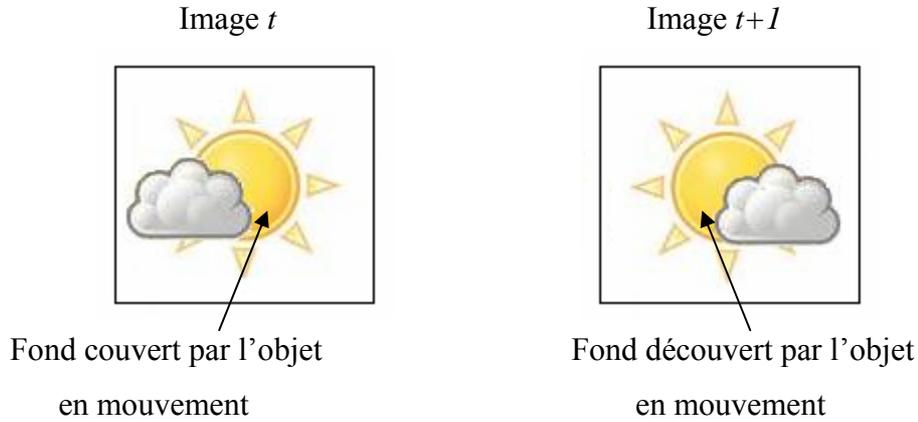


Figure 2.7 : Illustration du concept de fond couvert et découvert

Nous utilisons le terme d'énergie relatif au mouvement  $E_m(t)$  défini par :

$$E_m(t) = \frac{\sum_i \left( \sum_{p \in c_i} [I(\tau_i(p), t + d_i) - I(p, t)]^2 \right)}{\sum_i a_i} \quad (2.5)$$

Où :

- $I(p, t)$  est l'intensité du pixel  $p$  de coordonnées  $(x, y)$  dans l'image au temps  $t$ .
- $a_i$  est l'aire de la cellule triangulaire  $c_i$ .
- $\tau_i$  est une transformation avant (entre les images  $t$  et  $t+1$ ) reflétant le mouvement interne à chaque cellule.
- $d_i$  est la direction (avant ou arrière) choisie pour le mouvement interne de la cellule considérée.

L'estimation avec compensation directe du mouvement est classiquement utilisée dans la compression prédictive de séquences d'images ; en estimation de mouvement on utilise généralement l'estimation inverse.

Cependant ; on peut combiner [13] les deux estimations :

- Considérer uniquement la compensation avant si le mouvement estimé en avant donne une erreur de compensation plus faible qu'en arrière (la cellule considérée contient probablement une zone découverte).

- Si la compensation arrière est meilleure que la compensation avant, la cellule est le lieu d'une occultation à l'image  $t+1$ , choisir donc la méthode arrière.

- Dans le cas où la cellule ne possède aucune région occultée ou découverte, les deux directions d'estimation doivent fournir une erreur de compensation équivalente. Le choix de la direction de l'estimation est donc peu important.

Nous avons considéré, dans notre implémentation, le cas d'une estimation avant, le terme  $d_i$  dans l'équation (2.5) prend donc la valeur  $d_i = 1$ .

Ce terme d'énergie décrit dans cette section est l'erreur quadratique de compensation de mouvement normalisée (la DFD). En effet, l'estimation de mouvement est le bloc fonctionnel le plus coûteux en termes de calculs. Un modèle de mouvement simple rendra l'estimation de mouvement plus rapide, mais fournira une estimation médiocre dans les scènes où le mouvement est complexe. A l'inverse, un modèle de mouvement plus évolué permettra l'analyse de scènes complexes, mais pénalisera l'algorithme au niveau de la rapidité d'exécution.

Le modèle de mouvement le plus utilisé dans les méthodes d'estimation du mouvement est le modèle de mouvement constant ou modèle de translation, qui suppose que tous les pixels du bloc effectuent le même déplacement.

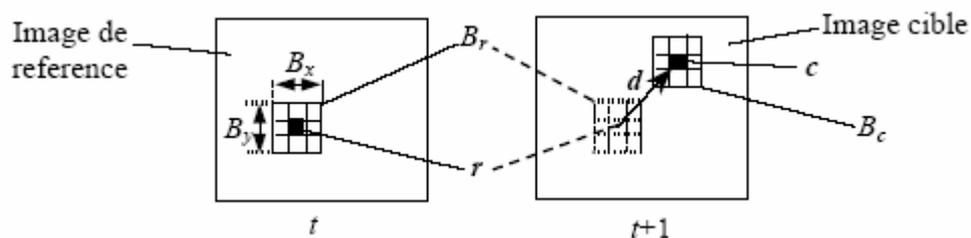


Figure 2.8 : Modèle de translation du mouvement.

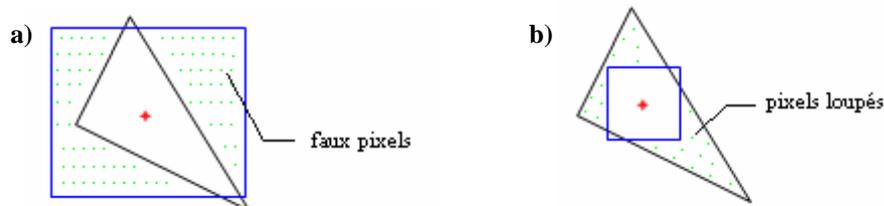
Ainsi un bloc  $B_r$  de pixels de l'image de référence, de dimensions  $B_x \times B_y$ , centré sur le pixel  $(r)$  de coordonnées  $(x,y)$  dans l'image de référence à l'instant  $t$ , sera mis en correspondance avec le bloc  $B_c$  centré en pixel  $c$ , dans l'image cible à l'instant  $t+1$

Nous avons utilisé un modèle de translation (2 paramètres par cellule). Un vecteur déplacement unique est estimé sur chaque cellule, indépendamment d'une cellule à l'autre.

Chaque cellule se verra donc attribuer un seul vecteur déplacement qui est le mouvement de son centre de gravité.

Le problème qui se pose lors de l'estimation du mouvement est que l'algorithme de recherche exhaustive utilise des blocs carrés alors que nous avons à traiter des cellules triangulaires de différentes formes et tailles.

- choisir une grande taille pour les blocs introduit au calcul beaucoup de pixels non appartenant à la cellule considérée (figure 2.9.a) ;
- un bloc de petite taille (figure 2.9.b) ne contiendra que des pixels à l'intérieur de la cellule courante, mais augmente l'erreur de prédiction.
- le maillage subissant de nombreuses déformations durant la segmentation alors l'aspect de ses cellules va changer à fur et à mesure, donc l'utilisation d'un bloc de taille fixe semble non satisfaisant.
- Une idée visant à ne pas fixer la taille du bloc et de l'adapter pour chaque triangle va donc diminuer les erreurs de prédiction mais ne corrige pas le problème.
- Adapter la forme et la taille des blocs corrige le problème théoriquement ; or l'implémentation de cette idée n'est pas possible, calculer la taille et prédire la forme pour chaque cellule du maillage pénalise l'algorithme, puisque cette opération sera répétée plusieurs fois jusqu'à minimiser le critère de mouvement pour chaque image. Cette procédure sera répétée le long de la séquence vidéo ce qui est très coûteux en temps de calcul.



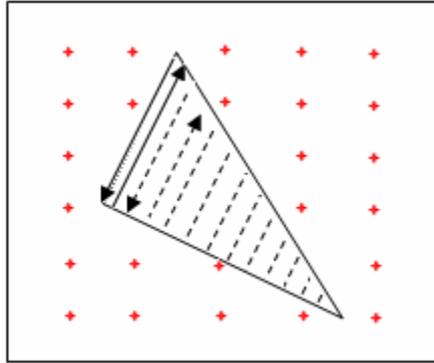
**Figure 2.9 : choix de la taille du bloc : a) bloc de grand taille, b) bloc de petite taille**

- Nous avons décidé de combiner entre l'utilisation des blocs carrés pour l'algorithme de mise en correspondance et l'adaptation de la forme et de la taille des blocs.

Le bloc retourné par l'algorithme sera celui qui minimise le critère de comparaison, habituellement le chargement des blocs se fait ligne par ligne, nous n'allons utiliser cette notion que pour les blocs parcourus dans la fenêtre de recherche dans l'image de référence. Or, pour le bloc courant nous allons procéder comme suit :

Au lieu de charger toutes les intensités des pixels du bloc, et puisque nous disposons à chaque moment de toutes les coordonnées de l'ensemble des nœuds, nous effectuons un balayage à l'intérieur du triangle considéré comme illustré dans la figure 2.10.

La valeur des pixels en commun entre le bloc et l'extérieur de la cellule sera fixé par l'utilisateur, il faut choisir cette dernière de sorte d'être sur qu'aucun pixel de l'image n'a cette valeur, ou bien utilisé tout simplement une variable logique pour distinguer entre les pixels à l'intérieur du triangle et ceux de l'extérieur.



**Figure 2.10 : parcours dans un bloc dans l'image courant**  
**Les points à l'extérieur du triangle n'entrent pas dans les calculs**

La deuxième étape est de chercher le bloc qui correspond le plus à celui de l'image courante :

- Centré sur la position du centre de gravité de la cellule considérée, on charge toutes les valeurs des intensités des pixels du bloc de référence,
- Calculer ensuite l'erreur de prédiction, seuls les pixels contenus dans la cellule rentrent en calcul.
- L'étape suivante consiste à parcourir de manière exhaustive l'ensemble de la fenêtre de recherche toute en passant par les étapes citées si dessus.

Notez pour finir que cette méthode non seulement diminue le temps de calcul mais aussi améliore l'estimation de mouvement en prenant en compte la forme et la taille des cellules, les résultats explorés dans le troisième chapitre le confirme.

Le terme d'énergie de mouvement implique la déformation du maillage pour que les cellules possèdent un mouvement homogène, mais n'est pas affranchi du problème d'ouverture qui rend l'estimation du mouvement très difficile dans les zones homogènes spatialement, comme l'illustre la figure 2.11 : le maillage est construit et déformé sur l'image 100 de la séquence « van ». Une déformation est effectuée par prise en compte unique du critère de mouvement. Dans les régions texturées, les arêtes du maillage sont bien placées sur les discontinuités de mouvement, mais dans la région la plus homogène (la route), l'erreur de compensation est

faible, les arêtes seront mal placées. Le critère spatial décrit dans la section suivante permet une meilleure localisation des contours des objets dans les zones homogènes.



Figure 2.11: maillage déformé uniquement par prise en compte d'un critère de mouvement

#### 2.4.2 Critère spatial

Le critère \_ spatial est introduit dans le terme d'énergie global, pour s'affranchir du problème d'estimation de mouvement dans les zones homogènes au sens de la luminance.

La prise en compte simultanée d'un critère spatial et d'un terme de mouvement doit être effectuée de manière à ce que ces deux critères ne soient pas contradictoires ; puisque si une arête est située dans une région fortement texturée, beaucoup de contours seront détectés et la distance au contour le plus proche sera donc faible quelque soit la position de l'arête. La déformation du maillage se fera donc principalement par la minimisation du critère de mouvement. Inversement, si l'arête est située dans une zone homogène traversée par un contour, le terme d'énergie relatif au mouvement sera peu significatif par rapport au terme spatial, qui permet à l'algorithme de se placer correctement sur le contour spatial de l'image.

Le terme d'énergie spatiale  $E_s(t)$  utilisé est défini par :

$$E_s(t) = \frac{\sum_j \left( r(e_j) \sum_{p \in c_j} dist(p) \right)}{\sum_j l_j} \quad (2.6)$$

Où :

- $e_j$  est une arête du maillage
- $l_j$  est la longueur de l'arête  $j$

- $dist(p)$  est la distance du pixel  $p(x,y)$  au contour détecté le plus proche
- $r(e_j) = 1$  si l'arête  $e_j$  est sur une frontière entre deux objets vidéo, Sinon  $r(e_j) = 0$ .

Cette énergie est similaire à une énergie de contour actif : le modèle de contour actif ou *snake* introduit par Kass et al [19] utilise comme idée le déplacement des points pour les rapprocher des zones de fort gradient tout en conservant des caractéristiques comme la courbure du contour ou la répartition des points sur le contour ou d'autres contraintes liées à la disposition des points.

La différence entre ces deux énergies est l'insertion du terme  $r(e_j)$  dans l'énergie spatiale (figure 2.12). En effet, la prise en compte des arêtes à l'intérieur des objets est inutile, car seule la frontière entre les objets est importante.

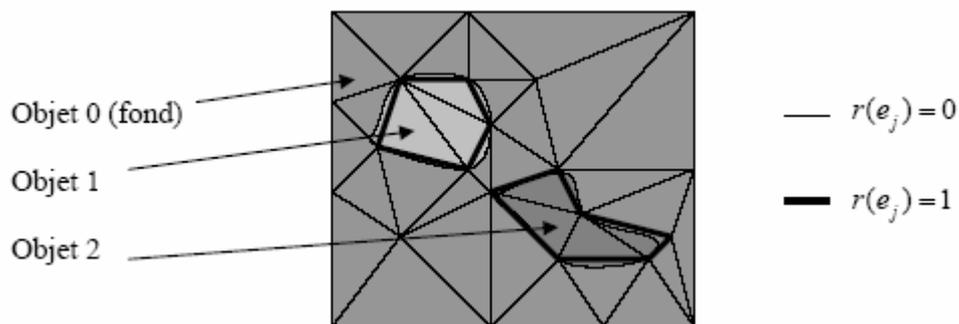
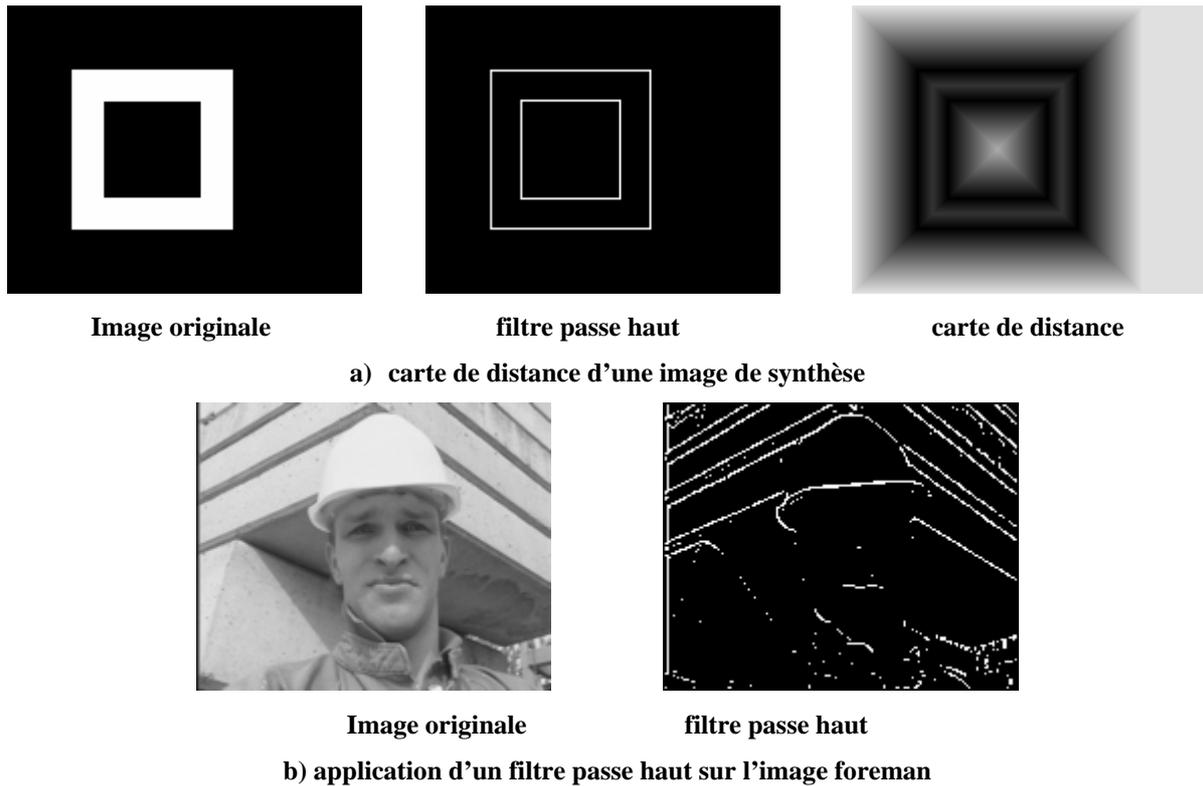


Figure 2.12 : détermination de  $r(e_j)$  pour chaque arête  $e_j$

La carte de distances, aussi appelée transformée de distances qui associe à chaque pixel de l'image la distance au *point obstacle* le plus proche est construite en passant par une étape de détection de contour.

- On applique un filtre passe haut ( $\{[-1,-1,-1], [-1,9,-1], [-1,-1,-1]\}$ ) ;

Contrairement au cas des séquences synthétiques (figure 2.13.a), la détection de contour pour des images réels utilisant un filtre passe haut (figure 2.13.b) nécessite une phase de fermeture des contours et suppression de toutes les crêtes pour obtenir une carte de distance significative.



**Figure 2.13 : filtrage et carte de distance**

Plusieurs autres filtres existent qui ont pour but d'augmenter le contraste et de mettre en évidence les contours (figure 2.15).

- L'utilisation du filtre Prewitt (figure 2.14.b) est basée sur l'application de deux filtres horizontal et vertical, qui seront additionnés pour former le résultat final.

- L'algorithme de Sobel est l'un des opérateurs les plus simples utilisés en traitement d'image pour la détection de contours, et qui donne toutefois des résultats corrects (figure 2.14.c).

Le filtre de Sobel est apprécié pour sa simplicité et sa rapidité d'exécution. Ces qualités posent des problèmes lorsqu'il s'agit de traiter une image complexe. Le filtre de Canny (figure 2.14.d) a été bâti autour de l'algorithme de Sobel pour améliorer ses résultats.

L'auteur l'a conçu pour être optimal suivant trois critères clairement explicités :

- 1- *bonne détection* : faible taux d'erreur dans la signalisation des contours,
- 2- *bonne localisation* : minimisation des distances entre les contours détectés et les contours réels,
- 3- *clarté de la réponse* : une seule réponse par contour et pas de faux positifs

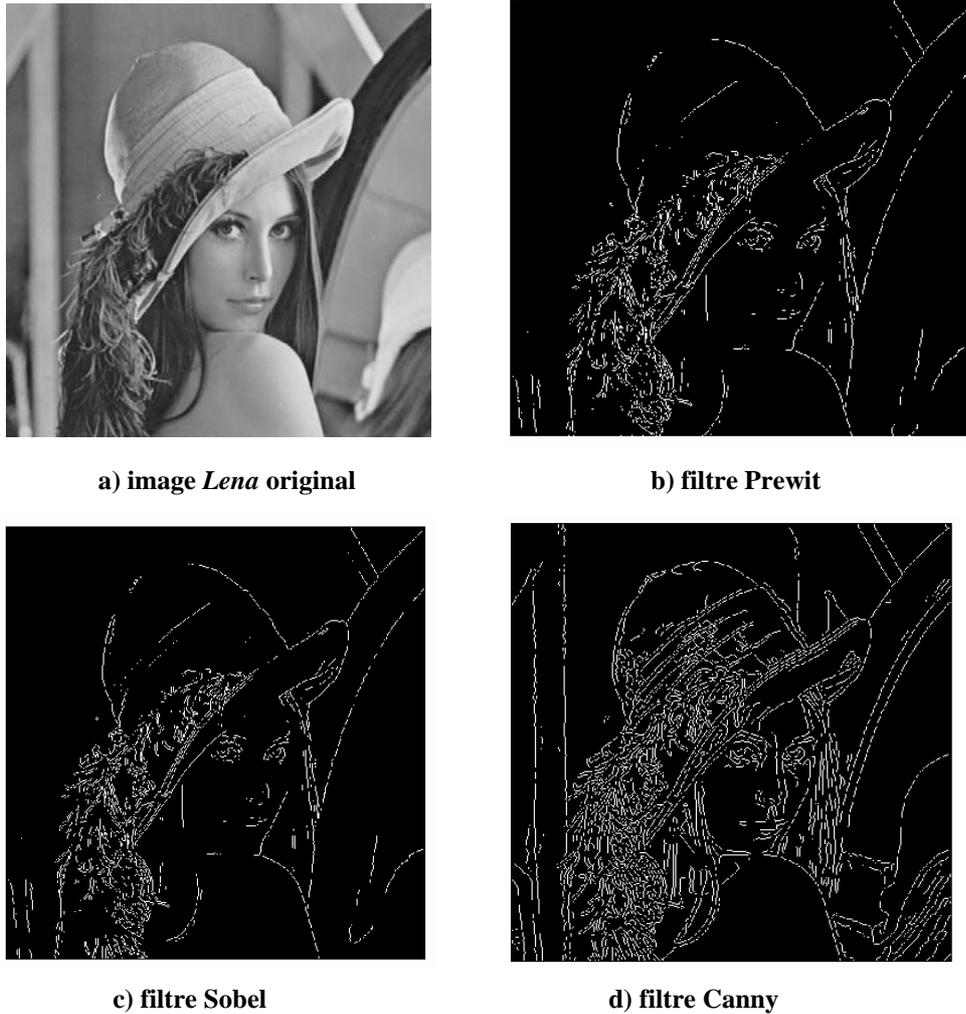


Figure 2.14 : filtrage de l'image « Lena »

Après détection de contour, nous utilisons une recherche exhaustive au tour de chaque pixel, le but est de trouver le contour le plus proche, l'image filtrer est une image binaire, les contours sont représentés par des pixels prenant la valeur 255 ; tandis que le reste des pixels prend une valeur nulle (le noir). On utilisant un pas de recherche égal a un, lorsque on coïncide un pixel appartenant à un contour, on affecte le déplacement effectué au pixel en cour de traitement, a la fin de cette opération l'intensité de chaque pixel sera sa distance au contour le plus proche.

## 2.5 Minimisation du critère d'énergie

Pour minimiser l'énergie globale un algorithme de relaxation déterministe est utilisé.

Nous utilisons l'algorithme avec 8 itérations pour chaque nœud, voire une itération pour le temps réel (i.e. Une fenêtre de dimensions 3 centrée sur le nœud considéré)

L'algorithme va déplacer itérativement chaque nœud du jusqu'à ce que le critère d'énergie ne puisse plus être diminué.

$$E(t) = \frac{\sum_i \left( \sum_{p \in c_i} [I(\tau_i(p), t + d_i) - I(p, t)]^2 \right)}{\sum_i a_i} + \alpha \frac{\sum_j \left( r(e_j) \sum_{p \in c_j} dist(p) \right)}{\sum_j l_j} \quad (2.7)$$

Nous voulons apporter une solution moins coûteuse en temps de calcul, la déformation du maillage n'intervient donc que sur les nœuds placés sur les frontières entre les objets.

En effet, le déplacement des nœuds intérieurs aux objets vidéo peut faire diminuer le critère d'énergie mais ne changera en rien le résultat de la segmentation.

Nous avons limité le nombre d'itérations de déformation du maillage toujours dans un souci de réduction du temps de calcul et afin d'assurer la fin des itérations lorsque l'énergie reste constante.

Le critère d'énergie contenant un terme relatif au mouvement, un algorithme d'estimation de mouvement a dû être implémenté. Après chaque déformation, l'estimation du mouvement des cellules ayant été déformées doit être recommencée, car le support d'estimation a lui aussi changé, même chose pour le calcul de l'énergie spatiale puisque chaque fois qu'une déformation de maillage existe on applique un maillage de Delaunay ; ce qui implique le changement de l'aspect des cellules, donc l'énergie spatiale doit être recalculer après chaque déformation.

L'algorithme de déformation du maillage se déroule ainsi :

- Marquer tous les nœuds  $n_j(x,y)$  placés sur les frontières entre les objets ;
- Estimer le mouvement sur toutes les cellules  $c_i$  du maillage ;
- Calculer l'énergie spatiale ;
- *Compteur*=1
- *maillage\_déformé* = 1 ;
- Tant que (*maillage\_déformé* = 1 et *Compteur* < seuil) faire
  - {
  - *maillage\_déformé* = 0 ;
  - Pour chaque nœud  $n_j(x,y)$  marqué faire :
    - {
    - Chercher la nouvelle position  $(x', y')$  du nœud minimisant  $E(t)$ .

```

- Si ( $x = x'$  et  $y = y'$ )
{
  - la position  $(x, y)$  était optimale, le nœud  $n_j(x, y)$  n'est plus marqué .
}
- Sinon :
{
  - Placer le nœud  $n_j(x, y)$  sur ses nouvelles coordonnées  $(x', y')$ .
  - Marquer tous les nœuds  $n_j(x, y)$  placés sur les frontières entre les objets .
  - Maillage de Delaunay.
  - Maillage_ déformé = 1
}
}

```

Compteur = Compteur+1.

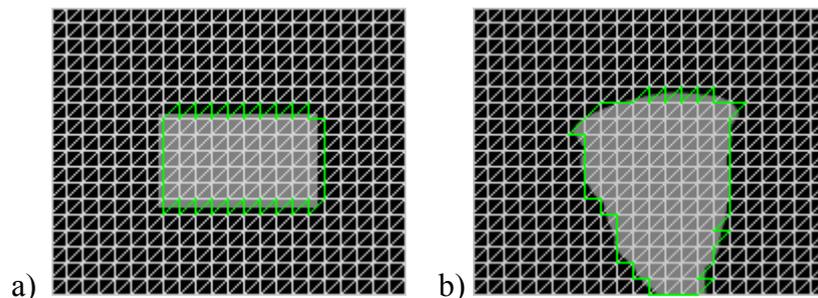
}

Comme présenté dans l'algorithme de déformation, une étape de marquage est indispensable après chaque changement de la forme des cellules.

Nous avons élaboré un algorithme de marquage automatique passant par les étapes suivantes :

- calculer les distances de toutes les arêtes par rapport au contour de l'objet (en utilisant un plan alpha).
- Prendre comme origine les coordonnées de l'arête ayant la valeur minimum de distance.
- Comparer les distances des arêtes qui ont un nœud en commun avec l'arête d'origine et choisir ensuite l'arête présentant la plus petite valeur de distance.
- Marquer les arêtes une par une jusqu'à retrouver le point de départ.

A la fin de cette procédure on aura marqué les nœuds et les arêtes contour, comme illustré dans l'image ci dessous :



**Figure 2.15 : Operation de marquage sur un plan alpha**  
a) objet rectangulaire, b) visage du foreman

Les arêtes marquées apparaissent dans la figure avec une couleur verte. Seules ces arêtes entreront dans le calcul du terme d'énergie.

- Le maillage régulier étant présent sur la première image, on peut alors effectuer un test pour avoir les points communs entre les nœuds du maillage et les points des contours pour avoir comme résultat les points à marquer.

Le critère global d'énergie à minimiser est une somme de deux critères pondérés par le coefficient  $\alpha$ . Les deux critères d'énergie  $E_m(t)$ ,  $E_s(t)$  sont normalisés par la surface des cellules pour  $E_m(t)$ , et par la longueur des arêtes pour  $E_s(t)$ , pour garantir leur invariance avec l'échelle. Ceci nécessiterait l'ajustement du paramètre  $\alpha$  pour que la contribution respective des deux énergies reste constante vis-à-vis de la taille des objets.

## 2.6 Prédiction du maillage d'une image à l'autre

Nous commençons par la création d'un maillage régulier pour la première image, qui sera déformé au cours de la séquence. Lorsque la construction du maillage est terminée sur l'image  $t$ , ce maillage sera considéré comme maillage initial pour l'image  $t+1$ , après une étape de prédiction.

Dans cette étape, chaque nœud du maillage est déplacé. L'introduction d'un terme d'énergie de mouvement et une énergie spatiale dans l'énergie total à minimiser permet d'une part d'effectuer une prédiction du maillage au temps  $t+1$  par le maillage au temps  $t$  et permet aussi de déplacer les nœuds intérieurs aux objets vidéo, ce qui améliore visuellement l'évolution du maillage au cours du temps. En effet, pendant l'étape de déformation du maillage, seuls les nœuds situés aux frontières entre les objets sont déplacés. Si l'on n'effectuait pas l'étape de prédiction, les nœuds intérieurs aux objets vidéo seraient immobiles, même si l'objet vidéo dont ils font partie est en mouvement.

## 2.7 Conclusion

Nous avons décrit dans ce chapitre les différentes étapes de l'algorithme de segmentation et de suivi d'objet utilisant un maillage actif : commençant avec un maillage régulier suivi d'une étape de déformation pour créer une carte de segmentation en utilisant un critère de mouvement et un critère spatial.

Il nous reste à faire des que de tester de la fiabilité de l'algorithme, c'est l'essentiel du troisième chapitre.

**Chapitre III**  
***Applications***

### 3.1 Introduction

Nous allons présenter dans cette section les résultats obtenus avec notre algorithme. L'application se fait en deux étapes, la première sur des images synthétique que ce soit dans le cas de segmentation ou de suivi d'objet, la seconde sur des séquences réelles tout en étudiant la minimisation du critère d'énergie,

Avant de passer à la segmentation et au suivi d'objet ; nous allons commencer par la présentation des résultats obtenus pour l'estimation du mouvement, tout en exposant une comparaison entre le résultat d'estimation du mouvement avec un maillage régulier et maillage non régulier

### 3.2 Résultats :

#### 3.2.1 Estimation du mouvement

Nous avons testé l'algorithme d'estimation de mouvement pour trois cas, selon le choix du bloc courant:

**Premier cas :** bloc carré plus grand que les cellules.

**Deuxième cas :** bloc carré de taille inférieure à la taille des cellules.

**Troisième cas :** notre proposition : un bloc triangulaire ajusté à la taille et la forme de chaque cellule du maillage.

Les résultats obtenus sont illustré dans la figure 3.1:

- Les images (c) et (d) montre l'erreur d'estimation effectué sur les images 105 de la séquence Foreman et l'image 176 de la séquence Coastguard (a et b) avec l'utilisation d'un bloc de taille inférieure à celle des cellules du maillage.
- Les images (e) et (f) présentent l'erreur obtenus pour le deuxième cas.
- L'erreur de compensation de troisième cas est exposée sur les images (g) et (h).

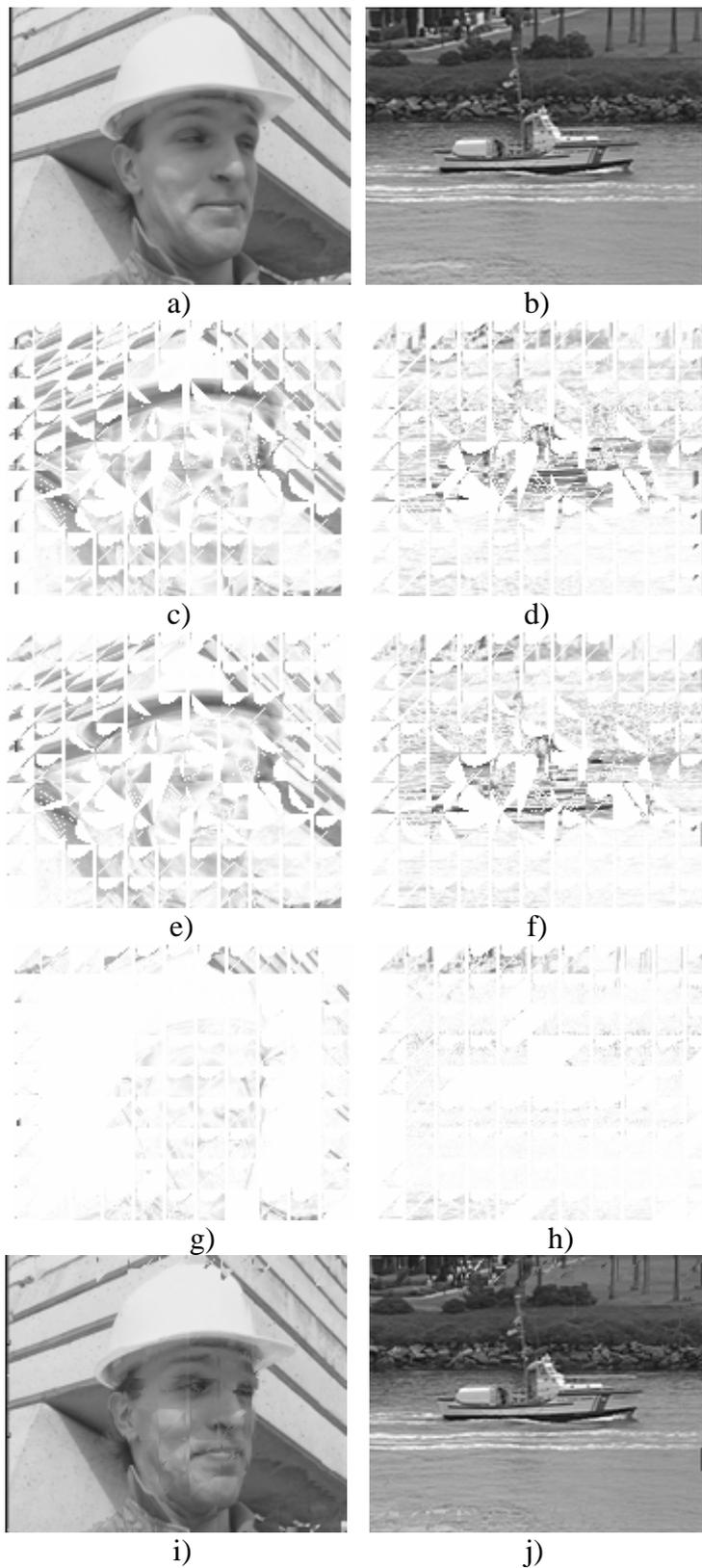


Figure 3.1 :(a) et (b) l'image 105 et 176 des séquences Foreman et Coastguard respectivement.  
 - les images erreur (c-h), ces images représentent la valeur absolue de l'erreur inversée.  
 - (i) et (j) les images après compensation du mouvement

Rappelons que les cellules de notre maillage ont une forme triangulaire; et puisque on utilise dans les deux premiers des blocs carrés, donc :

- Beaucoup de pixel non appartenant à la cellule considérée entreront dans le calcul de son mouvement dans le premier cas.
- Contrairement au premier cas, des pixels seront manqué pour le deuxième cas.

Pour les deux premiers cas, l'erreur de compensation du mouvement est significative, ceci est non seulement du à la forme rigide du maillage régulier utilisé, mais aussi à l'inhomogénéité existante entre la forme des cellules du maillage et le bloc de recherche.

Cependant, on constate que les résultats pour le dernier cas sont comparable. Une réduction remarquable de l'erreur figure dans les images (g) et (h).

On peut remarquer par exemple que le mur dans l'image de foreman ne présente presque pas d'erreur, le mouvement donc est correctement estimé.

Une très grande amélioration est aussi apportée au visage de foreman, ce qui implique que le mouvement est plus précis.

L'application sur la séquence de Coastguard confirme l'amélioration apportée suite à l'implémentation de notre proposition.

On peut conclure que l'algorithme est plus efficace et l'estimation du mouvement est meilleure.

- Les images (i) et (j) sont le résultat après compensation de mouvement, les erreurs les plus élevées sont ceux des bords de l'image puisque ces derniers nécessitent un traitement particulier.

Pour avoir idée plus claire sur la performance de l'algorithme d'estimation du mouvement, nous utilisons la mesure du PSNR (Peak Signal to Noise Ratio) entre l'image originale et celle reconstruite.

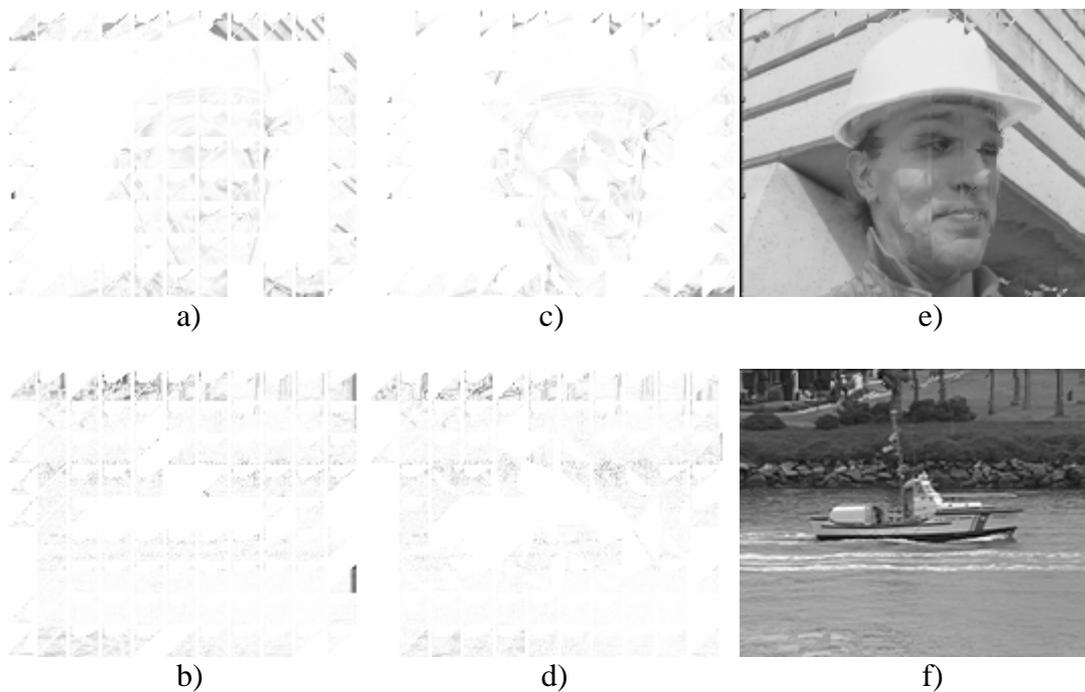
Séquence \ Cas	1	2	3
Foreman	16.360	15.861	<b><u>28.591</u></b>
Coastgard	18.033	17.742	<b><u>28.243</u></b>

**Tab 3.1 :** PSNR de la séquence « foreman » et « Costguard » pour les trois cas.

L'augmentation de la valeur PSNR dans le dernier cas, en comparaison avec les deux premiers, prouve que l'estimation de mouvement a été grandement améliorée par notre méthode.

La figure 3.2 est une comparaison entre l'estimation de mouvement avec maillage régulier (image a et b) et maillage déformé au sens du mouvement (image c et d).

Le résultat est encore mieux, on remarque que plusieurs parties du visage de Foreman (les yeux par exemple) ne présentent plus d'erreur avec un maillage déformé ; ce maillage, le tût de PSNR est comme suit : 28.595 pour la séquence Foreman et 28.372 pour Coastguard. On constate qu'une amélioration est encore apportée aux résultats, puisque plusieurs cellules sont homogènes au sens du mouvement.



**Figure 3.2:** (a-d) les images erreur  
(e) et (f) sont les images après compensation du mouvement

## 3.2.2 Segmentation

### 3.2.2.1 Images synthétiques

Nous allons commencer l'implémentation du programme avec des images contenant des formes de bases.

- figure 3.3.a : un maillage triangulaire initial régulier est appliqué sur cette image, les arêtes en vert sont le résultat d'une opération de marquage semi-automatique. Ces

dernier sont choisis de telle façon à avoir les arêtes qui se rapprochent le plus des frontières de l'objet.

- figure 3.3.b : cette figure représente le résultat obtenu après trois itérations. L'objet est bien segmenté.

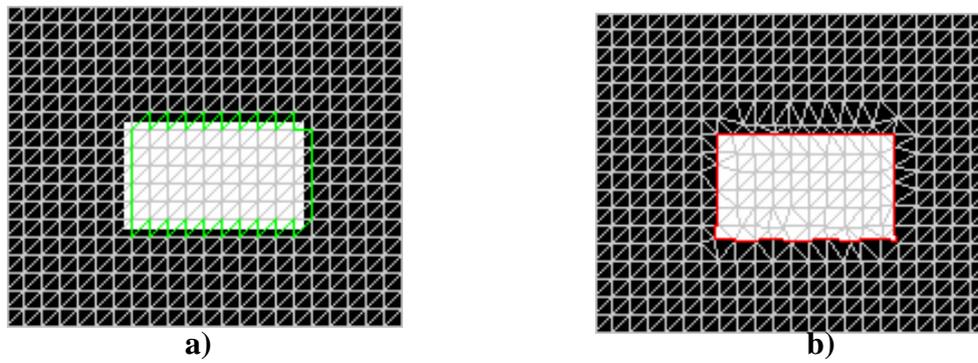
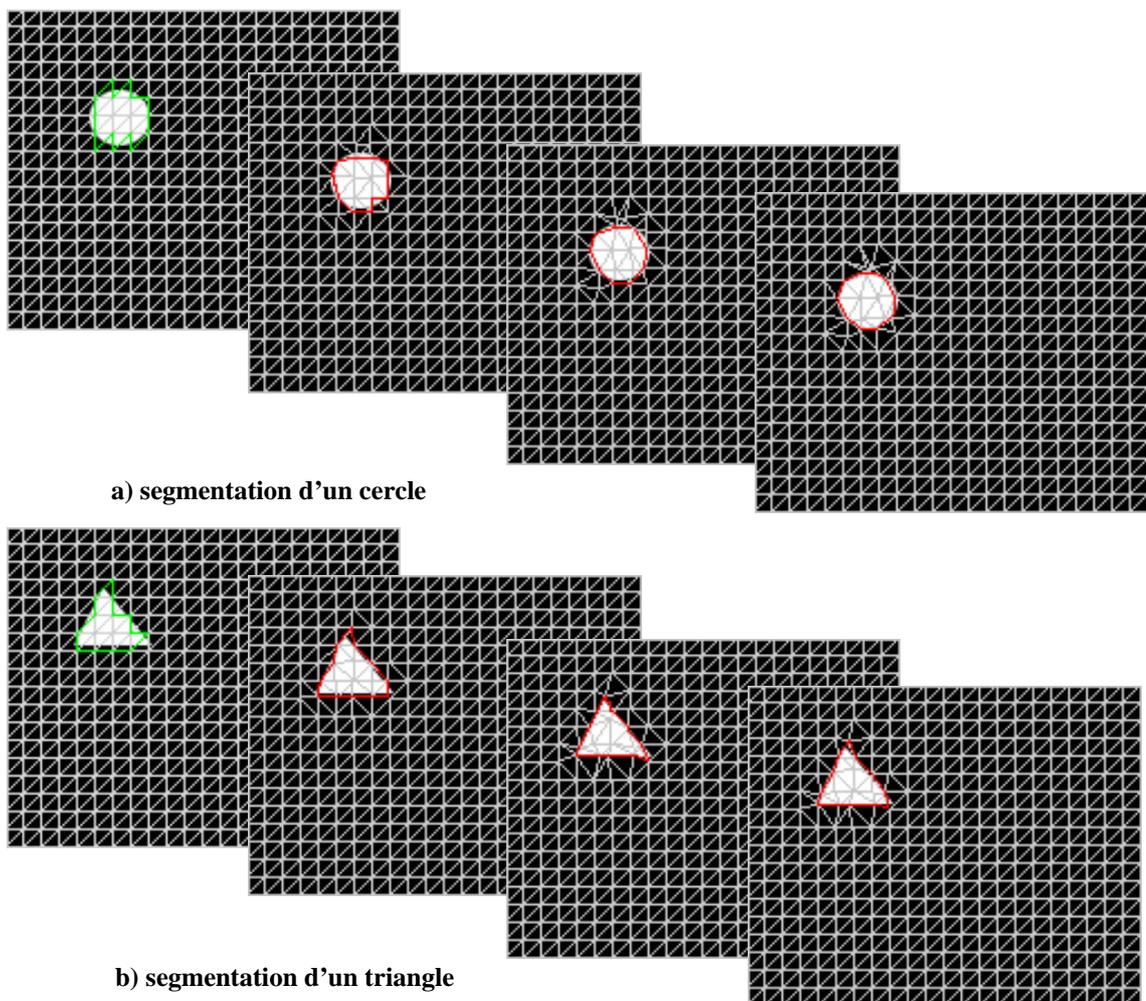


Figure 3.3 : segmentation d'une image de test (rectangle blanc sur fond noir)



a) segmentation d'un cercle

b) segmentation d'un triangle

Figure 3.4 : trois itérations de segmentation d'un cercle (a) et un triangle (b)

On constate pour les trois images de test que les trois objets élémentaires : rectangle, cercle et triangle (figure 3.3, 3.4.a et 3.4.b) sont correctement segmentés, à la condition que les mailles soient suffisamment petites notamment dans le cas des objets courbés (le cas du cercle). Cependant, on observe des déformations importantes dans certaines mailles (les plus proches des contours) après plusieurs itérations ; là où les cellules deviennent dégénérées et ne peuvent plus être déformées.

### 3.2.2.2 Images réelles

Nous avons testé notre algorithme sur deux séquences réelles :

- La figure 3.5 représente la segmentation du visage sur l'image 104 de la séquence « foreman ».

Un masque de segmentation est fourni à l'algorithme (figure 3.5.a) ; le contour en vert sur l'image (b) de la figure 3.5 représente les arêtes marquées du maillage initial.

Après trois itérations, le visage du « foreman » est segmenté, en effet il y'a une rigidité dans le maillage dans la partie inférieure de l'image puisque l'algorithme ne traite pas les bords de l'image.

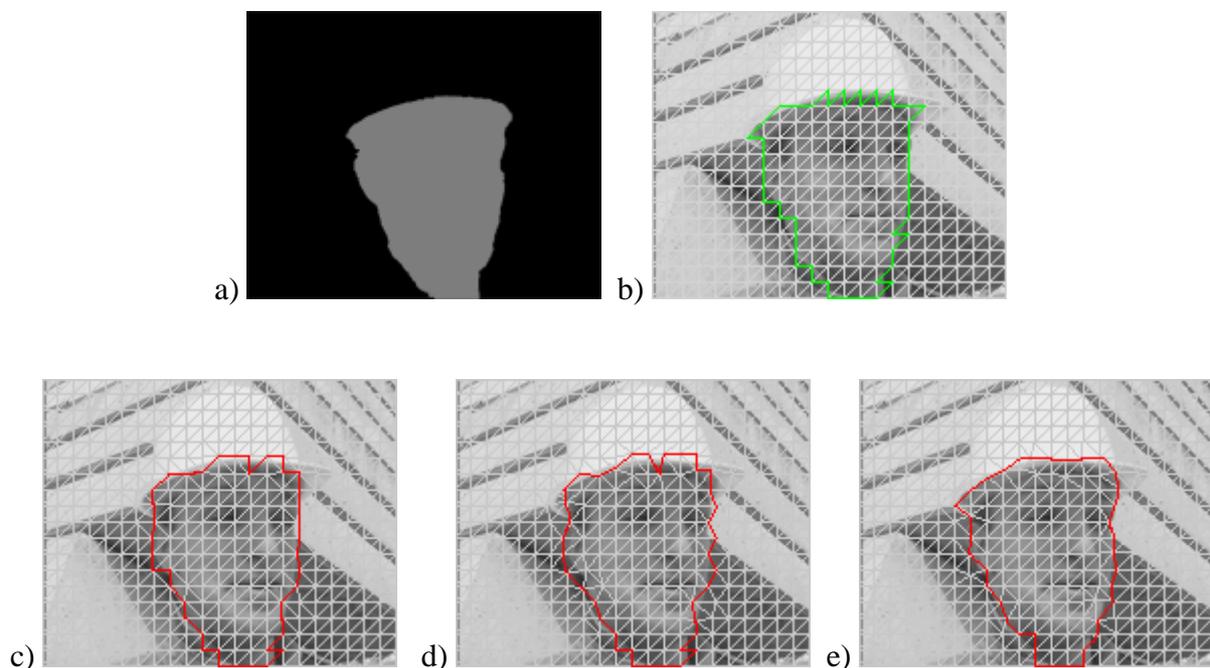


Figure 3.5: Résultat de segmentation sur « foreman »

La figure 3.6 illustre la minimisation itérative du terme d'énergie effectuée lors de la construction du maillage sur l'image 176 de la séquence « Coastguard ». Le masque de segmentation initial et la détection de contour sur l'image 175 sont présentés dans les figures 3.6.b, 3.6.c consécutivement.

Les images de la figure 3.6.e à 3.6.h montrent l'évolution du maillage pour les quatre premières itérations. Le paramètre de pondération utilisé est égal à 200.

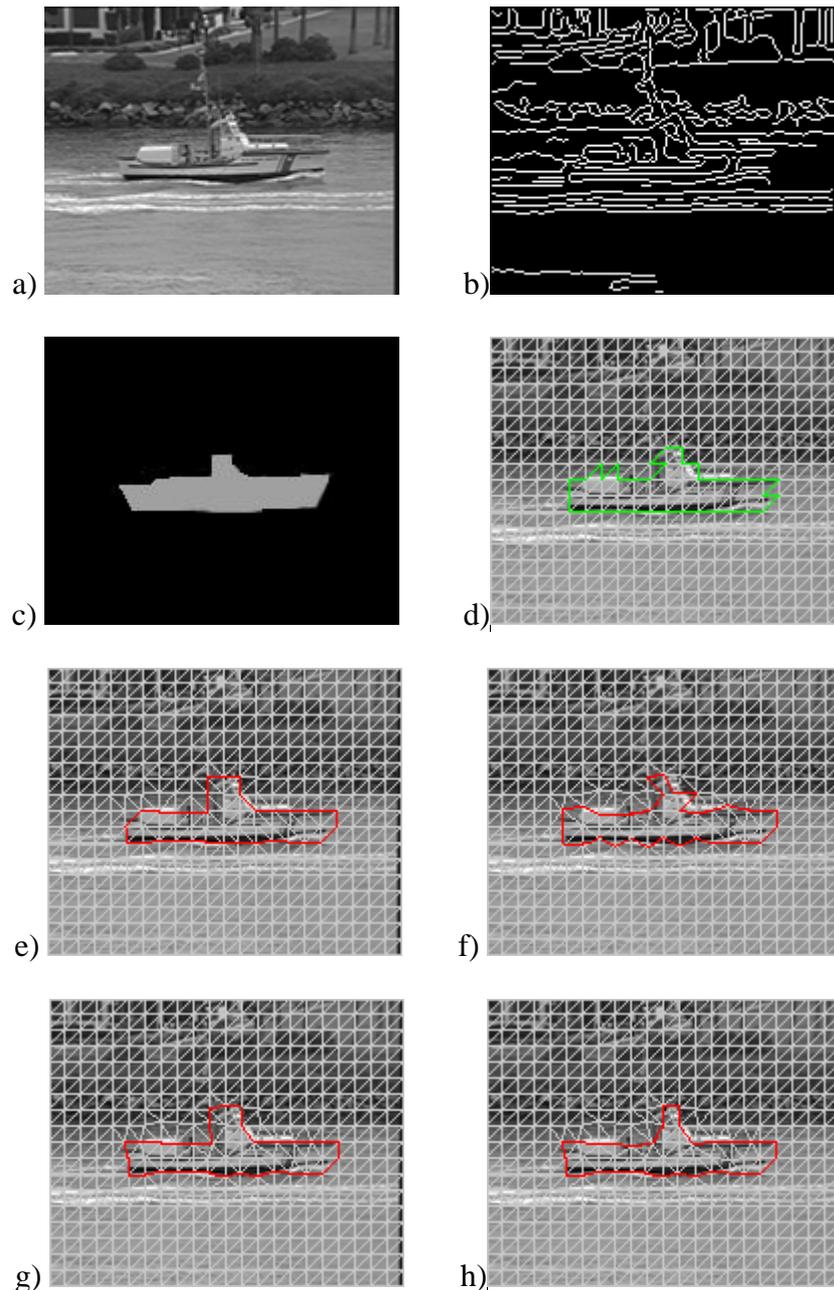
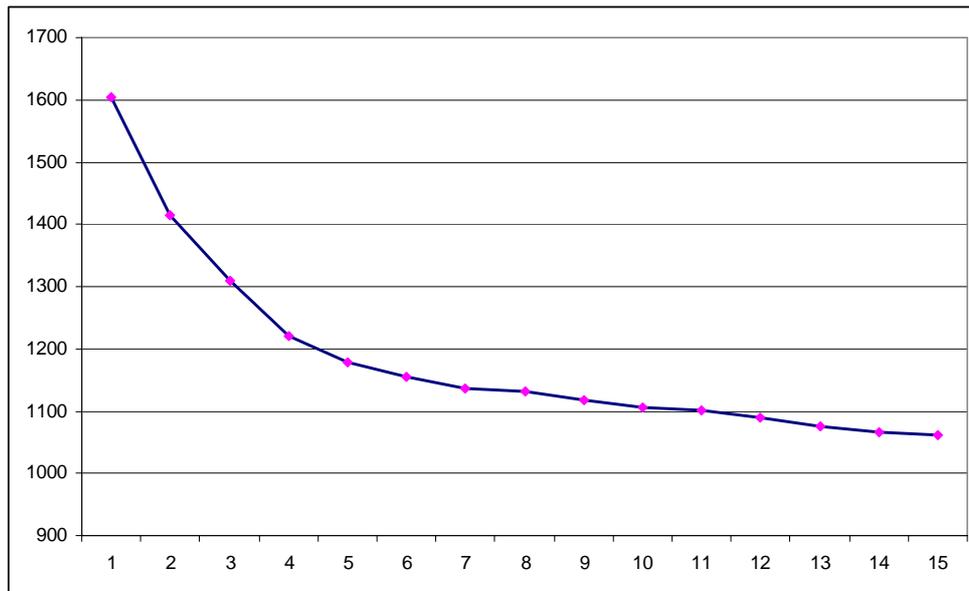


Figure 3.6: déformation du maillage lors de la minimisation du critère d'énergie: masque de segmentation sur l'image 175(c), maillage initial sur 104 (d), forme du maillage sur l'image après quatre itérations (e-h)

### 3.2.3 Suivi d'objet

Le suivi d'objet doit être précédé par une étape de segmentation; la courbe suivante illustre la minimisation itérative du terme d'énergie effectuée lors de la construction du maillage sur l'image 105 de la séquence « foreman » pour des blocs de taille 16. Pour les premières itérations l'énergie se diminue rapidement et l'algorithme s'arrête au bout de 15 itérations puisque l'énergie ne peut plus être diminué, le maillage est donc optimal.



**Figure 3.7: Évolution du critère d'énergie en fonction des itérations pour l'image 105 de la séquence « Foreman »**

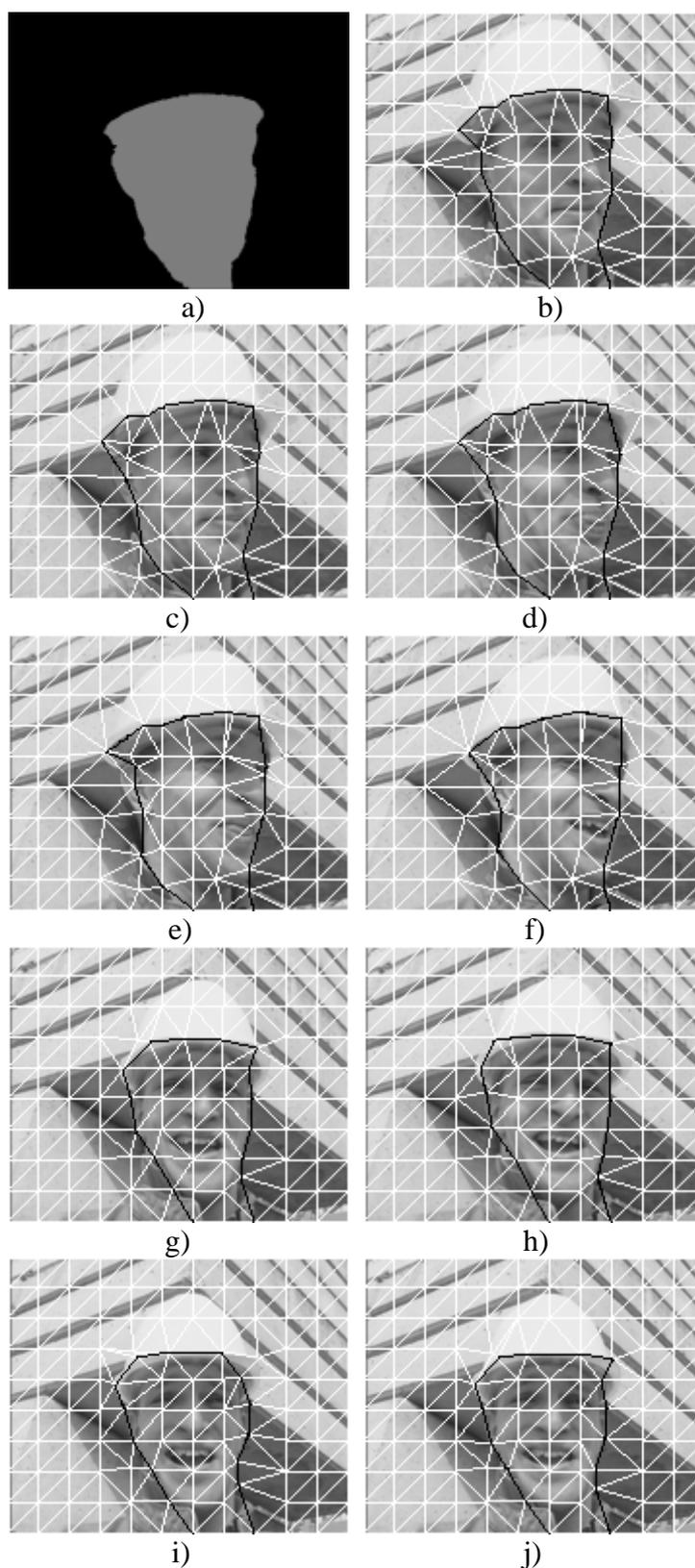
- La figure 3.8 présente le résultat du suivi d'objet de la séquence « foreman » en utilisant un maillage actif ; un masque de segmentation (image a) est fourni à l'algorithme ; le visage est correctement suivi sur les images 106, 107, 109, 111, 117, 119, 121 et 123 (b-j).

Les blocs utilisés sont de taille 16.

Quoique le filtrage utilisé et fourni à l'algorithme n'est pas vraiment exacte, les résultats peuvent être jugé satisfaisants.

- Le suivi par maillage 2-D est un outil intéressant pour la segmentation d'objet [20], le résultat précédent peut être représenté d'une autre façon, illustré par la figure 3.9.

Les images (a-e) sont le résultat du suivi d'objet, chaque image entre eux peut être considéré comme une segmentation initiale pour le suivi du reste de la séquence ; les images (f-j) représentent la segmentation obtenue du visage de foreman sous forme binaire, le visage est bien segmenté et correctement suivi.



**Figure 3.8:** suivi d'objets sur la séquence « foreman » :masque de segmentation initiale sur l'image 105 (a) et évolution du maillage sur les images 105, 106, 107, 109, 111, 117, 119, 121 et 123 (b-j)

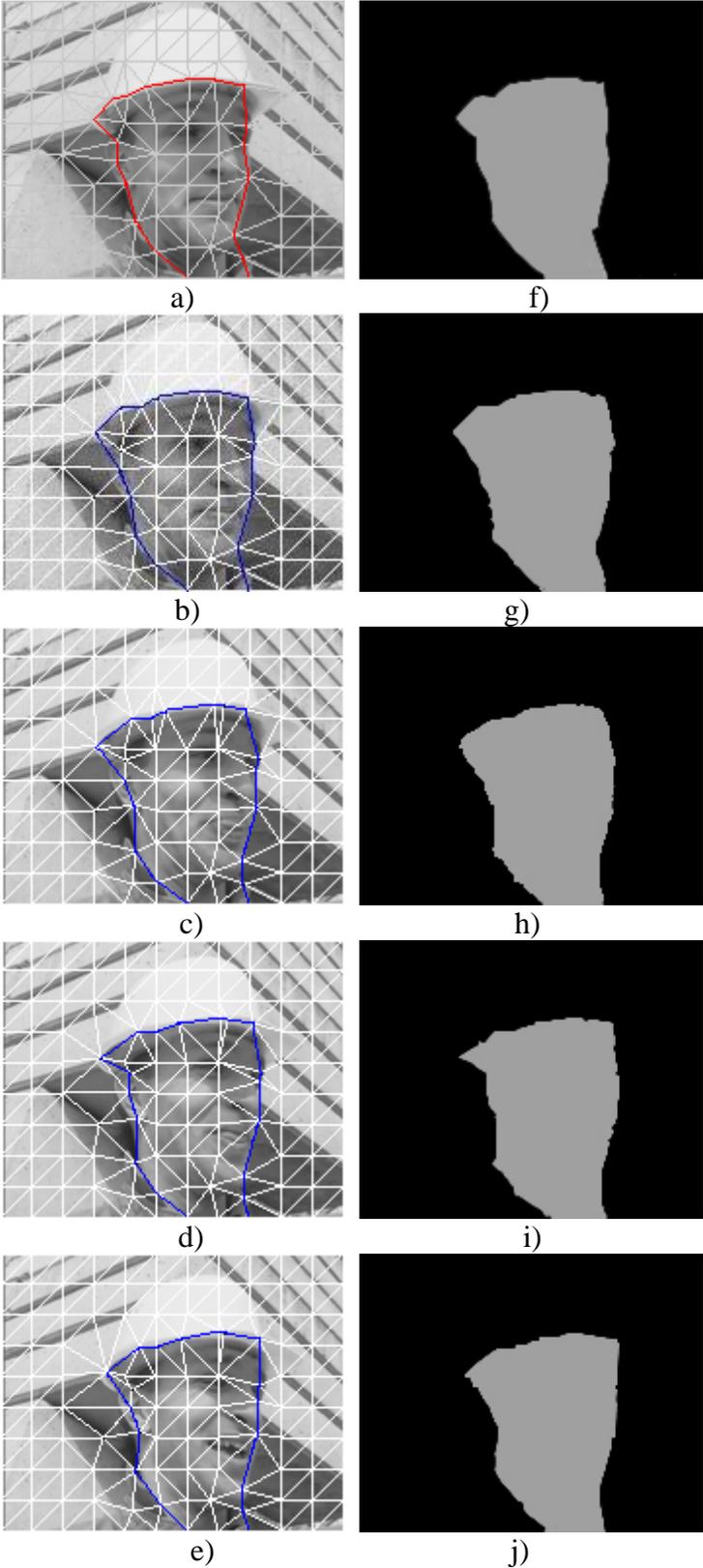
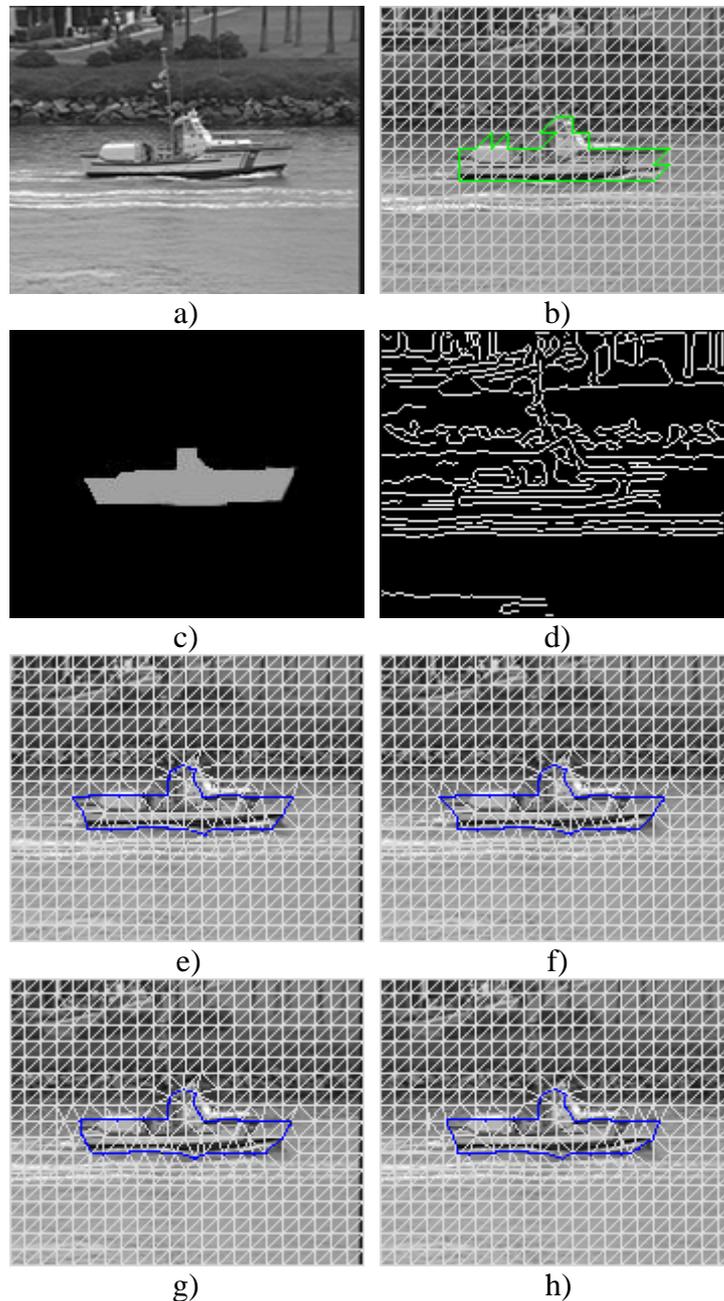


Figure 3.9: segmentation automatique et suivi sur la séquence « foreman »

- La figure 3.10 montre un autre exemple de suivi d'objet sur une séquence réelle. L'image b présente le marquage semi-automatique sur l'image 175 de la séquence costguard (image a), le filtre passe haut appliqué sur cette dernière donne le résultat illustré par l'image d ; un masque de segmentation contenant l'objet a suivre est aussi fourni à l'algorithme (image c).



**Figure 3.10 : suivi d'objets sur la séquence « Coastguard » :**  
- masque de segmentation initiale sur l'image 175 (a)  
- évolution du maillage sur les images 175, 177, 179, 181 (e-h)

L'utilisation des triangles de petite taille affine la segmentation et améliore le suivi d'objet vidéo, en revanche, la forme des cellules dégénère au fur et à mesure des déformations successives, une stratégie de remaillage des zones du maillage trop déformées doit être ajoutée dans l'ensemble de l'algorithme pour apporter plus d'efficacité à la méthode.

- La figure 3.11 illustre le suivi des deux bateaux présents dans la séquence « Coastguard ». Ici le maillage est déformé manuellement sur l'image 101 (a).

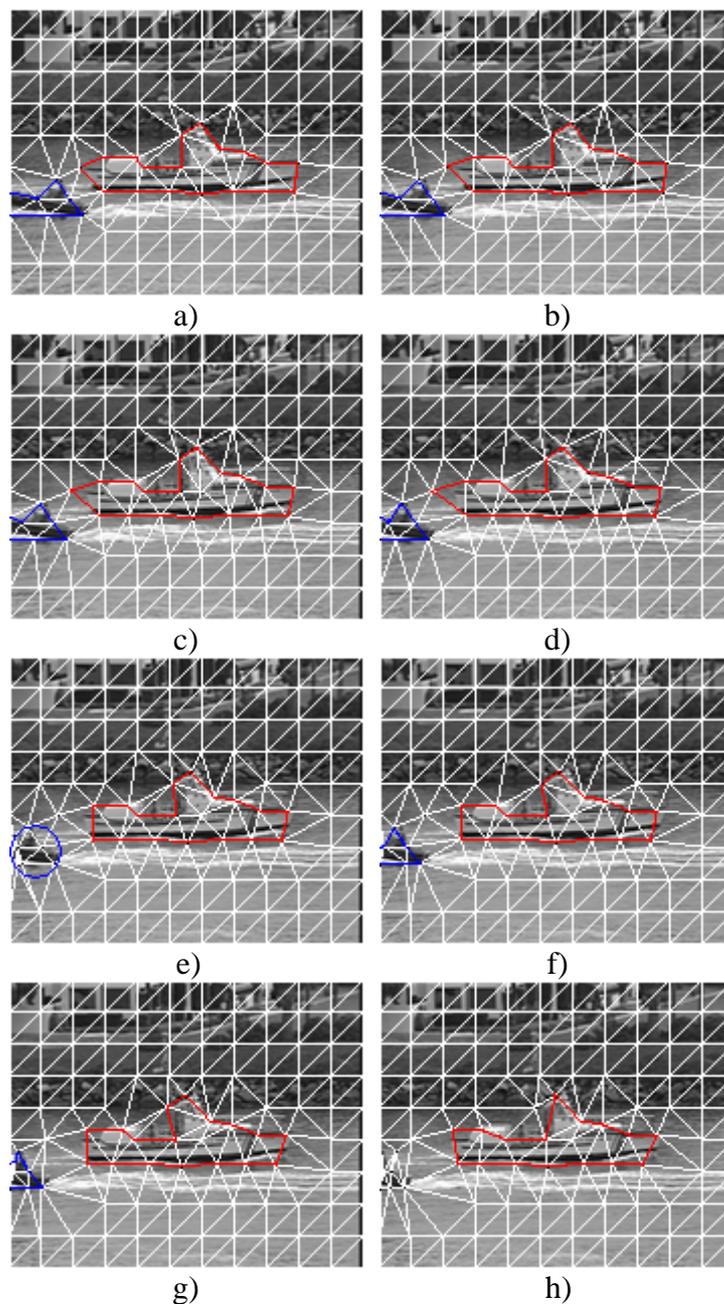


Figure 3.11 : Séquence « coastguard » : segmentation manuelle sur l'image 101 (a), évolution du maillage sur les images 103, 105, 107, 108, 110 et 112 (b-h)

La figure 3.11.e montre la dégénération du maillage dans ça partie gauche à cause des successions de déformation, un remaillage manuel sur la même image conduit au résultat présenté sur la figure 3.11.f (image 108 de la séquence).

Les deux bateaux sont correctement suivis par notre algorithme, si on compte remplacer l'étape de segmentation manuelle sur la première image, un masque de segmentation contenant les deux objets doit être fourni à l'algorithme.

### 3.3 Conclusion

Nous avons présenté dans ce chapitre les résultats obtenus pour la segmentation et le suivi d'objet dans une séquence vidéo en utilisant un maillage actif.

Une estimation de mouvement est mise on œuvre et une contrainte spatial qui force les frontières des objets segmentés à rester dans les zones à fort gradient.

Ainsi, après chaque passe d'optimisation (déformation), l'aspect général des cellules du maillage doit être observé, et les cellules considérées comme dégénérées doivent être remaillées ; l'adaptation de notre algorithme semble donc être une tâche réussite, sachant que l'ajout d'une étape de remaillage automatique est indispensable.

## Conclusion générale

Nous avons présenté dans ce mémoire une méthode de segmentation et de suivi d'objet dans les séquences vidéo en utilisant un maillage actif.

Le travail débute par la création d'un maillage régulier ; l'algorithme le déforme pour créer la carte de segmentation c a d il déplace les nœuds du maillage pour faire coïncider ses arêtes avec les frontières entre les objets.

La déformation du maillage est l'étape fondamentale de l'algorithme. Le résultat désiré de cette opération est d'avoir des cellules homogènes en sens du mouvement et que les arêtes frontières entre les objets vidéo doivent être situés sur les zones à fort gradient spatial.

Nous avons utilisé pour l'estimation du mouvement un algorithme de recherche exhaustive utilisant des blocs triangulaires à différentes tailles.

Le critère spatial introduit dans le terme d'énergie global et le terme relatif au mouvement doivent être précédé par une opération de marquage ; cette dernière a pour but d'éviter d'inclure dans les calculs les arêtes qui n'appartiennent pas à l'objet. Finalement la déformation du maillage se fera principalement par la minimisation de l'énergie globale.

Le maillage résultant sur une l'image courante sera considéré comme maillage initial pour l'image suivante après une étape de prédiction.

Elaborer un algorithme de remaillage est l'une des pistes les plus essentielles pour la poursuite de nos travaux. D'autre part, la prochaine étape serait de développer une méthode de génération d'un maillage tétraédrique qui s'adapte à la forme des régions d'un objet volumique.

## Bibliographie

- [1] G. Marquant, "Représentation par maillage adaptatif déformable pour la manipulation et la communication d'objets vidéo", thèse de Doctorat, Université de Renne I, 2000.
- [2] A. Mahboubi, "Représentation des scènes vidéo par des maillages triangulés", CORESA France, 2006
- [3] I.Danaila, F.Hecht, O.Pironneau, "Simulation numérique en c++ : cours et exercices corrigés", Dunod, Paris 2003.
- [4] P. Alliez, O. Devillers, M. Isenburg et S.Valette, "Compression de maillages, un état de l'art", CORESA, France, 2003
- [5] E. Zhang, "Konstantin Mischaikow, and Greg Turk. Feature-based surface parameterization and texture mapping". ACM Transactions on Graphics, 24(1) :1–27, 2005.
- [6] B. Lévy, S. Petitjean, N. Ray, and J. Maillot. "Least squares conformal maps for automatic texture atlas generation". In SIGGRAPH '02 : Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques, pages 362–371, New York, NY, USA, Jul 2002. ACM Press.
- [7] D. Julius, V. Kraevoy, and A. Sheffer. D-charts : "Quasi-developable mesh segmentation". Computer Graphics Forum, Proceedings of Eurographics 2005, 24(3) pp: 581–590, 2005.
- [8] A. Sheffer. "Model simplifications for meshing using face clustering". Computer-Aided Design, 33(13) pp: 925–934, 2001.
- [9] M.Zilske, H.Lamecker and S.Zachow. "Adaptive Remeshing of Non-Manifold Surfaces". Konrad-Zuse-Zentrum fur Informationstechnik Berlin, January 15, 2008
- [10] I. Li, T. Toon, T. Tan and Z. Huang, "Decomposing polygon meshes for interactive applications". symposium on Interactive 3D graphics, 2001, pp : 35-42.
- [11] P. Parisot, V. Charvillat, G. Mori, "Compensation de Mouvement par Maillage : Apprentissage et Maintien de la Connectivité". Dans : CORESA'04, Lille, 25/05/04-26/05/04, France-Telecom et Enic, pp: 161-164, mai 2004.
- [12] S. Zhu, K. Ma, "A new diamond search algorithm for fast block-matching motion estimation", IEEE Trans. On Image Processing, vol. 9, February 2000.
- [13] C. H. Cheung and L. M. Po, "Novel Cross diamond hexagonal search algorithm for Fast Block Motion Estimation". IEEE Trans. On Multimedia, Vol 7, No 1, Feb 2005.

- [14] S.Valette, "Modèles de maillages déformables 2D et multirésolution surfaciques 3D sur une base d'ondelettes", thèse de Doctorat, institut national des science appliquées de Lion, pp 155, 2002.
- [15] V. Garcia, "Estimation de mouvement subpixélique par blocs adaptée à la couleur avec modèle de mouvement". Thèse de magistère, Université de Nice –Sophia Antipolis, 2004. Rapport de Stage DEA 2004.
- [16] A. Nosratinia, "New kernels for fast mesh-based motion estimation, departement of electrical and computer engeneering, rice university", IEEE trnasactions on circuits and systems for video, Vol. 11, NO. 1, JANUARY 2001.
- [17] S. Valette, I.E. Magnin, R. Prost, "Active mesh for video segmentation and objects tracking", IEEE International Conference on Image Processing ICIP 2001, Thessaloniki, Greece, Vol. II, pp. 77–80, October 7–10-2001.
- [18] G. Marquant, S. Pateux, C. Labit, "Mesh and "Crack Lines": Application to object-based motion estimation and higher scalability", IEEE International Conference on Image Processing ICIP'00, Vancouver, Canada, September 2000.
- [19] N. Cammas, S. Pateux, L. Morin, B. Le Guen, "Estimation de mouvement par maillages et lignes de rupture pour le codage de séquences vidéo", 20th GRETSI Symposium on Signal and Image Procesing, Louvain-la-Neuve, Belgium, September, 2005.
- [20] M. Kass, A. Witkin, D. Terzopoulos, Snakes: "Active Contour Models, International Journal of Computer Vision", 321-331, Boston, 1988.