Estimation et segmentation du mouvement pour le codage vidéo à très bas débit

par

Viet-Nam Dang

Institut National de la Recherche Scientifique
INRS-Télécommunications
16, Place du Commerce, Verdun, Québec, H3E 1H6

Mémoire présenté comme exigence partielle de la
maîtrise ès sciences en télécommunications

Septembre 1995

Remerciements

Tout d'abord, je tiens à remercier le professeur Janusz Konrad, mon directeur de thèse, pour ses conseils, sa rigueur scientifique et la confiance qu'il m'a témoignés tout au long de cette recherche.

J'aimerais remercier également le CRSNG pour son support financier durant mes études de maîtrise.

Merci aussi à Abdol-Reza Mansouri pour la segmentation spatiale des images, Jean-Bernard Chartier pour sa contribution à l'approximation des contours, Gaingar Djelassem pour son aide dans la rédaction de ce mémoire et toute l'équipe du Groupe Communications Visuelles de l'INRS-Télécommunications.

Spéciale dédicace à mes parents.

Sommaire

Ce mémoire décrit une méthode d'estimation et de segmentation du mouvement, susceptible d'être utilisée dans un schéma de codage vidéo à très bas débit, dans lequel l'information de mouvement est transmise sous la forme de régions et paramètres de mouvement.

Le principe général qui a orienté et guidé cette étude repose sur une observation fondamentale: les contours de mouvement coïncident presque toujours avec les contours d'intensité; conséquemment, les régions d'intensité sont des sous-objets des objets en mouvement. Ceci nous a conduit à concevoir et implanter un algorithme qui exploite les cartes de segmentation basée sur l'intensité pour estimer et segmenter le mouvement en trois étapes successives: estimer le mouvement des régions d'intensités; fusionner les régions adjacentes ayant des mouvements similaires; ajuster les contours d'intensité afin qu'ils soient consistants au sens du mouvement.

De nombreuses simulations ont été effectuées afin d'évaluer la qualité des prédictions et de valider l'approche proposée. Les résultats obtenus sont très favorables. La qualité des images prédites est très bonne et ne souffre pas d'effet de bloc comme dans les méthodes traditionnelles. Les cartes de segmentation basée sur le mouvement correspondent raisonnablement bien autant à la forme qu'au nombre des objets physiques de la scène.

D'autre part, la comparaison des résultats de simulation sur les séquences typiques du visiophone ("*Miss America*", "*Carphone*", "*Foreman*") a montré qu'en moyenne, la méthode "basée région" a un gain relatif en terme de "peak prediction gain" (PPG) de 3 à 4dB par rapport à la méthode d'appariement de blocs; tandis que par rapport à la méthode "basée pixel", elle est inférieure d'environ 2dB pour une qualité comparable des mouvements estimés; par contre, notre approche s'est révélée meilleure pour des mouvements globaux (zoom, mouvement de la caméra...). Par ailleurs, la qualité visuelle des images prédites de la méthode "basée région" est équivalente à celle de la méthode "basée pixel" et dépasse de loin celle de la méthode d'appariement de bloc. Cette qualité, qui constitue certainement l'originalité de notre méthode, s'explique par une concentration des erreurs de prédiction à proximité des contours d'intensité où elles sont très peu visibles grâce à l'effet de masque du système de vision humain.

Bien qu'il reste à démontrer que le codage du mouvement (régions et paramètres de mouvement) soit possible à très bas débit, nous anticipons pour conclure qu'avec son pouvoir de dissimuler des erreurs, la méthode "basée région" possède un important atout qui mérite d'être considéré sérieusement, car nous croyons que, même avec un débit restrictif pour leur codage, les erreurs resteraient peu visibles dans l'image encodée; pendant ce temps, les distorsions de la méthode d'appariement de blocs resteraient vraisemblablement présentes; quant à la méthode "basée pixel", il est bien connu qu'une faible erreur de prédiction implique généralement un champ de déplacement peu corrélé; inversement, un champ lisse provoque des erreurs de prédiction élevées; autrement dit, la diminution de l'entropie de l'erreur entraîne une augmentation de celle du mouvement et vice-versa.

Table des Matières

Liste des Figures

Liste des Tableaux

Chapitre 1

Introduction

Le codage vidéo numérique est le sujet d'intenses recherches depuis des années au cours desquelles de nombreux algorithmes de codage ont été proposés. Les uns sont simples et peu performants alors que d'autres offrent de meilleures qualités mais sont plus complexes et difficiles à mettre en œuvre. De plus, ils utilisent diverses structures pour représenter les images codées de sorte qu'il n'y a pas de compatibilités entre ces méthodes. Afin de résoudre le problème d'inter-compatibilité des équipements provenant de divers fabricants, les chercheurs internationaux ont mis beaucoup d'efforts pour établir des standards de codage vidéo. Leurs travaux ont abouti à la mise en place depuis quelques années des standards H.261 et MPEG-1, tandis que l'établissement du standard MPEG-2 est maintenant pratiquement achevé. Ces standards, destinés à une large gamme d'applications exigeant des qualités différentes, permettent de coder les signaux vidéo à des débits allant de $p \times 64\ Kb/s$ $(p = 1, 2, \ldots)$ pour H.261 à quelques Mb/s et plus pour MPEG-1 et MPEG-2 [CCI90, LeG91]. Avec l'adoption de ces standards, le problème du codage vidéo à bas débit est pratiquement résolu, du moins pour un avenir à court terme.

Mais à l'heure actuelle, de nouvelles applications, tels le visiophone, la télésurveillance, les bases de données multimédias interactives..., deviennent envisa-

geables grâces aux développements de nouveaux protocoles et de nouvelles techniques de transmission. Ces services, susceptibles d'être proposés à large échelle, laissent entrevoir des perspectives économiques importantes dans les années à venir. Mais pour être viables, ils doivent offrir une qualité d'image acceptable, tout en utilisant des médias de transmission à très bas débit, tels les lignes téléphoniques, les canaux radio à bande étroite de la communication mobile ou des systèmes de stockage à faible capacité. À ce stade de la recherche, il n'existe pas encore de méthodes de codage éprouvées, spécialement développées pour satisfaire ces exigences particulièrement contraignantes. Le problème de codage à très bas débit reste donc largement ouvert à toutes nouvelles propositions.

Il est bien connu que les standards mentionnés ci-dessus ne donnent pas une qualité d'image acceptable à très bas débit (de l'ordre de quelques dizaines de Kb/s). En effet, ces méthodes adoptent une segmentation arbitraire de l'image en blocs carrés de même taille. Bien qu'ils soient très pratiques pour le codage, ces blocs, dont la forme ne correspond pas à celle de la plupart des objets de la scène, provoquent des artefacts de codage bien connus, par exemple, l'effet de bloc ou le phénomène "mosquito noise". À très bas débit, l'indispensable diminution du nombre de bits alloués pour coder l'erreur résiduelle fait en sorte que ces artefacts deviennent très visibles et agaçants à l'œil, puisque le système de vision humain (SVH) est particulièrement sensible à ces effets. Par ailleurs, la segmentation arbitraire de l'image en blocs et l'hypothèse sous-jacente du mouvement translationnel ne permettent pas de prendre en compte la discontinuité du mouvement entre les objets de la scène, ce qui empêche d'obtenir de bonnes estimations du mouvement. Ces problèmes fondamentaux expliquent d'ailleurs les difficultés majeures rencontrées dans l'adaptation de ces méthodes pour le codage à très bas débit.

Étant donné les limites des méthodes traditionnelles, la communauté scientifique de chercheurs en compression vidéo se mobilise de nouveau à la recherche de nou-

velles méthodes plus efficaces pour le codage vidéo à très bas débit. Il s'agit d'une tâche très difficile qui devient, dans le cadre du projet MPEG-4 [Sch93, RC94], le sujet d'intenses recherches dans de nombreuses institutions universitaires et privées.

Récemment, une nouvelle technique de codage dite "basée région" fut proposée comme une alternative aux anciennes méthodes [MHO89, Die91, Sti94]. Cette nouvelle voie de recherche abandonne la segmentation arbitraire en blocs et la remplace par une segmentation de l'image en régions ou objets de forme quelconque dont l'intensité est quasi-uniforme dans le cas du codage intra-trame, et dont le mouvement est homogène dans le cas du codage inter-trame. Dans le contexte du codage à très bas débit, la texture et le mouvement des régions pourraient être représentés par des descripteurs compacts issus des modèles plus performants que ceux utilisés dans la méthode des blocs, afin de permettre un taux de compression élevé tout en préservant une qualité d'image acceptable. Après traitement au niveau de l'émetteur, la carte de segmentation et ses descripteurs seront codés et envoyés au récepteur.

Du point de vue conceptuel, cette approche paraît très attrayante. En effet, la notion de segmentation en régions qui coïncident éventuellement avec les objets réels de la scène permettrait d'éviter les problèmes de discontinuité du mouvement et de l'intensité qui sont, comme nous l'avons déjà mentionné, les faiblesse inhérentes à la structure de bloc. Aussi, il est à prévoir que si la segmentation était assez bonne, les artefacts de codage seraient concentrés au voisinage de la frontière des objets où ils seront beaucoup moins visibles, grâce à l'effet de masque du SVH [Jai89]. Bref, le codage basé région offre des possibilités intéressantes. Cependant, pour pouvoir exploiter tous ces avantages, le codage "basé région" a besoin de méthodes d'analyse efficaces, capables de segmenter et de traiter correctement l'image au niveau des régions.

Ce mémoire, motivé par les activités de MPEG-4, traite le problème d'estimation et de segmentation du mouvement dans des séquences d'images pour le codage vidéo

à très bas débit, ou plus précisément, pour le codage inter-trame basé région utilisant la prédiction compensée par le mouvement. L'objectif principal de notre étude est de mettre au point un algorithme de segmentation des images en régions homogènes au sens du mouvement, et de fournir l'ensemble des descripteurs de mouvement de ces régions, et ce, en exploitant la segmentation basée sur l'intensité fournie par un module indépendant.

Ce rapport est divisé en 6 chapitres. Le chapitre 2 présentera quelques principes fondamentaux en estimation et en segmentation du mouvement ainsi que des concepts de base en codage vidéo. Le troisième chapitre fera un survol des méthodes proposées dans la littérature concernant l'estimation et la segmentation du mouvement. Un nouvel algorithme d'estimation et de segmentation du mouvement exploitant la segmentation spatiale sera proposé au chapitre 4. Il s'agira d'une procédure permettant de résoudre le problème d'estimation et de segmentation en trois étapes successives: estimation du mouvement des régions d'intensité, fusion des régions adjacentes ayant des mouvements similaires et ajustement des contours de régions afin qu'ils soient conformes avec le mouvement. L'originalité de cette méthode réside dans le fait qu'elle est capable de concentrer les erreurs de prédiction au voisinage des contours d'intensité où elles sont peu visibles grâce à l'effet de masque du SVH. De plus, au cours de ce chapitre, la modélisation et la résolution du problème de fusion et de l'ajustement des contours seront adressées en détails pour la première fois, ce qui constitue certainement notre contribution aux efforts visant à résoudre le problème de codage "basé région". Dans le chapitre 5, nous évaluerons la performance de cette nouvelle méthode, les résultats de simulation seront analysés et comparés avec ceux des deux méthodes d'appariement de blocs et "basée pixel". Précisons ici qu'à défaut d'avoir un schéma de codage complet, cette évaluation de performance ne portera que sur la qualité des prédictions et non sur les images décodées. Dans ce même chapitre, nous explorerons également la possibilité de coder avec

perte les contours de mouvement; pour cela nous étudierons l'impact de ce codage sur la qualité des images reconstruites par compensation de mouvement. Enfin, le chapitre de conclusion résumera les principaux éléments de cette étude et proposera des améliorations envisageables.

Chapitre 2

Concepts de base

Ce chapitre a pour but de présenter les principes de base du codage vidéo ainsi que les concepts fondamentaux en estimation et en segmentation du mouvement qui jouent un rôle important dans le codage inter-image. Nous décrivons, dans la première section, les principaux composants d'un système de codage vidéo. Les sections suivantes présentent les notions de bases de l'estimation et de la segmentation du mouvement. Nous essayons également de mettre en évidence les caractéristiques du problème que nous avons à résoudre, afin d'éclairer nos choix quant à la méthodologie servant de base à la méthode d'estimation et de segmentation du mouvement que nous présenterons dans les chapitres suivants.

2.1 Principes de base d'un système de codage vidéo

La représentation numérique des images à l'état brut requiert une grande quantité d'informations qui sont fortement corrélées. La transmission ou le stockage direct des images constitue donc un gaspillage de ressources, et parfois est impossible à réaliser, puisque la quantité d'informations est si grande qu'elle dépasse la capacité

des systèmes de transmission ou de stockage. À titre d'exemple, la transmission à une fréquence de 10 Hz/s d'une séquence d'images au format CIF-352 $\times$ 288 (Common Intermediate Format) ou QCIF-176 $\times$ 144 (Quarter CIF) nécessite des débits approximatifs de 16 Mb/s et 4 Mb/s respectivement (avec une quantification à 8 bits pour la luminance et 8 bits pour les deux composantes de chrominance), alors que les lignes téléphoniques n'ont qu'une largeur de bande limitée à 64 Kb/s. On doit donc comprimer les images, afin que leur transmission ou stockage soit possible et efficace.

Le but du codage est de trouver une autre forme de représentation des signaux vidéo permettant d'éliminer le plus possible les informations superflues à l'intérieur d'une image ou encore, entre les images d'une même séquence. Il existe deux techniques de codage qu'on peut utiliser pour coder une séquence d'images, soient le codage intra-trame et le codage inter-trame.

2.1.1 Codage intra-trame

Développé dans le but de coder les images fixes, ce type de codage cherche à éliminer la redondance spatiale au sein de l'image. En général, on applique une transformation orthogonale de type DCT, KLT ou Fourier sur les régions de l'image préalablement segmentée soit arbitrairement en blocs de forme et de taille régulière, soit selon un critère basé sur la luminance. Cette transformation a pour but de concentrer l'énergie du signal dans un petit nombre de coefficients. Dans les codeurs de type DCT/DPCM, les coefficients significatifs seront utilisés pour prédire la texture originale. L'erreur de prédiction, les coefficients de la transformée et la carte de segmentation constituent une version comprimée de l'image à coder et seront envoyés au récepteur.

La complexité et la performance de ce type de codage dépendent non seulement de la transformation utilisée mais aussi de la segmentation. Une segmentation de

l'image en régions de forme quelconque, en tenant compte de la distribution de la texture, permettra de mieux exploiter la propriété de stationnarité et d'augmenter ainsi la capacité de compression; mais elle est complexe et difficile à réaliser en pratique. Par contre, une segmentation a priori de l'image en blocs carrés, comme dans le standard H.261, facilite grandement le codage, puisqu'il n'est pas nécessaire de transmettre l'information de la segmentation. En revanche, une telle structure est inefficace en terme de compression à cause des problèmes de discontinuité et de stationnarité à l'intérieur d'un bloc.

2.1.2 Codage inter-trame

Le codage inter-trame vise à réduire la redondance temporelle entre les images successives dans une séquence. En principe, connaissant la trajectoire de mouvement de chaque point de l'image, on pourrait effectuer la compression en ne transmettant que la première trame et l'information sur le mouvement. Pour reconstruire la séquence au décodeur, il suffirait de propager chaque point le long de sa trajectoire. Cependant, le déplacement des pixels n'est pas directement disponible et ne peut être connu de façon exacte [DK93]. On doit faire appel à un module qui estime le mouvement à partir des observations. Le déplacement estimé est utilisé ensuite dans une prédiction compensée par le mouvement au codeur. L'erreur de prédiction et l'information du mouvement représentent l'information nouvelle (innovation) entre deux images. Elles seront codées et envoyées au décodeur.

Le débit de transmission pour le codage inter-trame dépend directement de la méthode d'estimation et de segmentation du mouvement utilisée. Les méthodes basées sur une structure de bloc n'ont besoin que de très peu de bits pour coder le mouvement, parce que la segmentation est connue au récepteur et qu'il y a seulement un vecteur de déplacement par bloc; par contre, l'erreur de prédiction est élevée et nécessite un codage avec beaucoup de bits afin d'assurer une qualité d'image

acceptable. Inversement, dans le cas des méthodes du champ dense (un vecteur par pixel), on peut transmettre très peu d'informations pour l'erreur de prédiction, mais il en faut beaucoup plus pour le mouvement; cela vient du fait que l'erreur résiduelle de ces méthodes est en général très faible, et que le déplacement d'un pixel doit être représenté par un vecteur à deux composantes. Quant aux méthodes basées sur une segmentation en régions ou objets de forme quelconque selon un critère de mouvement, on espère que le débit alloué à l'une ou à l'autre des deux composantes (l'erreur de prédiction et le mouvement) se situe entre ces deux extrêmes.

2.1.3 Codage "basé région"

Bien qu'il soit possible d'utiliser séparément l'un ou l'autre type de codage décrit ci-dessus pour coder des séquences d'images, on les combine en pratique dans un schéma de codage qui alterne ces deux modes intra et inter-trame afin d'obtenir des taux de compression élevés. La figure 2.1 présente le diagramme-bloc d'un codeur prédictif "basé région" utilisant ces deux techniques de codage. Le principe de ce codeur repose sur une représentation de la scène en objets ou régions d'intensité uniforme dans le cas du codage intra-trame et de mouvement homogène dans le cas du codage inter-trame. Son fonctionnement peut être résumé comme suit: d'abord, on effectue un sous-échantillonnage temporel de la séquence originale, afin d'en réduire la fréquence de trames (la fréquence de trames des applications à très bas débit est typiquement 10 Hz/sec). On y applique ensuite la technique de codage intra ou inter-trame, de sorte qu'on ait à la sortie de ce codeur deux types de trame, comme le montre la figure 2.2. Les trames codées en mode inter-trame (notée trame P) y sont majoritaires, puisqu'elles permettent un meilleur taux de compression. Les trames intra (notée trame I) sont insérées au début de chaque séquence et aussi de

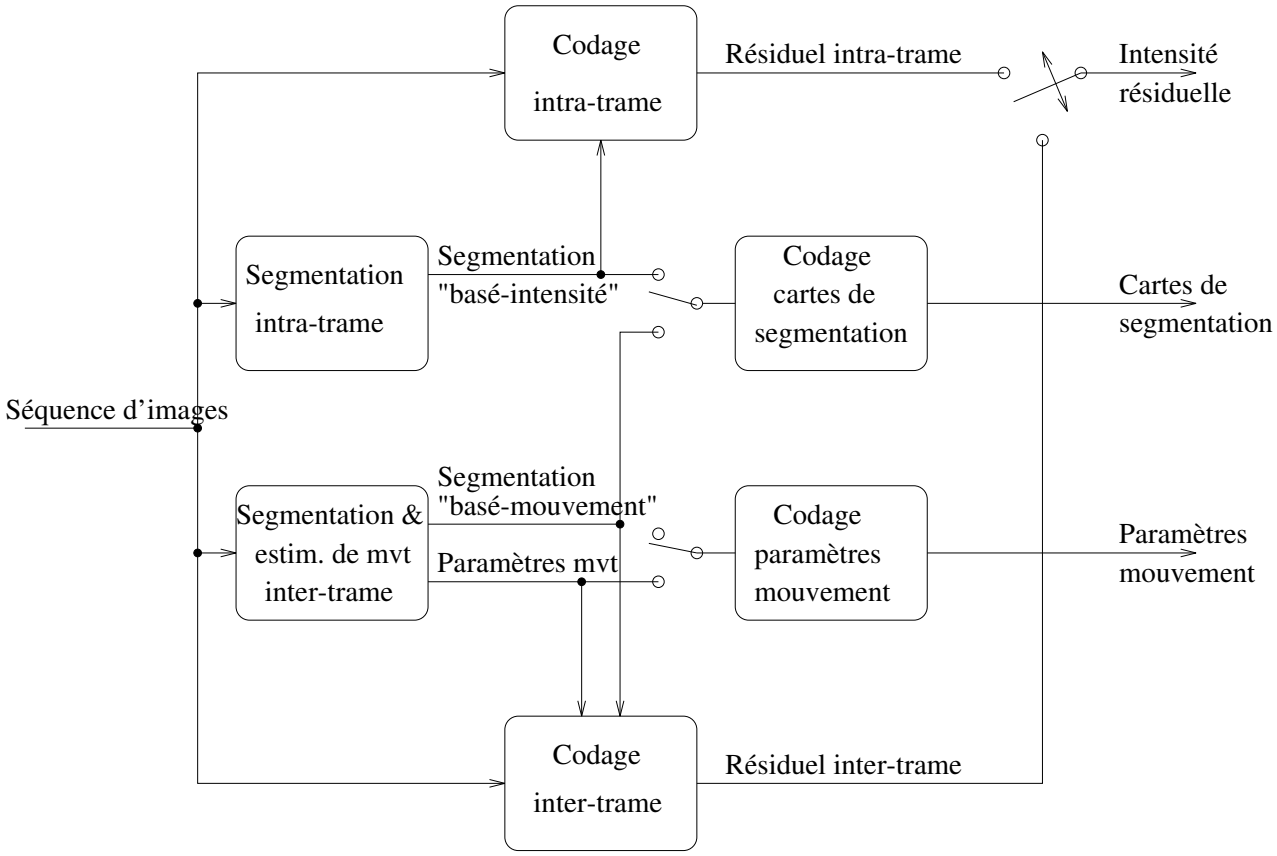


FIG. 2.1 - Exemple de diagramme d'un système de codage vidéo "basé-région" pour des applications à très bas débit

façon parsemée à l'intérieur d'une séquence. Le rôle de ces trames I est multiple:

- celles qui sont insérées au début de chaque séquence (après un changement de scène) se servent de trames de référence pour la reconstruction au récepteur;
- celles qui se trouvent à l'intérieur d'une séquence empêchent, d'une part, la propagation et l'accumulation des erreurs qui pourraient dégrader la qualité d'image; d'autre part, elles rendent possible des fonctionnalités nécessaires lors de l'édition; par exemple, l'accès aléatoire et la recherche en avant ou en arrière.

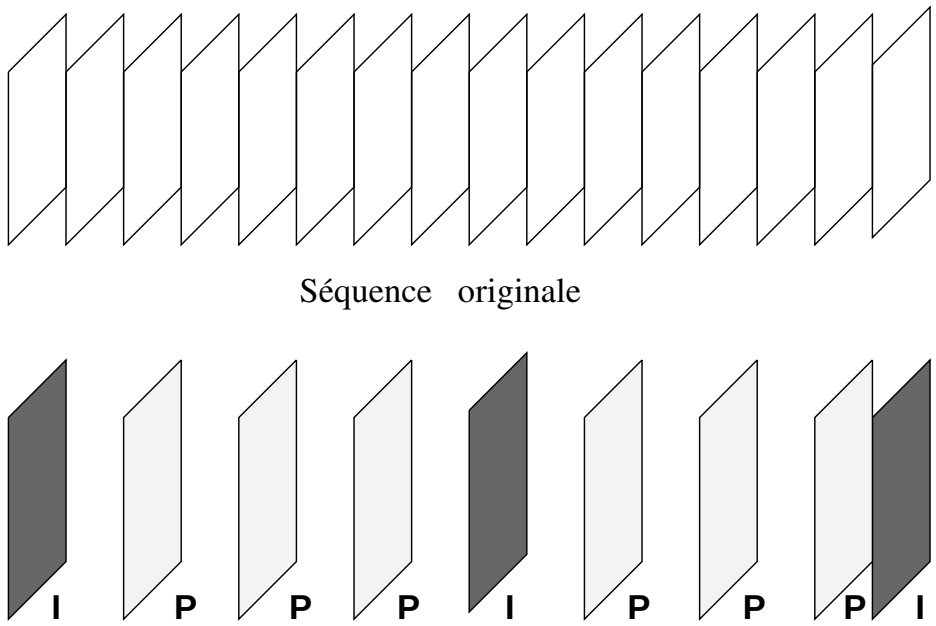


FIG. 2.2 - Deux types de trame d'une séquence codée

Il est à noter que l'on peut utiliser les informations obtenues lors du codage intra-trame pour renforcer le codage inter-trame. En effet, la segmentation intra-trame est une source d'information très utile pour traiter le problème de discontinuité de mouvement, car on observe que ces discontinuités coïncident presque toujours avec les contours d'intensité [HKLM88, KD89].

2.2 Notions fondamentales en estimation et en segmentation du mouvement

2.2.1 Mouvement apparent 2D

Une image est formée par la projection d'une scène tri-dimensionnelle dans le plan focal de la caméra. Sous l'hypothèse qu'il n'existe pas d'objets transparents, chaque point de l'image correspond à un point de la surface d'un des objets de la scène. Le mouvement relatif entre la caméra et la scène induit un mouvement des points de l'image. Ce mouvement se manifeste par des variations spatio-temporelles dans l'intensité de l'image. C'est l'observation de ces variations qui permet de déduire le mouvement apparent des points ou des objets de l'image. Il faut distinguer ici les méthodes d'estimation du mouvement apparent 2D pour fin de codage de celles qui cherchent à analyser quantitativement ou qualitativement le mouvement 3D pour des applications en vision artificielle. Les premières cherchent simplement le déplacement décrivant le correspondant d'un point dans deux images adjacentes, alors que les dernières s'intéressent au mouvement physique des objets de la scène. Précisons ici que notre étude concerne seulement le mouvement apparent 2D.

Il y a des situations où il est impossible de déduire le mouvement à partir des observations de l'intensité de l'image. Par exemple, une région uniforme en déplacement apparaîtra stationnaire; de même, le mouvement d'un motif périodique, d'une quantité égale à la période ne provoquera pas de mouvement apparent.

2.2.2 Champ de déplacement

On appelle le champ de déplacement l'ensemble des vecteurs, associé à chaque point de l'image, décrivant le déplacement des pixels entre deux trames successives. Le champ de déplacement établit donc une correspondance entre les points d'une

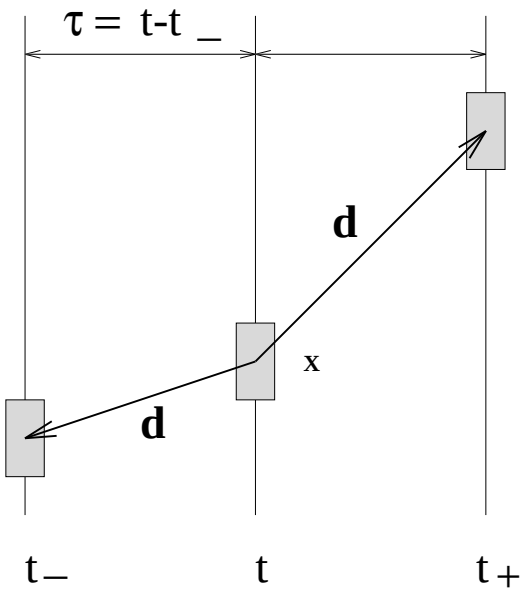


FIG. 2.3 - Exemple de champ de déplacement en avant et en arrière

image au temps t et l'image précédente au temps t_- (champ de déplacement en arrière) ou l'image suivante au temps t_+ (champ de déplacement en avant). La figure 2.3 illustre les déplacements en avant et en en arrière d'un point.

En général, le déplacement d'un point suit une trajectoire non linéaire dans le temps. Mais si la fréquence temporelle des trames dans une séquence d'images est suffisamment grande, le déplacement d'un point entre deux trames successives peut être considéré comme une translation. Dans cette étude, nous assumons que cette hypothèse est toujours vraie. Le lecteur intéressé au mouvement dont la trajectoire est non linéaire peut se référer à [CK95].

Il peut être impossible de trouver le correspondant d'un point dans une image adjacente dû au phénomène d'occlusion qui survient quand un objet en mouvement est occulté par un autre objet se trouvant en devant-scène. Une illustration de ce phénomène est présentée à la figure 2.4; on peut y voir qu'une région découverte suite au mouvement d'un objet n'a pas de correspondant dans l'image précédente. Évidemment, l'estimation du mouvement basée sur la variation de l'intensité de

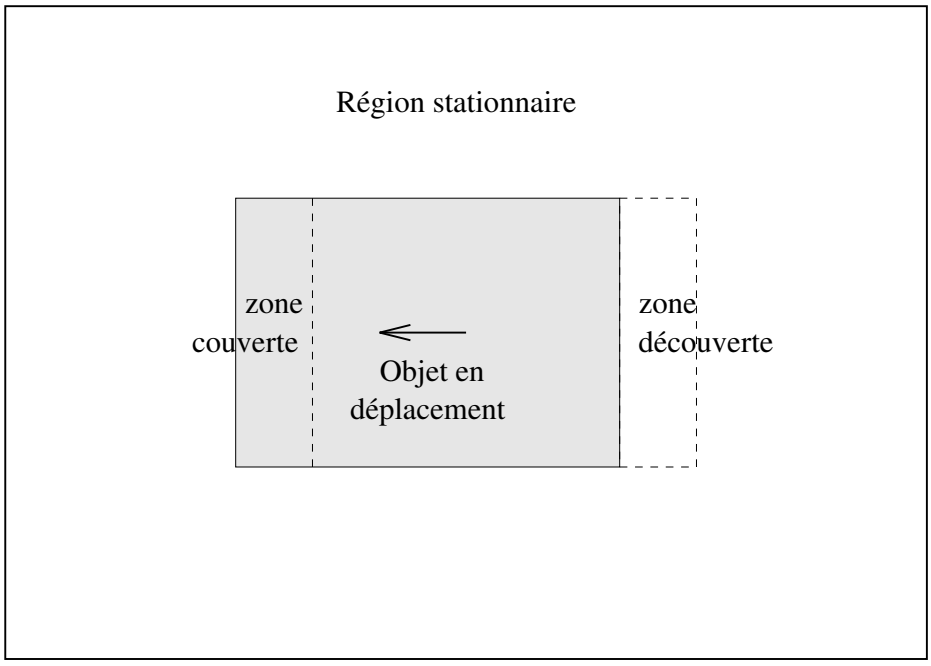


FIG. 2.4 - Exemple de phénomène d'occlusion; en pointillé: objet au temps t_-, en continu: objet au temps t

cette région sera fausse.

Il y a une forte corrélation entre les vecteurs de déplacement voisins. Ceci découle du fait qu'à l'intérieur d'un même objet, tous les points se déplacent de la même façon. Cependant, aux frontières des objets, il peut y avoir des discontinuités du mouvement. Le champ de déplacement est donc généralement lisse mais comporte des discontinuités.

2.2.3 Équation de la contrainte spatio-temporelle

Fondamentalement, le mouvement n'est pas directement mesurable dans une séquence d'images; on peut y voir seulement les effets du mouvement et non le mouvement lui-même. Par conséquent, il doit être estimé à partir d'un modèle structurel qui le relie aux observations. Ce modèle s'appuie sur une hypothèse fondamentale qui consiste à supposer que l'intensité d'un point reste constante le long de sa trajec-

toire, et que toute variation de l'intensité dans une séquence d'images est uniquement causée par le mouvement. Mathématiquement, cette hypothèse est exprimée dans [HS81] comme suit: soit $g(\mathbf{x}, t)$ la fonction d'intensité (continue) d'un point qui se trouve à la position $\mathbf{x} = (x, y)^T$ au temps t. Si cette intensité est constante dans le temps alors,

$$\frac{d}{dt} g(\mathbf{x}, t) = 0 \tag{2.1}$$

Par la règle de dérivation en chaîne on a

$$\frac{\partial g}{\partial x}\frac{dx}{dt} + \frac{\partial g}{\partial y}\frac{dy}{dt} + \frac{\partial g}{\partial t} = 0 \tag{2.2}$$

Posons

$$u = \frac{dx}{dt} \qquad \text{et} \qquad v = \frac{dy}{dt} \tag{2.3}$$

qui sont les vecteurs de vitesse horizontal et vertical. En les remplaçant dans l'équation (2.2), on obtient une équation linéaire à deux inconnus u et v

$$g_x u + g_y v + g_t = 0 \tag{2.4}$$

où g_x, g_y sont les gradients spatiaux, g_t est le gradient temporel (la notation a été simplifiée par soucis de clarté).

Cette équation, appelée *équation de la contrainte spatio-temporelle*, indique que le mouvement d'une région contenant beaucoup de détails provoque beaucoup de changements temporels. Inversement, le mouvement d'une région dont l'intensité est presque uniforme n'induit que très peu de changements temporels. Notons qu'elle ne permet de déterminer que la composante parallèle au gradient d'intensité du vecteur de vitesse [HS81].

Dans le cas des images discrètes, l'hypothèse d'invariance de l'intensité peut être exprimée sous une forme impliquant directement le champ de déplacement. En effet, si $\mathbf{d}(\mathbf{x}) = (d_x, d_y)^T$ est le déplacement en arrière d'un point se trouvant à la position $\mathbf{x}$ au temps t, alors la position de ce point au temps t_- est $\mathbf{x} - \mathbf{d}(\mathbf{x})$. Puisque son

intensité reste constante, on peut écrire

$$g(\mathbf{x}, t) - g(\mathbf{x} - \mathbf{d}(\mathbf{x}), t_-) = 0 \tag{2.5}$$

La différence du terme à gauche est appelée "différence inter-pel déplacée" et dénotée souvent DPD qui vient d'une terminologie anglaise "Displaced Pixel Difference". Il s'agit de l'erreur de compensation par le mouvement du point $\mathbf{x}$. Notons que la somme de l'erreur de tous les points de l'image est appelée "différence inter-trame déplacée ("Displaced Frame Difference") et notée DFD. Ces deux fonctions sont largement utilisées dans des algorithmes d'estimation du mouvement pour fin de codage, car elles permettent de minimiser directement l'erreur de reconstruction.

L'hypothèse de l'invariance de l'intensité sous la forme de l'équation (2.5) ou (2.4) ne mène pas à une solution unique, car pour chaque point de l'image, il faut déterminer les deux composantes du vecteur de déplacement ou de vitesse à partir d'une seule équation. Il s'agit en fait d'un problème mal posé [BPT88]. Pour résoudre ce genre de problème, on peut utiliser des techniques de régularisation qui consistent à poser d'autres contraintes sur le champ de déplacement. En général, ces hypothèses ne sont pas toujours valides. Bien entendu, la qualité du mouvement estimé dépend de la nature de l'hypothèse utilisée. Nous reviendrons sur ce point lors de la description des méthodes d'estimation du mouvement au chapitre 3.

Il est également important de préciser que, même si l'hypothèse d'invariance de l'intensité est utilisée explicitement ou implicitement dans presque toutes les méthodes d'estimation du mouvement, elle n'est pas néanmoins toujours valide. Effectivement, la variation de l'intensité d'image n'est pas uniquement causée par le mouvement; d'autres facteurs, tels le changement d'illumination, les occlusions, le bruit intrinsèque du matériel vidéo, etc., peuvent provoquer aussi des changements dans l'intensité de l'image. Il faut donc être conscient que les méthodes d'estimation du mouvement pour fin de codage utilisant l'hypothèse d'invariance de l'intensité ne donnent qu'une estimation du mouvement apparent dans le but de réduire la

quantité d'informations à transmettre ou à stocker. Une estimation sera considérée comme correcte non pas parce qu'elle correspond au mouvement physique, mais parce qu'elle permet de minimiser l'erreur de reconstruction de l'image ou de la région par les méthodes de compensation de mouvement.

2.2.4 Région homogène au sens du mouvement

Dans un schéma de codage vidéo qui nécessite la transmission de l'information du mouvement, le champ de déplacement doit être représenté de façon optimale. Une des méthodes possibles serait une segmentation de l'image en régions de mouvement homogène.

Une région homogène au sens du mouvement peut être définie comme étant une région dans laquelle ne figure qu'un type de mouvement 2D [Nic92]. En d'autres termes, une région homogène au sens du mouvement sera une région où le modèle de mouvement choisi décrit correctement le mouvement apparent des pixels contenus dans cette région. Il est important de préciser ici que ce critère ne garantit pas l'obtention des objets physiques contenus dans la scène, car un objet en mouvement trop complexe sera divisé en plusieurs sous-objets, afin que le modèle soit vérifié; d'autre part, deux ou plusieurs objets physiquement distincts, mais ayant des mouvements similaires, pourraient être regroupés ensemble si le modèle était capable de décrire leur mouvement avec les mêmes paramètres.

Bien sûr, plus un modèle de mouvement est complexe, mieux il sera capable de décrire des mouvements compliqués. Il permettra d'éviter une trop grande sur-segmentation des objets réels de la scène, et d'obtenir une carte de segmentation plus simple (contenant peu de régions) qui nécessitera peu de bits pour le codage de ses contours. Cependant, un modèle de mouvement trop complexe est très difficile à utiliser en pratique à cause des difficultés de calcul; de surcroît, il peut être même une source de gaspillage en terme de débit de transmission. En effet, dans une

scène contenant plusieurs objets en déplacement dans différentes directions, si le mouvement de ces objets est tellement simple (translation par exemple) qu'il suffisse de quelques paramètres pour le décrire, alors non seulement l'utilisation d'un modèle complexe ne permet pas de regrouper ces objets afin de réduire le débit réservé aux contours de mouvement, mais elle exige la transmission inutile des paramètres superflus. Donc, le modèle de mouvement doit faire l'objet d'un choix judicieux, en considérant la complexité de la scène et des calculs, ainsi que le nombre de bits nécessaires au codage de l'information de mouvement.

Chapitre 3

Méthodes d'estimation et de segmentation du mouvement

Nous avons exposé au chapitre 2 les concepts et les problèmes fondamentaux associés à l'estimation et à la segmentation du mouvement. Nous examinons dans ce chapitre les principales méthodes proposées dans la littérature permettant d'analyser le mouvement apparent 2D. Nous présentons dans la première section les méthodes d'estimation du mouvement. La section suivante décrit les méthodes qui combinent l'estimation et la segmentation dans une procédure jointe.

3.1 Méthodes d'estimation du mouvement

Les méthodes d'estimation du mouvement peuvent être divisées en deux groupes: les méthodes fréquentielles et celles du domaine spatio-temporel. Les sections suivantes décrivent successivement chacune de ces deux familles de méthodes tout en montrant leurs avantages et leurs faiblesses au point de vue codage à très bas débit.

3.1.1 Méthodes fréquentielles

Ces méthodes travaillent dans le domaine des transformées (par exemple la transformée de Fourier). Elles sont basées sur l'observation des effets d'un mouvement 2D de l'image sur ses composantes fréquentielles. Quelques uns de ces effets sont très intéressants; par exemple, une translation spatiale correspond à un décalage de phase entre deux images successives ou se traduit par une inclinaison du plan du spectre tri-dimensionnel[Tho87, JW87]. Donc, une mesure de la variation de phase ou la détermination du plan sur lequel se concentre l'énergie du spectre de Fourier permet de déduire le mouvement translationnel. L'analyse fréquentielle peut se faire sur des blocs, mais ceux-ci doivent avoir une taille suffisante pour le calcul de la transformée.

Cette approche permet de retrouver le mouvement global d'un bloc. Elle est utilisée pour déterminer le déplacement translationnel d'un objet devant un fond fixe. L'extension de ces méthodes à des cas de mouvements complexes, telles les séquences à mouvement naturel, est difficile.

3.1.2 Méthodes du domaine spatio-temporel

Il est bien connu que l'estimation du mouvement 2D à partir des observations de l'intensité dans une séquence d'image est un problème mal posé. Rappelons que le modèle structurel, qui traduit l'hypothèse de l'invariance de l'intensité sous la forme DPD ou celle de la contrainte spatio-temporelle, ne permet de déterminer que la composante parallèle au gradient de la luminance [HS81]. Pour lever l'indétermination, il faut recourir à des techniques de régularisation qui consistent à poser et à modéliser certaines hypothèses sur le champ de déplacement. Ces hypothèses ne sont pas toujours valides, et évidemment, la performance d'une méthode d'estimation dépend directement de la validité de l'hypothèse impliquée. Beaucoup de solutions ont été proposées depuis les deux dernières décennies, mais en général,

elles suivent essentiellement deux approches:

1. La première cherche à décrire le mouvement d'une région ou d'un objet de la scène par un modèle paramétrique. Le mouvement estimé dans ce cas est global pour toute la région (ou l'objet). Dans le reste de ce mémoire, nous appelons parfois les méthodes utilisant cette approche par "méthodes basées régions".

2. La deuxième approche est locale, elle vise à exploiter la corrélation spatiale du champ de déplacement et produit un vecteur de déplacement pour chacun des points de l'image. Pour mettre l'accent sur sa différence avec les méthodes d'estimation "basée région", on appelle ces méthodes par "méthodes basées pixel".

Nous présentons dans la suite de cette section les principales méthodes utilisant ces deux approches.

3.1.2.1 Méthodes basées sur le modèle paramétrique

Cette technique de régularisation consiste à segmenter l'image en blocs (ou suppose qu'on connaît déjà une segmentation au sens du mouvement) et à y associer un modèle de mouvement pré-défini ayant un certain nombre de paramètres. Ce modèle sert de contrainte de mouvement pour les pixels contenus dans la région. L'intégration du modèle de mouvement dans le modèle structurel (DPD) permet d'estimer les paramètres de mouvement d'une région à partir des observations. D'une manière plus formelle, l'estimation du mouvement d'une région, notée R_n, revient à minimiser une fonction mesurant l'erreur de prédiction compensée par le mouvement:

$$\min_{\{\phi_n\}} \sum_{\mathbf{x}\in R_n} [g(\mathbf{x},t) - g(\mathbf{x} - h(\mathbf{x},\phi_n), t_-)]^2, \qquad n = 1, ..., N \tag{3.1}$$

où ϕ_n est le vecteur de paramètres de mouvement; $h(\mathbf{x},\phi_n) = (d_x, d_y)^T$ est le modèle de mouvement et N est le nombre de régions dans l'image.

Cette optimisation peut être interprétée comme un appariement de régions entre deux trames successives. Mathématiquement, il s'agit de résoudre un système d'équations sur-déterminé. Des méthodes itératives sont proposées dans [MHO89, Nic92, Die91] pour résoudre ce problème. En général, les paramètres de mouvement d'une région sont déterminés par l'équation itérative suivante:

$$\Phi^{i+1} = \Phi^i - \epsilon \mathbf{G}^i \tag{3.2}$$

où $\mathbf{G}^i$ est le terme correctif à la i^{ieme} itération qui dépend du modèle de mouvement utilisé. On peut trouver dans ces références les équations analytiques définissant ce terme pour différents modèles ayant de 2 à 12 paramètres.

La performance de ces méthodes d'estimation du mouvement dépend essentiellement de deux facteurs: la segmentation et le modèle de mouvement utilisés. Beaucoup de méthodes d'estimation utilisant différents modèles de mouvement ont été proposées. En général, ces modèles sont obtenus par la projection du mouvement 3D d'un objet de la scène dans le plan d'image moyennant quelques hypothèses simplificatrices sur la forme des objets ou sur le type de leur mouvement.

Tsai et Huang proposent dans [TH81] un modèle non linéaire à huit paramètres

$$\begin{pmatrix} d_x \\ d_y \end{pmatrix} = \begin{pmatrix} \frac{a_1 x + a_2 y + a3}{a_7 x + a_8 y + 1} \\ \frac{a_1 x + a_2 y + a3}{a_7 x + a_8 y + 1} \end{pmatrix} \tag{3.3}$$

Ce modèle est capable de décrire le mouvement en translation, rotation et déformation des objets plans et rigides. Ils démontrent que les paramètres du modèle sont déterminés par un système d'équations linéaires obtenu à partir des observations de deux trames.

Afin de couvrir une plus large classe d'objets que les objets plans, Diehl propose dans [Die91] un modèle plus complexe à 12 paramètres (modèle quadratique) pour décrire la structure et le mouvement d'une surface parabolique.

$$\begin{pmatrix} d_x \\ d_y \end{pmatrix} = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} + \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} \begin{pmatrix} x^2 \\ y^2 \end{pmatrix} + \begin{bmatrix} a_3 \\ a_4 \end{bmatrix} (xy) \tag{3.4}$$

Cependant, ce modèle n'est pas capable de décrire les surfaces très prononcées (par exemple celles d'un cube). De plus, la complexité de calcul de ce modèle augmente considérablement par rapport à celle de Tsai et Huang.

Par soucis de réduire la complexité de calcul, d'autres auteurs [LN91, San94] utilisent des modèles plus simples ayant de 2 à 6 paramètres. Ces modèles sont des versions plus simples du modèle quadratique (élimination des termes quadratiques, mixtes ou linéaires); dépendant du nombre de paramètres retenus, elles peuvent décrire les mouvements de translation, rotation, divergence (effet de zoom) et déformations linéaires.

Une autre forme de transformation est aussi proposée dans des travaux de Stiller [SS92] et de Sanson [San94]. Il s'agit de la transformation polynomiale (d'ordre aussi grand qu'on veut)

$$\begin{pmatrix} d_x \\ d_y \end{pmatrix} = \sum_{i=0}^{n} \sum_{j=0}^{i} \mathbf{a}_{ij} x^{i-j} y^j \tag{3.5}$$

où $\mathbf{a}_{ij} = [a_{ij}^x a_{ij}^y]^T$ sont les paramètres de mouvement, et n est l'ordre de la transformation. Ce modèle est en fait une généralisation des modèles paramétriques linéaires et quadratiques cités plus haut. Il permet d'appréhender des mouvements complexes si le nombre de paramètres est suffisamment grand. Cependant, la résolution des modèles comportant plus de 12 paramètres est extrêmement complexe, voire impossible à implanter en pratique à cause des difficultés de calculs. Néanmoins, il est intéressant de noter que souvent, seuls les paramètres d'ordre inférieur sont significatifs. Il serait possible alors d'utiliser de façon adaptative le nombre de paramètres, afin de limiter le débit de transmission tout en gardant le contrôle sur la qualité du

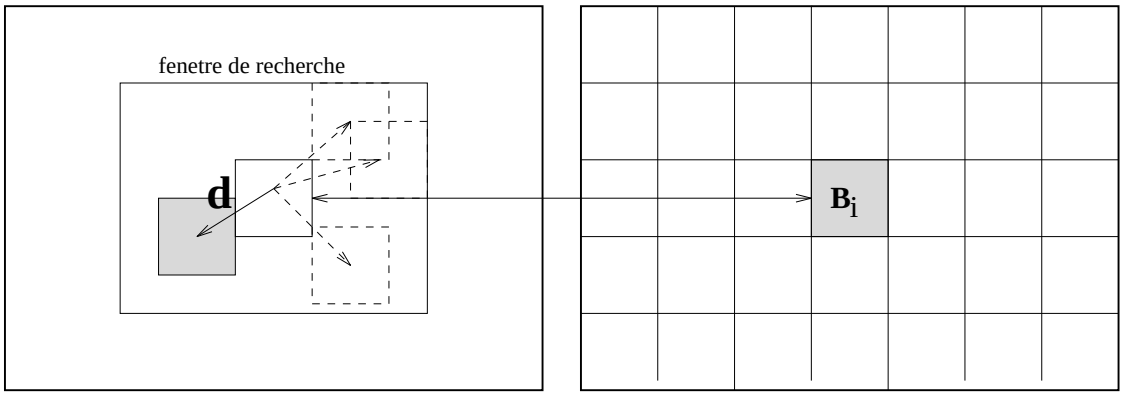


FIG. 3.1 - Illustration de la méthode d'appariement de blocs

mouvement estimé.

En général, les algorithmes utilisant ces modèles produisent de bons résultats, si le mouvement des objets est simple et si l'image est bien segmentée au sens du mouvement (le critère de segmentation doit inclure le modèle de mouvement utilisé). Or une telle segmentation ne peut être réalisée tant que l'information sur le mouvement n'est pas encore connue. En pratique, on utilise une segmentation arbitraire en blocs de forme et de taille régulière, en association avec un modèle de mouvement translationnel (2 paramètres); tous les pixels d'un bloc ont ainsi le même déplacement. Grâce à cette simplification, le problème d'estimation du mouvement revient à faire l'appariement des blocs entre deux trames successives (voir la figure 3.1). Le critère d'appariement est basé sur l'erreur quadratique entre un bloc et son correspondant. Afin de faciliter l'appariement, les standards H.261 et MPEG restreignent l'amplitude du déplacement à des multiples de la moitié de la distance entre deux pixels. De cette façon, l'espace d'état des solutions possibles est fortement réduit. Ce qui permet une recherche exhaustive dans une fenêtre délimitant le mouvement maximal permis.

Cette méthode, appelée communément *méthode d'appariement de blocs*, est très

simple à mettre en œuvre et ne nécessite pas la transmission de la carte de segmentation. C'est la raison pour laquelle elle a été choisie dans les standards de codage mentionnés ci-haut. Par contre, sur le plan de la performance, elle souffre de nombreux problèmes inhérents à la structure de bloc. En effet, une partition arbitraire de l'image en blocs ne permet pas de prendre en compte les discontinuités de mouvement aux frontières des objets. De plus, le modèle de mouvement translationnel est trop simple pour des mouvements complexes, ce qui fait que l'estimation du mouvement est souvent mauvaise. Il en résulte alors beaucoup d'erreurs dans la prédiction compensée par le mouvement. Pour que la qualité d'image soit acceptable, les algorithmes de codage utilisant cette méthode d'estimation doivent réserver une bonne part de la largeur de bande disponible pour la transmission de l'erreur résiduelle. Mais si le nombre de bits alloué à cette erreur est très restrictif (c'est le cas de codage à très bas débit), alors les artefacts de codage seront très visibles et deviendront inacceptables.

3.1.2.2 Méthodes "basées pixel"

Afin d'éviter les problèmes liés à une segmentation arbitraire en blocs, les méthodes de cette catégorie cherchent à estimer localement le mouvement de chaque pixel et produisent ainsi un champ de déplacement dense. Dans ces méthodes, les régions se réduisent à des pixels eux-mêmes et la notion de modèle de mouvement n'a plus d'utilité, car le déplacement le plus simple d'un point d'une image à l'autre est nécessairement translationnel (pour une estimation utilisant seulement deux trames). Pour régulariser le problème d'estimation du mouvement, il faut donc exploiter la similarité du déplacement d'un point avec celui de ses voisins (rappelons que le champ de déplacement est généralement lisse et comporte des discontinuités). Une des hypothèses les plus utilisées suppose que le champ de déplacement est globalement lisse. L'ajout de cette contrainte supplémentaire permet d'établir une

fonction (appelée *fonction d'énergie*) ayant une forme générale suivante:

$$U = U_i + \lambda U_d \tag{3.6}$$

où U_i représente le modèle structurel qui relie le champ de déplacement aux observations, et U_d est le terme mesurant le lissage du champ de déplacement. Dans [HS81], Horn et Schunck modélisent U_i par le carré de la contrainte spatio-temporelle

$$U_i(\mathbf{x}, u, v) = (g_x u + g_y v + g_t)^2 \tag{3.7}$$

et U_d par la somme des carrés des modules des gradients du champ de déplacement

$$U_d(\mathbf{x}, u, v) = (\frac{\partial u}{\partial x})^2 + (\frac{\partial u}{\partial y})^2 + (\frac{\partial v}{\partial x})^2 + (\frac{\partial v}{\partial y})^2 \tag{3.8}$$

Selon l'hypothèse d'invariance de l'intensité, l'énergie U_i doit être nulle. Mais en pratique, elle ne l'est pas à cause du bruit qui est omni-présent dans les signaux vidéo. Le problème d'estimation revient alors à minimiser la fonction d'énergie suivante:

$$\min_{\{u,v\}} \int_x \int_y \left((g_x u + g_y v + g_t)^2 + (\frac{\partial u}{\partial x})^2 + (\frac{\partial u}{\partial y})^2 + (\frac{\partial v}{\partial x})^2 + (\frac{\partial v}{\partial y})^2 \right) dxdy \tag{3.9}$$

Dans [KD88], un modèle équivalent à (3.9) est proposé sous une forme discrète utilisant la DPD qui est plus pratique pour le codage que la contrainte spatio-temporelle:

$$\min_{\mathbf{d}} \sum_{\mathbf{x}_i} \left((g(\mathbf{x}_i, t) - g(\mathbf{x}_i - \mathbf{d}(\mathbf{x}_i), t_-))^2 + \sum_{\mathbf{x}_j \in \eta(\mathbf{x}_i)} \|\mathbf{d}(\mathbf{x}_i) - \mathbf{d}(\mathbf{x}_j)\|^2 \right) \tag{3.10}$$

où $\eta(\mathbf{x}_i)$ est le voisinage de $\mathbf{x}_i$.

L'optimisation de la fonction d'énergie sous l'une ou l'autre forme est équivalente à un appariement de pixels entre deux trames successives et peut être interprétée comme suit: parmi plusieurs champs de déplacement minimisant l'erreur de compensation par le mouvement (U_i), on choisira celui dont les vecteurs voisins se ressemblent le plus (minimisant en même temps U_d).

Cette technique de régularisation donne des estimations valables à l'intérieur des régions ou objets, mais aux frontières de mouvement, les estimations sont généralement fausses à cause du sur-lissage imposé par la contrainte de lissage global, ou des occlusions où la contrainte spatio-temporelle n'est pas valide. Il en ressort donc que, pour améliorer la qualité de l'estimation, le modèle doit tenir compte des discontinuités et des occlusions.

Prendre en compte la discontinuité du mouvement

Étant donné les problèmes engendrés par un lissage global, Konrad et Dubois [KD89, KD92] relaxent cette contrainte en permettant au champ de déplacement d'avoir une certaine discontinuité locale. Cet algorithme modélise le champ de déplacement et les discontinuités de mouvement par des champs markoviens qui obéissent à la distribution de Gibbs. Ce procédé, utilisant le critère MAP (Maximum A Posteriori) dans le cadre de l'approche bayesienne, permet de modéliser a priori les caractéristiques des champs à estimer par une fonction d'énergie, qui favorise (ou défavorise) certaines configurations géométriques qu'on croit probable (ou peu probable). Cette fonction d'énergie est définie en pratique par une fonction de potentiel et des cliques dans un système de voisinage; plus concrètement, à chaque configuration géométrique (clique) des discontinuités, on associe une certaine valeur de potentiel (pénalité); l'énergie totale est la somme de potentiel de toutes les cliques. L'introduction du champ de discontinuité à étiquettes binaires permet de suspendre le lissage lorsqu'une discontinuité est détectée. L'estimation du mouvement requiert alors l'obtention simultanée de ces deux champs, ce qui rend la résolution du problème très difficile. On doit avoir recours à des techniques de résolution très complexes et très lentes, par exemple les techniques de recuit simulé [GG84], ou d'autres techniques plus rapides mais sous-optimales, telle la méthode ICM (Modes Conditionnels Itérés) [Bes86].

Bien que l'utilisation du champ de discontinuité permette une certaine améliora-

tion de la compensation par le mouvement, le champ obtenu ne reflète généralement pas les discontinuités réelles de la scène. En particulier, les sites actifs du champ de discontinuité ne forment pas de courbes fermées, alors que conceptuellement, elles devraient l'être. Dans [SH93], Stiller a proposé une méthode très semblable à celle de Konrad et Dubois, mais permettant d'éviter ce problème. Pour cela, il remplace le champ de discontinuité dont les sites sont définis à mi-chemin entre les pixels par un champ d'étiquettes de région (qui associe à chaque pixel de l'image une étiquette de région). Ce champ permet de décrire implicitement une discontinuité de mouvement par une différence des étiquettes de deux pixels adjacents. Le choix d'une fonction d'énergie favorisant le regroupement des pixels de mouvement semblable a permis d'obtenir des régions disjointes. Nous reviendrons à cette méthode avec plus de détails dans la section décrivant les méthodes de segmentation du mouvement.

Dans [Ana84], Anadan propose simplement une méthode de post-traitement afin d'améliorer la méthode de Horn et Schunck [HS81]. Cette technique consiste à détecter les discontinuités par l'association à chaque vecteur de déplacement une mesure de fiabilité basée sur l'erreur d'appariement (il est supposé qu'un lissage à travers une frontière de mouvement provoque une erreur élevée). Par la suite, les vecteurs seront lissés de façon inversement proportionnelle à leur fiabilité.

Traitement des occlusions

Une extension de [KD92] permettant de prendre en compte le traitement des occlusions est proposée dans [DD92]. L'idée est d'ajouter un champ d'occlusion qui entre en interaction à la fois avec le champ de discontinuité et le modèle structurel. Ce champ sert à annuler l'hypothèse d'invariance de l'intensité dans les zones occultées. Par une modélisation semblable au champ de discontinuité, une fonction de potentiel, un système de voisinage et ses cliques ont été définis heuristiquement, afin de favoriser la création des zones d'occlusion au voisinage des discontinuités de mouvement. Driessen et Biemond [DB91] ont obtenu dans un travail indépendant

un modèle très semblable à celui de [DD92]. Heitz et Bouthemy [HB90] modélisent aussi le champ de discontinuités par champ markovien binaire pour tenir compte des discontinuités de mouvement; mais une nouvelle technique est proposée pour le traitement des occlusions: ils utilisent le gradient spatio-temporel et les informations obtenues d'un détecteur de contours photométriques dans un test d'hypothèse afin de détecter les régions d'occlusion où la contrainte spatio-temporelle n'est pas valide. Le champ de déplacement y subira un lissage spécial à partir de l'information des voisins.

En général, les algorithmes qui considèrent le problème de discontinuité du mouvement et des occlusions produisent de meilleures compensations de mouvement par rapport à la méthode de Horn et Schunck. Mais l'inconvénient est que les calculs y sont très complexes. Par ailleurs, toutes ces méthodes produisent un vecteur de déplacement pour chaque pixel, auquel il faut ajouter éventuellement d'autres informations pour décrire les occlusions. Le mouvement représente donc un trop grand volume d'informations pour le codage à très bas débit.

3.2 Méthodes de segmentation basées sur le mouvement

Nous avons vu que les méthodes d'estimation du mouvement ne donnent pas de bons résultats à cause des problèmes de discontinuité et d'occlusion qui résultent d'une absence ou d'un choix inadéquat (au sens du mouvement) de la segmentation. De son côté, le problème de segmentation du mouvement à partir des variations spatio-temporelles de l'intensité dans une séquence d'images est ambigu, puisqu'on essaie de segmenter un champ de déplacement sans y avoir accès. Par ailleurs, la segmentation d'un champ de déplacement dense, produit par un algorithme d'estimation du mouvement indépendant [LN91], dépend surtout de la qualité du dépla-

cement estimé et ne garantit en aucun cas une reconstruction optimale. En fait, le problème de l'estimation et celui de la segmentation du mouvement sont intimement liés et doivent être effectués de façon interdépendante [MB87, Nic92]. Ce qui rend le problème très difficile à modéliser.

Il existe encore très peu de travaux traitant ce sujet. Les quelques articles publiés ces dernières années suivent principalement deux approches. La première propose des procédures plus ou moins heuristiques qui utilise le modèle paramétrique à la fois comme critère de segmentation et d'estimation du mouvement. La deuxième méthode se base sur le critère MAP dans le cadre de l'approche bayesienne pour modéliser explicitement l'interaction entre l'estimation et la segmentation du mouvement.

3.2.1 Estimation et segmentation basée sur le modèle paramétrique

Le principe de cette technique est de supposer que le mouvement de tous les objets constituant l'image peut être décrit par un modèle de mouvement paramétrique. Un objet est donc un regroupement des pixels adjacents vérifiant le modèle avec les mêmes paramètres. Cette hypothèse (n'est pas toujours vérifiée) sert en même temps de critère de segmentation et de contrainte pour l'estimation du mouvement des objets.

Dans [MHO89], Musmann et al. proposent une procédure itérative et hiérarchique utilisant le modèle à 8 paramètres dans l'estimation et segmentation du mouvement. Dans une première étape de la procédure (voir la figure 3.2), les objets initiaux sont obtenus à partir d'un détecteur de régions qui ont changé ou non dans les deux premières trames. Pour chacun de ces objets primitifs, les paramètres sont estimés. En se basant sur ces paramètres, une reconstruction des objets est effectuée. La deuxième étape se charge de la segmentation. En se basant sur la variance

de l'erreur de reconstruction, les régions qui ne sont pas correctement décrites par leurs paramètres seront détectées et considérées comme objets de second niveau. La procédure se répétera jusqu'à ce que la taille des régions qui ne respectent pas leur modèle soit plus petite qu'un seuil donné. Les régions dont les paramètres ne sont pas disponibles doivent être codées par des méthodes de codage intra-trame.

On remarque que cet algorithme nécessite une segmentation initiale et suppose qu'il existe un mouvement dominant dans un objet qui est composé possiblement de plusieurs régions de différents mouvements. Cette hypothèse n'est pas toujours vraie en pratique. Par ailleurs, une segmentation initiale, détectée à partir de la différence de deux trames successives, ne permet pas d'éviter le problème de discontinuité du mouvement, car ce type de segmentation contient inévitablement des discontinuités ou des occlusions (la figure 3.3 illustre ce problème). Cela aura certainement un impact négatif sur l'estimation des paramètres et par conséquent, sur le processus de division subséquent. Selon les auteurs, la procédure proposée donne de bons résultats au cas où le mouvement des objets est simple et il y a peu d'objets dans la scène; mais lorsque le mouvement est plus complexe, elle tend à diviser un vrai objet en plusieurs sous-objets, afin que le modèle soit respecté.

Pour remédier au problème de sur-segmentation, Diehl [Die91] remplace le modèle à 8 paramètres par un modèle quadratique à 12 paramètres, afin de pouvoir décrire les objets plus complexes. Il propose aussi d'utiliser l'information de contours et de texture, obtenue éventuellement d'un algorithme de codage intra-trame, pour renforcer la consistance de la segmentation. Sur ce point, Diehl ne fait pas de modélisation explicite pour incorporer cette information dans le critère de segmentation, mais l'utilise surtout pour fusionner les petites régions adjacentes et lever l'ambiguïté de segmentation lorsqu'un pixel, se trouvant sur un contour, appartient à plusieurs objets. La méthode de Diehl améliore celle de Musmann et al., mais elle augmente considérablement la complexité des calculs. Cette méthode souffre également du pro-

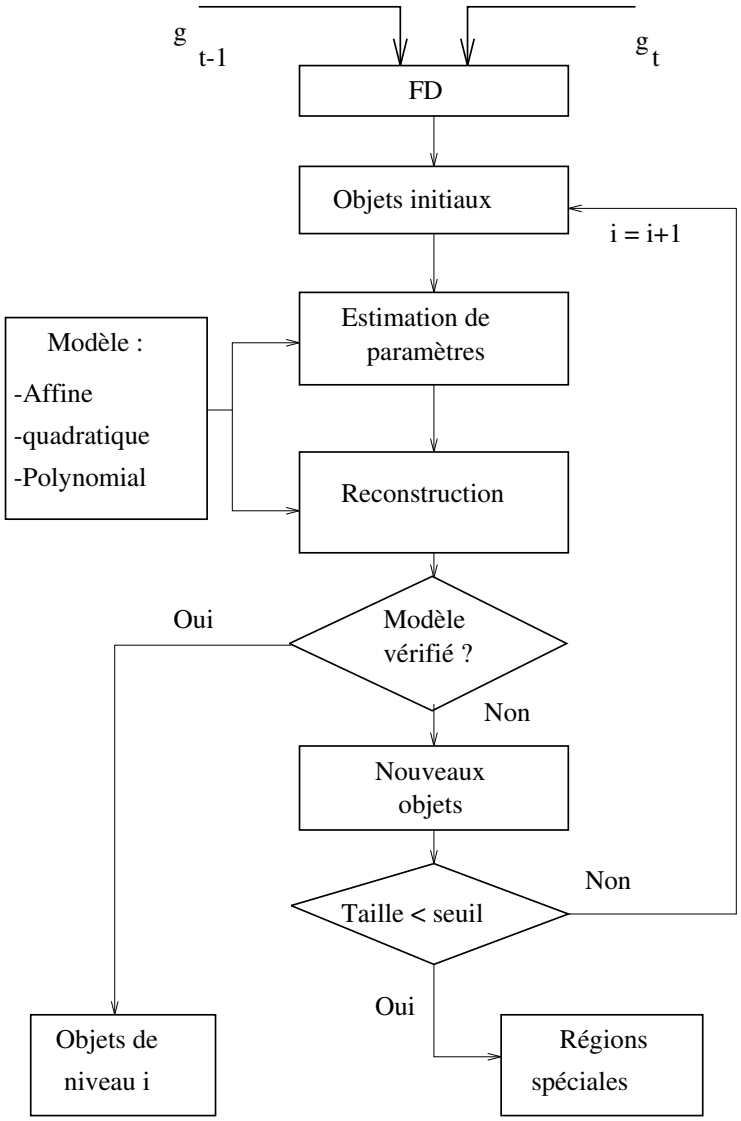


FIG. 3.2 - Algorithme de Musmann et al. pour l'estimation et la segmentation du mouvement

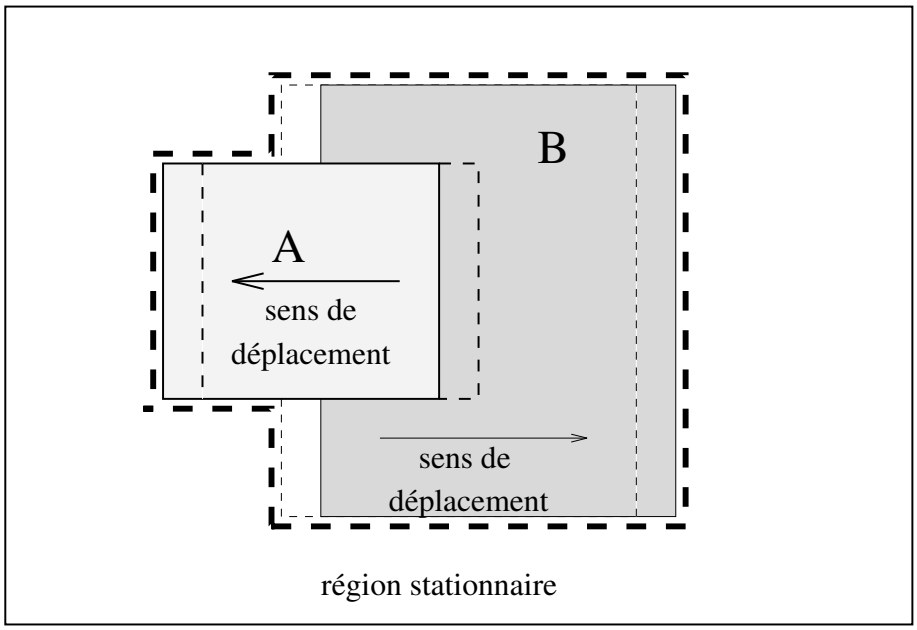


FIG. 3.3 - Exemple de discontinuité de mouvement d'une région obtenue par différence de trames. En pointillé gras : objet A au temps t_-; en pointillé fin: objet B au temps t_-; en continu gras: objet A au temps t; en continu fin : objet B au temps t; en pointillé très gras: région obtenu par différence de trames (dessinée légèrement plus grande pour fin d'illustration); cette dernière contient deux objets en mouvement dans deux directions opposées, plus une partie du fond qui est fixe

blème de discontinuité du mouvement, parce qu'elle utilise la même segmentation initiale que celle de la méthode de Musmann et al.

Dans [San94], Sanson utilise le modèle polynomial dans un algorithme itératif d'estimation et de segmentation basé sur une représentation de l'image en arbre quartenaire (quadtree). Le parcours de l'arbre s'effectue du niveau de plus basse résolution au niveau de plus haute résolution. À chaque niveau, les régions sont regroupées et sont constituées de blocs connectés ayant les mêmes paramètres de mouvement. La diminution de la taille des blocs lors du passage d'un niveau à l'autre permet d'affiner la forme des régions par un processus de division et de réallocation des blocs, selon un critère minimisant l'erreur de prédiction. De nouvelles régions seront créées si cette erreur dépasse un seuil donné. Cette méthode est en fait

une amélioration de la méthode d'appariement de blocs. Elle permet de créer des régions plus grandes et de réduire donc la redondance inter-bloc, mais il est évident que les problèmes inhérents à la structure de bloc ne sont pas résolus.

Un autre algorithme visant à améliorer la méthode d'appariement de blocs est proposé dans [DMMN94]. Ici, l'idée de base repose sur l'hypothèse qu'un bloc n'appartienne au plus qu'à deux objets de mouvement différent. Avec cette hypothèse, les blocs dont l'erreur de prédiction compensée par le mouvement est élevée seront divisés en deux parties par une ligne droite, dont les directions possibles ont été limitées et pré-définies dans un livre de codes; le mouvement des blocs voisins sera assigné à chacune des parties. Cette méthode permet de réduire l'erreur résiduelle, mais la segmentation reste quand même grossière.

3.2.2 Estimation et segmentation jointes du mouvement "basées pixel"

Dans le cadre de cette technique, on vise à modéliser explicitement l'interaction entre l'estimation et la segmentation du mouvement dans un modèle global.

Stiller et Hürtgen [SH93] proposent, dans le cadre d'une formulation bayesienne, d'associer à chaque site du champ de déplacement une étiquette de région (multi-états). L'ensemble des étiquettes forme un champ markovien. Ce champ contient implicitement l'information sur les discontinuités du mouvement et les occlusions (qui sont modélisées explicitement comme des champs markoviens par Konrad et Dubois [KD92]). En effet, une discontinuité de mouvement survient lorsque les étiquettes de deux pixels adjacents sont différentes, et ces dernières peuvent être ordonnées, afin d'inclure l'information sur l'ordre en profondeur des objets. La fonction d'énergie, obtenue via la distribution de Gibbs, est composée de trois termes. Le premier (modèle d'observations) modélise l'erreur de prédiction compensée par le mouvement de chaque région par un processus stationnaire gaussien généralisé. Le

deuxième terme impose que le champ de déplacement soit lisse dans chaque région (modèle de mouvement). Enfin, le dernier modélise a priori les contours de région qui sont supposés être spatialement lisses; en particulier un système de voisinage et des cliques ont été choisis, afin de favoriser le regroupement des pixels adjacents. Les deux derniers termes forment donc un critère de segmentation du mouvement, selon lequel, une région est formée par le regroupement des pixels adjacents ayant des déplacements similaires (lisse).

L'algorithme de Stiller et Hürtgen donne, selon les auteurs, une bonne estimation du champ de déplacement. Mais faute d'avoir une définition précise du critère de segmentation (le lissage n'est pas un critère assez fort), il divise l'image en un trop grand nombre de régions qui ne coïncident pas nécessairement avec les vrais objets de la scène. De plus, le mouvement est représenté sous la forme d'un champ de déplacement dense et d'un ensemble d'étiquettes de région, qui doivent être transformées sous une forme plus compacte, afin de réduire le débit de transmission. Dans [Sti94], l'auteur propose d'approximer a posteriori le déplacement des pixels dans une région par un modèle de mouvement de type polynomial. L'inconvénient de cette approximation est qu'elle pourrait entraîner des distorsions importantes de l'image reconstruite.

En considérant le fait que la contrainte de lissage n'est pas un bon critère de segmentation, alors que le modèle paramétrique semble plus adéquat pour la segmentation, mais ne permet de décrire que des mouvements assez simples, Chang et al.[CST94] combinent ces deux modèles de mouvement dans un modèle global, dans lequel la contrainte de lissage est utilisée pour estimer le mouvement, tandis que le modèle paramétrique sert de critère de segmentation. En utilisant une formulation bayesienne, les auteurs proposent deux modèles, l'un pour l'estimation et l'autre pour la segmentation. L'interaction entre les deux modèles est exprimée par un terme, qui pénalise la différence entre les deux types de déplacement $\mathbf{d}$ et $\mathbf{d}'$

d'un point ($\mathbf{d}$ représente le déplacement du champ lisse, $\mathbf{d}'$ est issu du modèle de mouvement et est fonction de la position du point et des paramètres ϕ).

Plus précisément, à partir d'une carte de segmentation initiale et ses paramètres de mouvement [1], le champ de déplacement dense ($\mathbf{d}$), les paramètres de mouvement (ϕ) et la segmentation seront raffinés par une procédure itérative. Chaque itération comporte deux étapes. La première étape remet à jour le champ de déplacement dense ($\mathbf{d}$) en minimisant une fonction composée de trois termes: le premier terme mesure l'erreur de compensation de mouvement (en utilisant le champ lisse $\mathbf{d}$); le deuxième terme est la contrainte de lissage (pour le champ $\mathbf{d}$) à l'intérieur de chaque région; le dernier terme impose que les deux types de déplacement $\mathbf{d}$ et $\mathbf{d}'$ d'un pixel soient similaires. La deuxième étape effectue simultanément la ré-estimation des paramètres de mouvement et la ré-affectation des étiquettes de région. Pour cela, les auteurs proposent de minimiser une fonction d'énergie qui mesure l'erreur de compensation de mouvement (utilisant le déplacement $\mathbf{d}'$), la différence entre $\mathbf{d}$ et $\mathbf{d}'$ et la complexité des contours. On retrouve dans cette étape une similitude au modèle de Stiller; la différence majeure réside dans le remplacement du modèle de mouvement lisse par un modèle de mouvement paramétrique.

Malheusement, bien que l'idée sous-jacente de cette approche semble intéressante, le modèle qui en résulte est très complexe et très difficile à résoudre. Malgré tout, les auteurs rapportent que leur algorithme produit de bonnes estimations et segmentations du mouvement. En particulier, son avantage par rapport à l'algorithme de Stiller est que les sorties sont sous plusieurs formes: champ de déplacement dense ou régions avec leurs paramètres de mouvement; évidemment, cette dernière forme est préférable, puisqu'elle se prête mieux au codage. Toutefois, tout comme la méthode de Stiller, cette méthode divise l'image en un nombre élevé de régions, qui ne coïn-

1. La carte de segmentation initiale est obtenue par regroupement grossier d'un champ de déplacement dense, qui résulte d'une estimation "basée pixel" utilisant uniquement la contrainte de lissage comme modèle de mouvement. Les paramètres initiaux viennent d'une estimation "basée région" utilisant la segmentation initiale.

cident pas nécessairement avec les objets de la scène. Il s'agit là d'un inconvénient majeur pour le codage à très bas débit.

Chapitre 4

Estimation et segmentation du mouvement utilisant la segmentation d'intensité

D'après le chapitre précédent, les méthodes d'estimation et de segmentation du mouvement proposées ont des difficultés à cause du problème de discontinuité du mouvement, ou sont biaisées par le choix d'une segmentation initiale. De plus, les contours de mouvement obtenus ne sont pas nécessairement ceux des objets réels de la scène. Or c'est précisément l'une des caractéristiques les plus intéressantes et les plus recherchées dans le codage à très bas débit, puisque cela permet de dissimuler les artefacts de codage.

Dans ce chapitre, nous proposons un nouvel algorithme d'estimation et de segmentation du mouvement qui concentre les erreurs de codage au voisinage des contours d'intensité, afin de bénéficier de l'effet de masque du système de visions humain. Cet algorithme est basé sur le fait que les contours d'intensité coïncident généralement avec les contours de mouvement, d'où, une segmentation basée sur l'intensité peut être utilisée comme une segmentation initiale au sens du mouve-

ment. En nous basant sur ces observations, nous proposons de résoudre le problème d'estimation et de segmentation du mouvement en trois étapes:

1. Estimer le mouvement des régions d'intensité.
2. Fusionner des régions adjacentes ayant des mouvements semblables.
3. Ajuster les contours d'intensité, afin qu'ils soient consistants au sens du mouvement.

La description détaillée de ces trois étapes sera présentée dans les trois avant-dernières sections de ce chapitre. Auparavant, nous consacrerons la première et la deuxième section à la description de la méthode de segmentation spatiale et des hypothèses fondamentales sur lesquelles s'appuient les trois étapes mentionnées ci-dessus. Enfin dans la dernière section, nous discuterons de la complexité de cet algorithme.

4.1 Segmentation spatiale basée sur l'intensité

La segmentation spatiale est un domaine de recherche à part entière en traitement d'images. Cependant, elle ne constitue pas l'objet de notre recherche, puisque nous utilisons ce type de segmentation simplement pour initialiser notre algorithme d'estimation et de segmentation du mouvement. Nous consacrons néanmoins cette section à la description de la méthode qui nous a permis d' obtenir les segmentations spatiales nécessaires à notre algorithme. Il s'agit d'une technique de segmentation utilisant l'algorithme développé dans [Lec89] et qui calcule une estimation de la longueur de description minimale (MDL), conditionnée sur l'image originale. La description qui suit montre le principe de base de cette méthode.

Soit D^n la description de l'image correspondant à la n-ième image I^n de la séquence, et L la fonction de longueur de code idéale. La description optimale D^n

conditionnée sur I^n minimise:

$$L(D^n \mid I^n) \quad = \quad L(I^n \mid D^n) + L(D^n), \tag{4.1}$$

Dans le cas de segmentation, la description D^n est modélisée en une fonction constante par parties, et l'image I^n comme la somme $I^n = D^n + \eta$, où η est un bruit blanc stationnaire gaussien de moyenne nulle et de variance σ^2. Le premier terme de droite dans l'équation MDL définit le codage d'entropie de la variable aléatoire $\eta = I^n - D^n$, tandis que le second terme est la mesure de la complexité associée à la description D^n. Cette mesure de complexité est choisie comme étant proportionnelle au nombre de points frontières dans D^n. De ce fait, l'équation peut être reformulée par:

$$\begin{aligned} L(D^n \mid I^n) \quad &= \quad \frac{1}{2}\sum_i \frac{(D_i^n - I_i^n)^2}{2\sigma^2} + \frac{1}{2} log(2\pi\sigma^2) \\ &+ \quad \frac{1}{2}\sum_i \sum_{j \in N(i)} [1 - \delta(D_i^n - D_j^n)] \end{aligned}$$

expression dans laquelle i et j sont les indices des pixels de l'image, $N(i)$ un voisinage du pixel i et $\delta(.)$ symbolise la fonction de Kronecker. On retrouve dans cette équation une similitude aux formulations bayesiennes d'un problème de segmentation. Par analogie, le calcul de l'estimation MDL s'effectue par procédure itérative de type recuit déterministe ("deterministic annealing"). Malheureusement, la technique de recuit déterministe ne peut converger que très lentement vers l'estimation. Afin d'apporter une compensation à cet inconvénient, le processus est interrompu après un certain nombre d'itérations (quand l'estimation est assez proche de la valeur optimale), et est suivi par une procédure de regroupement ("clustering") basée-luminance. La dernière étape consiste à étiqueter l'image pour l'obtention de la segmentation désirée. Un exemple est donné à la figure 4.1.

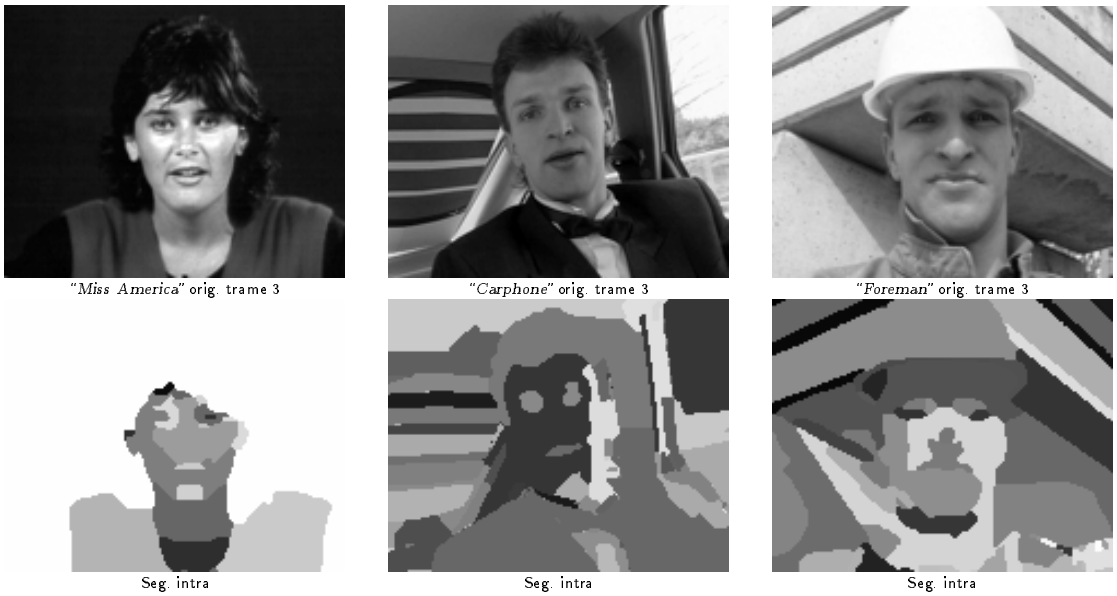


FIG. 4.1 - Exemples de segmentation basée sur l'intensité par l'algorithme MDL

4.2 Modèle de région et de mouvement

Soit Ψ une partition de l'image en N régions connexes. Ψ peut être obtenue d'une segmentation arbitraire en blocs ou basée sur l'observation de l'intensité de l'image. Dans les algorithmes de codage standards H.261 ou MPEG, Ψ est l'ensemble des blocs constituant l'image, tandis que dans les méthodes de codage utilisant le champ de déplacement dense, Ψ est l'ensemble de tous les pixels. Dans ce projet, nous choisissons Ψ comme étant la carte de segmentation basée sur l'intensité, issue de l'algorithme MDL que nous avons décrit à la section précédente. Ce choix est motivé par une observation importante: *dans une séquence d'images, les contours de mouvement coïncident presque toujours avec les contours d'intensité* (l'inverse n'est pas vrai) [HKLM88, KD89]. Sous cette hypothèse, les régions d'intensité sont des sous-objets des objets réels en mouvement. Autrement dit, chaque région homogène spatialement possède aussi un mouvement homogène. Évidemment, cette hypothèse n'est pas toujours vérifiée. Les exceptions correspondent par exemple au cas où deux

objets ayant la même intensité se déplacent dans différentes directions. Nous considérons que ces exceptions sont plutôt rares et ne les prenons pas en considération. Par conséquent, il est à prévoir que l'estimation du mouvement de ces régions sera généralement fausse. Toutefois, nous nous attendons à ce que l'erreur de prédiction y soit faible et peu visible du fait qu'il y a très peu de variation de l'intensité dans ces régions (sinon la méthode MDL devrait être capable de les segmenter correctement). Il existe également d'autres causes, tels l'erreur de quantification et les ombrages d'un objet sur ses voisins, qui peuvent aussi provoquer un certain écart entre les deux types de contour. Toutefois, les observations montrent que, dans ces cas, les décalages sont en général très faibles (de l'ordre de quelques pixels seulement) [Nic92]; le rapport entre le nombre de pixels en erreur et la taille d'une région serait donc minime. Par conséquent, ces petits écarts ne sont pas très critiques pour une estimation globale du mouvement d'une région. Étant donné toutes ces considérations, il serait plausible de penser que l'hypothèse de mouvement dominant dans une région s'appliquera mieux dans notre cas que dans celui où les objets sont obtenus par la différence de trames.

Nous avons eu l'occasion de mentionner dans le chapitre 2 que, dans un algorithme de codage à très bas débit, il est impératif d'avoir une description compacte du mouvement d'une région pour minimiser la quantité d'information à transmettre. Dans cette étude, nous adoptons la méthode de représentation du mouvement d'une région par modèle affine proposé dans [Nic92]. Ce modèle assume que le déplacement de chaque point $\mathbf{x} = [x, y]^T$ d'une région est décrit par la transformation

$$\begin{bmatrix} d_x \\ d_y \end{bmatrix} = h(\mathbf{x}, \phi) = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} . \begin{bmatrix} x - x_g \\ y - y_g \end{bmatrix} \tag{4.2}$$

où $\mathbf{d} = [d_x, d_y]^T$ est le déplacement du point $\mathbf{x}$; (x_g, y_g) sont les coordonnées du centre de gravité de la région; $\phi = [a_1, a_2, b_{11}, b_{12}, b_{21}, b_{22}]^T$ sont des paramètres de mouvement, capable de décrire les mouvements en translation, rotation, divergence

et déformation linéaire d'un objet plan [Nic92]. Donc, le mouvement de chacune des régions $\psi_n \in \Psi \quad (n = 1, \ldots, N)$ est déterminé par le vecteur de paramètres ϕ_n. Par conséquent, l'estimation du mouvement revient à estimer les paramètres ϕ_n de ces régions.

4.3 Estimation du mouvement des régions d'intensité

4.3.1 Formulation du problème d'estimation du mouvement "basé région"

Le codage par compensation de mouvement nécessite d'estimer des descripteurs de mouvement minimisant l'erreur de reconstruction, c'est à dire permettant d'obtenir la "meilleure" image reconstruite possible. Le problème est qu'il n'existe pas de critère parfait permettant de mesurer la qualité d'une image reconstruite. En effet, la perception visuelle d'un défaut dépend non seulement de l'amplitude de l'erreur, mais également de sa position dans l'image, car on ne percevra pas une erreur de la même façon, selon qu'elle se situe dans une zone en mouvement ou dans une zone fixe; il en est de même pour les cas où elle se trouve dans une région très texturée ou presque uniforme. Il est clair alors qu'on ne peut mesurer la qualité d'une image en considérant uniquement des critères mathématiques; c'est plutôt sur la qualité visuelle que les images doivent être jugées. Néanmoins, d'un point de vue algorithmique, il est nécessaire de choisir un critère mathématique. Dans cette étude, nous choisissons l'erreur quadratique entre l'image compensée par le mouvement et l'originale comme étant la mesure de qualité. Nous pouvons dès lors formuler le problème d'estimation du mouvement.

Connaissant une partition Ψ de l'image au temps t, nous pouvons estimer les

paramètres de mouvement $\phi_1, \ldots, \phi_n$ qui définissent la transformation de chacune des régions $\psi_n \in \Psi$ $(n = 1, \ldots, N)$ dans la trame au temps t_-. En effet, soit $\Phi = [\phi_1^T, \ldots, \phi_n^T]^T$ l'ensemble de paramètres de toutes les régions de la partition Ψ au temps t, c'est à dire que ϕ_n décrit le mouvement de la région ψ_n, alors sous l'hypothèse d'invariance de l'intensité, Φ peut être estimé par minimisation d'une fonction d'énergie mesurant l'erreur quadratique de prédiction compensée par le mouvement

$$\hat{\Phi} = \arg\min_{\{\phi_n\}} \sum_{n=1}^{N} \sum_{\mathbf{x} \in \psi_n} [g(\mathbf{x}, t) - \tilde{g}(\mathbf{x} - h(\mathbf{x}, \phi_n), t_-)]^2 \tag{4.3}$$

où $\tilde{g}(\mathbf{x} - h(\mathbf{x}, \phi_n), t_-)$ est la prédiction compensée par le mouvement du pixel $\mathbf{x}$ entre les instants t et t_- et interpolée spatialement $(\tilde{g})$ par un opérateur bicubique [Key81] (nous décrirons ce type d'interpolateur ci-après). Cette formulation du problème d'estimation du mouvement est similaire à celles proposées dans [Die91, Nic92]. La différence majeure est qu'ici, nous utilisons une segmentation spatiale, tandis que dans les autres méthodes, il s'agit de la différence de trames [Die91] ou des blocs d'un arbre quartenaire [Nic92]. De plus, comme nous le verrons un peu plus loin, la méthode de résolution que nous utiliserons est aussi différente de celles proposées dans ces deux articles.

4.3.1.1 Interpolation spatiale

Généralement, le déplacement d'un point sera représenté par un vecteur dont les composantes sont réelles, car le déplacement n'est pas nécessairement d'un nombre entier de pixels. Il est donc nécessaire, afin d'évaluer la prédiction compensée par le mouvement d'un point, de connaître l'intensité de l'image en des points qui ne font pas partie de la grille d'échantillonnage. Cette opération est en fait une reconstruction d'un signal échantillonné et requiert un filtrage passe-bas pour éliminer les répliques du spectre de base.

Par ailleurs, comme nous le verrons dans la section suivante, la méthode de

minimisation de la fonction d'énergie exige le calcul du gradient de la fonction d'intensité, ce qui implique que la fonction interpolée doit avoir certaine propriété de continuité. Pour cela, nous choisissons la méthode d'interpolation bicubique présentée dans [Key81]. Cet interpolateur modélise localement la fonction à interpoler par un polynôme de troisième degré, de sorte que la fonction interpolée et sa dérivée soient continues. Mathématiquement, cette interpolation est exprimée comme la convolution avec un filtre séparable dont la réponse impulsionnelle est composée de polynômes cubiques par morceaux:

$$h(s) = \begin{cases} \frac{3}{2}|s|^3 - \frac{5}{2}|s|^2 + 1 & 0 < |s| < 1 \\ -\frac{1}{2}|s|^3 + \frac{5}{2}|s|^2 - 4|s| + 2 & 1 < |s| < 2 \\ 0 & 2 < |s| \end{cases}$$

La réponse impulsionnelle uni-dimensionnelle de ce filtre est présentée à la figure 4.2

Pour reconstruire l'image (2D), on applique cet interpolateur d'abord horizontalement, puis verticalement.

Notons que cette méthode d'interpolation permet d'obtenir facilement le gradient spatial. En effet, puisque l'interpolation de l'intensité est donnée par un filtrage séparable dont la convolution uni-dimensionnelle est

$$W(y) = \sum_x h(y - x)V(x) \tag{4.4}$$

où y est la variable continue et x est la variable discrète. Sachant que le système est linéaire, on peut alors écrire

$$\frac{\partial W(y)}{\partial y} = \sum_x \frac{\partial h(y - x)}{\partial y} V(x) \tag{4.5}$$

d'où le gradient spatial peut être calculé par la convolution de la fonction d'intensité avec un filtre séparable dont la réponse impulsionnelle est la dérivée de $h(s)$. La figure 4.3 montre la réponse impulsionnelle de ce filtre.

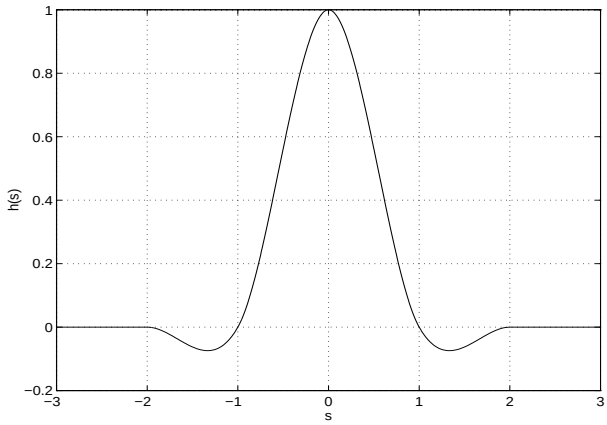


FIG. 4.2 - Réponse impulsionnelle du filtre d'interpolation cubique de l'intensité

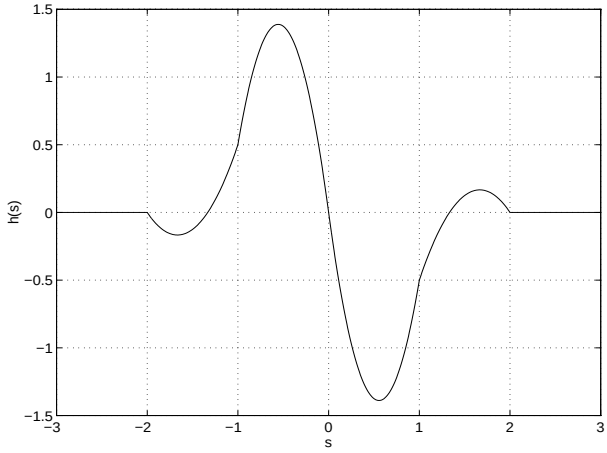


FIG. 4.3 - Réponse impulsionnelle du filtre dans le calcul du gradient spatial

4.3.2 Méthode de résolution

Le problème d'optimisation défini par (4.3) est séparable par région. Autrement dit, les paramètres de mouvement de chaque région de l'image peuvent être estimés indépendamment de ceux des autres régions par la minimisation suivante:

$$\widehat{\phi_n} = \arg\min_{\{\phi_n\}} E(\phi) = \sum_{\mathbf{x}\in\psi_n} [g(\mathbf{x},t) - \tilde{g}(\mathbf{x} - h(\mathbf{x},\phi), t_-)]^2 \qquad n = 1,\ldots,N. \tag{4.6}$$

Il s'agit d'un problème de type moindre-carré classique, qui peut être résolu par des méthodes itératives.

En général, ces méthodes de résolution s'appuient sur une approximation de la fonction d'énergie par expansion en série Taylor au voisinage de la solution courante. Souvent, on limite l'expansion au terme linéaire ou quadratique. Dans [Nic92], l'auteur propose l'utilisation de la méthode de descente du gradient (approximation linéaire), alors que Diehl [Die91] propose de combiner la méthode Newton modifiée et la méthode quasi-Newton (approximation quadratique) dans une méthode plus complexe mais plus adéquate pour son modèle de mouvement quadratique (à 12 paramètres). Dans notre cas, nous préférons la méthode Gauss-Newton [GMW81] qui est plus performante que la méthode du gradient et qui est moins complexe que la méthode de Diehl. Toutefois, il est bien connu que toutes ces méthodes ne permettent pas d'obtenir l'optimum global. En pratique, on doit se contenter d'un minima local, ce qui est souvent le cas en estimation du mouvement. Notons que la qualité de l'optimum local dépend et du choix d'une solution initiale et du degré de non-linéarité de la fonction à optimiser.

Selon la méthode de Gauss-Newton, la valeur des paramètres à l'itération courante k est donnée par l'équation itérative

$$\phi_n^k = \phi_n^{k-1} - \epsilon H^{-1} \nabla_\phi(E(\phi_n)), \tag{4.7}$$

où ϵ est le gain, H définit le hessien et $\nabla_\phi(E(\phi_n))$ est le gradient de la fonction d'énergie dans le domaine des paramètres.

Nous détaillons dans ce qui suit les calculs analytiques pour la détermination du terme de mise à jours (deuxième terme de l'équation (4.7)).

4.3.2.1 Calcul du gradient

Puisque les calculs sont les mêmes pour toutes les régions, les indices de région deviennent superflus et alourdissent inutilement les expressions mathématiques. Afin d'alléger la notation dans les calculs qui suivent, nous les supprimons et remplaçons l'indice de région n dans ϕ_n par $i = (1, \ldots, 6)$ qui est l'indice des composantes du vecteur des paramètres. Nous notons donc

$$\phi = [\phi_1, \ldots, \phi_6]^T = [a_1, a_2, b_{11}, b_{12}, b_{21}, b_{22}]^T.$$

ϕ_i indique alors le i^{ieme} paramètre de la région courante ψ.

Avec cette nouvelle notation, le gradient peut être calculé de la manière suivante:

$$\nabla_\phi E(\phi) = \sum_{\mathbf{x}\in\psi} \begin{pmatrix} \frac{\partial}{\partial\phi_1} DPD^2(\mathbf{x}, h(\mathbf{x},\phi)) \\ \vdots \\ \frac{\partial}{\partial\phi_i} DPD^2(\mathbf{x}, h(\mathbf{x},\phi)) \\ \vdots \\ \frac{\partial}{\partial\phi_6} DPD^2(\mathbf{x}, h(\mathbf{x},\phi)) \end{pmatrix} \tag{4.8}$$

Avec

$$\frac{\partial}{\partial\phi_i} DPD^2(\mathbf{x}, h(\mathbf{x},\phi)) = 2\, DPD \frac{\partial}{\partial\phi_i} DPD(\mathbf{x}, h(\mathbf{x},\phi)) \tag{4.9}$$

Par définition de la DPD on a

$$\frac{\partial}{\partial\phi_i} DPD(\mathbf{x}, h(\mathbf{x},\phi)) = -\frac{\partial}{\partial\phi_i} \tilde{g}(\mathbf{x} - h(\mathbf{x},\phi), t_-) \tag{4.10}$$

En remplaçant $h(\mathbf{x},\phi)$ par $\mathbf{d} = [d_x d_y]^T$ et en appliquant la formule de dérivation en chaîne on obtient

$$\begin{aligned} \frac{\partial}{\partial\phi_i} \tilde{g}(\mathbf{x} - h(\mathbf{x},\phi), t_-) \;=\; & -[\frac{\partial}{\partial u} \tilde{g}(u, v, t_-)|_{u=x-d_x, v=y-d_y}] \frac{\partial}{\partial\phi_i}(x - d_x) \\ & -[\frac{\partial}{\partial v} \tilde{g}(u, v, t_-)|_{u=x-d_x, v=y-d_y}] \frac{\partial}{\partial\phi_i}(y - d_y) \end{aligned} \tag{4.11}$$

Rappelons que pour le modèle affine

$$d_x = a_1 + b_{11}(x - x_g) + b_{12}(y - y_g) = \phi_1 + \phi_3(x - x_g) + \phi_4(y - y_g)$$

$$d_y = a_2 + b_{21}(x - x_g) + b_{22}(y - y_g) = \phi_2 + \phi_5(x - x_g) + \phi_6(y - y_g)$$

et, sachant que x et y sont indépendants de ϕ, ce qui donne:

$$\begin{pmatrix} \frac{\partial}{\partial \phi_1}\tilde{g} \\ \frac{\partial}{\partial \phi_2}\tilde{g} \\ \frac{\partial}{\partial \phi_3}\tilde{g} \\ \frac{\partial}{\partial \phi_4}\tilde{g} \\ \frac{\partial}{\partial \phi_5}\tilde{g} \\ \frac{\partial}{\partial \phi_6}\tilde{g} \end{pmatrix} = \begin{pmatrix} \frac{\partial}{\partial a_1}\tilde{g} \\ \frac{\partial}{\partial a_2}\tilde{g} \\ \frac{\partial}{\partial b_{11}}\tilde{g} \\ \frac{\partial}{\partial b_{12}}\tilde{g} \\ \frac{\partial}{\partial b_{21}}\tilde{g} \\ \frac{\partial}{\partial b_{22}}\tilde{g} \end{pmatrix} = - \begin{pmatrix} \nabla_x \tilde{g}(\mathbf{x} - \mathbf{d}, t_-) \\ \nabla_y \tilde{g}(\mathbf{x} - \mathbf{d}, t_-) \\ (x - x_g)\nabla_x \tilde{g}(\mathbf{x} - \mathbf{d}, t_-) \\ (y - y_g)\nabla_y \tilde{g}(\mathbf{x} - \mathbf{d}, t_-) \\ (x - x_g)\nabla_y \tilde{g}(\mathbf{x} - \mathbf{d}, t_-) \\ (y - x_y)\nabla_x \tilde{g}(\mathbf{x} - \mathbf{d}, t_-) \end{pmatrix} \tag{4.12}$$

où :

$$\nabla_x(\tilde{g}(\mathbf{x} - \mathbf{d}, t_-)) = \frac{\partial}{\partial u}\tilde{g}(u, v, t_-)|_{u=x-d_x, v=y-d_y}$$

et

$$\nabla_y(\tilde{g}(\mathbf{x} - \mathbf{d}, t_-)) = \frac{\partial}{\partial v}\tilde{g}(u, v, t_-)|_{u=x-d_x, v=y-d_y}$$

Finalement, par substitution on obtient l'expression définissant le gradient de la fonction d'énergie

$$\nabla_\phi E(\phi) = 2 \sum_{\mathbf{x} \in \psi} [g(\mathbf{x}, t) - \tilde{g}(\mathbf{x} - h(\mathbf{x}, \phi_n), t_-)] \begin{pmatrix} \nabla_x \tilde{g}(\mathbf{x} - \mathbf{d}, t_-) \\ \nabla_y \tilde{g}(\mathbf{x} - \mathbf{d}, t_-) \\ (x - x_g)\nabla_x \tilde{g}(\mathbf{x} - \mathbf{d}, t_-) \\ (y - y_g)\nabla_y \tilde{g}(\mathbf{x} - \mathbf{d}, t_-) \\ (x - x_g)\nabla_y \tilde{g}(\mathbf{x} - \mathbf{d}, t_-) \\ (y - x_y)\nabla_x \tilde{g}(\mathbf{x} - \mathbf{d}, t_-) \end{pmatrix} \tag{4.13}$$

4.3.2.2 Calcul du hessien

Soit $J(\phi)$ le jacobien de la fonction $DPD(\mathbf{x}, \phi)$, alors selon [GMW81], H est donné par

$$H = J(\phi)^T J(\phi) + Q(\phi) \tag{4.14}$$

où $Q(\phi)$ est le terme des dérivées secondes. Le calcul de $Q(\phi)$ est coûteux en temps de calcul. Théoriquement, dans la mesure où l'erreur résiduelle de la DPD est faible, $Q(\phi)$ tend vers zéro. La méthode Gauss-Newton suppose que cette condition est vraie et néglige $Q(\phi)$ dans le calcul du hessien. Donc, la détermination du hessien nécessite seulement les dérivées de premier ordre.

En ce qui concerne le jacobien, on peut le calculer de la manière suivante: si la région contient m pixels notés $(\mathbf{x}_1, \ldots, \mathbf{x}_m)$, alors $J(\phi)$ est défini par une matrice de dimension $m \times 6$

$$J(\phi) = \begin{bmatrix} \frac{\partial}{\partial \phi_1} DPD(\mathbf{x}_1, h(\mathbf{x}_1, \phi)) & \cdots & \frac{\partial}{\partial \phi_6} DPD(\mathbf{x}_1, h(\mathbf{x}_1, \phi)) \\ \vdots & \vdots & \vdots \\ \frac{\partial}{\partial \phi_1} DPD(\mathbf{x}_m, h(\mathbf{x}_m, \phi)) & \cdots & \frac{\partial}{\partial \phi_6} DPD(\mathbf{x}_m, h(\mathbf{x}_m, \phi)) \end{bmatrix} \tag{4.15}$$

Il s'en suit donc que le hessien est une matrice 6×6 dont chaque élément $H_{p,q}$ est défini par

$$H_{p,q} = \sum_{k=1}^{m} \frac{\partial}{\partial \phi_p} DPD(\mathbf{x}_k, h(\mathbf{x}_k, \phi)) \cdot \frac{\partial}{\partial \phi_q} DPD(\mathbf{x}_k, h(\mathbf{x}_k, \phi)). \tag{4.16}$$

Dans cette expression, les dérivées partielles dans la direction des ϕ_i sont les mêmes que celles impliquées dans le calcul du gradient (voir les équations (4.10) et (4.12)). Donc, la détermination du hessien ne nécessite que très peu de temps de calcul supplémentaire.

4.3.3 Mise en œuvre

La mise en œuvre de la méthode de résolution présentée à la section précédente a été réalisée selon l'algorithme suivant:

- Pour chaque région ψ_n de l'image faire
  - Initialisation des paramètres de mouvement $\phi = \phi^0$

- $k = 1$
  - Calculer ϕ^k selon l'équation (4.7)
  - $E^k - E^{k-1} < 0$? (test de convergence)
  - si non (diverge)
    - recalculer ϕ^k en diminuant ϵ par un facteur de 2
    - répéter 3 fois ou jusqu'à ce que $E^k - E^{k-1} < 0$
    - si $E^k - E^{k-1} > 0$? (diverge définitivement)
    - $\phi^k = \phi^{k-1}$
    - fin
  - si oui (converge)
    - arrêter?
    - si non $k = k + 1$
    - si oui fin

- fin

Comme nous avons vu de ce qui précède, le fonctionnement de l'algorithme nécessite la définition d'un critère d'arrêt et le choix d'une solution initiale. Nous détaillons tous ces points dans ce qui suit.

4.3.3.1 Critère d'arrêt

Pour définir le critère d'arrêt d'une équation itérative, il est possible de définir les critères suivants:

1. un seuil de l'erreur d'appariement de la région en-dessous duquel l'estimation des paramètres est considérée comme acceptable;
2. un seuil de variation relative (décroissance) de la fonction d'énergie entre deux ou plusieurs itérations consécutives;

3. un nombre maximal d'itérations.

Sachant que l'erreur d'appariement peut être due uniquement à une mauvaise segmentation (cas où l'hypothèse de mouvement dominant s'applique), une erreur élevée ne signifie donc pas nécessairement que l'estimation est mauvaise. Pour cette raison, nous écartons le premier et ne retenons que les deux derniers critères pour notre algorithme d'optimisation. Plus concrètement, ces deux critères ont été établis comme suit:

- la décroissance relative de la fonction d'énergie, c.à.d.

$$\frac{E_i - E_{i-1}}{E_{i-1}},$$

  doit être plus petite que -0.001% pendant 3 itérations consécutives;

- le nombre maximal d'itération égale à 30.

4.3.3.2 Choix d'une solution initiale

En général, la méthode de Gauss-Newton ne permet d'obtenir qu'un minimum local. La qualité de ce minimum ainsi que la convergence dépend du choix d'une solution initiale. Théoriquement, il est toujours souhaitable de choisir une solution qui soit la plus près possible de l'optimal. En estimation du mouvement, cette initialisation peut être obtenue en utilisant des paramètres estimés de la trame précédente. Dans le cas de l'estimation "basée région", un tel scénario est envisageable si le nombre de régions reste constant, ou si on dispose d'un lien temporel permettant d'identifier la correspondance d'une région dans l'image précédente. Malheureusement, dans notre cas, cette stratégie n'est pas applicable, puisque la segmentation spatiale a été effectuée par un module indépendant qui ne prend pas l'évolution temporelle des régions en considération. Par conséquent, nous choisissons de mettre systématiquement à zéro la valeur initiale des paramètres de mouvement. Nous nous

attendons donc à des problèmes de convergence lorsque le mouvement est fort (amplitude beaucoup plus grande que 0).

4.3.4 Résultats

Afin d'évaluer l'algorithme d'estimation du mouvement présenté à la section précédente, nous avons effectué deux expériences. La première consiste à estimer les paramètres de mouvement d'une séquence d'images dont on connaît exactement le mouvement (mouvement synthétique). Cette expérience a pour but de valider notre algorithme d'optimisation. La deuxième expérience est effectuée sur une séquence de mouvement naturel. Elle est destiné à valider le modèle d'estimation du mouvement proposé, ainsi que les hypothèses sous-jacentes concernant les régions.

Nous présentons dans la suite de cette section les expériences effectuées et les résultats obtenus.

4.3.4.1 Résultats pour des mouvements synthétiques

Pour effectuer cette expérience, nous avons généré des mouvements synthétiques à partir d'une image naturelle (trame 3 de la séquence "*Miss America*"). Quatre types de mouvement ont été générés: translation, rotation dans le plan d'image, divergence (effet de zoom) et une combinaison de divergence-rotation (divergence suivie de rotation). Dans le cas de la rotation et celui de la divergence, le centre géométrique de l'image a été choisi comme le centre de ces deux transformations. La première ligne des tableaux 4.1, 4.2, 4.3 et 4.4 montrent les valeurs exactes des paramètres pour ces quatre types de mouvement. Les images générées synthétiquement sont considérées comme images à l'instant t_-, tandis que l'image originale est prise pour la trame au temps t (voir la figure 4.4). Notons que les déplacements ne sont pas des multiples de la distance entre deux pixels, et que les images synthétiques sont obtenues par une interpolation bicubique; de sorte qu'il n'y ait pas d'appariement

parfait entre les trames aux temps t_- et t.

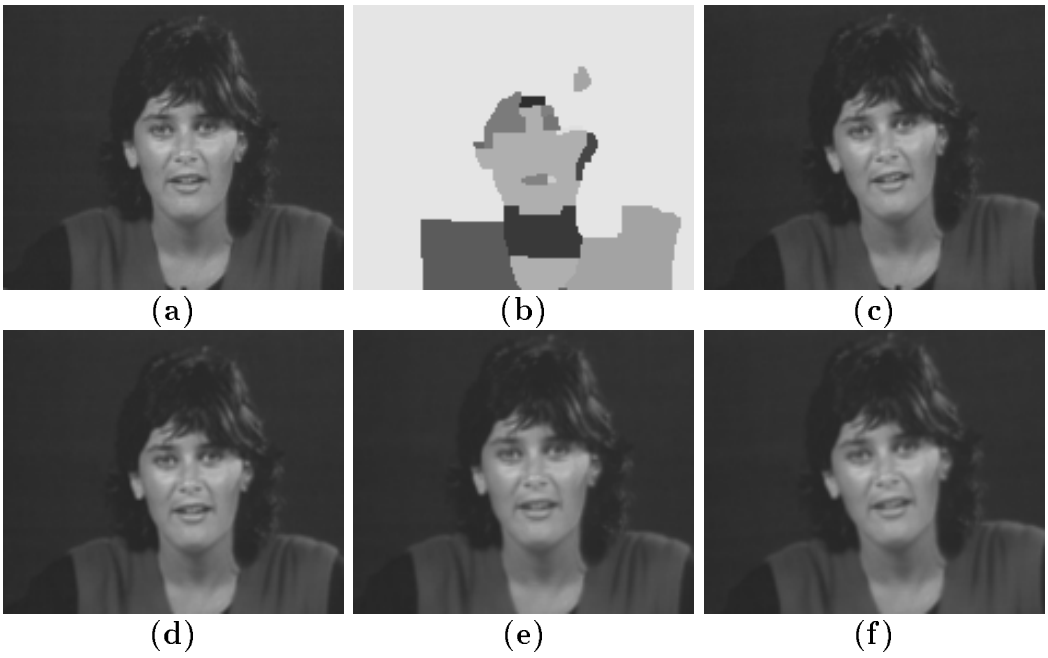


FIG. 4.4 - Images obtenues par la génération des mouvements synthétiques; (a) original, (b) segmentation spatiale de l'original (20 régions), (c) rotation, (d) translation , (e) divergence, (f) divergence suivie de rotation.

Pour chacun des quatre types de mouvement, deux séries de résultats ont été obtenues. L'une vient d'une estimation utilisant une segmentation exacte (l'image contient une seule région). L'autre est obtenu d'une estimation utilisant une sur-segmentation de l'image (l'images au temps t est segmentée spatialement en 20 régions par l'algorithme MDL, comme le montre la figure 4.4b). Les résultats sont présentés et analysés dans les points 1 et 2 ci-après:

1. **Cas d'une seule région**

   Il s'agit du cas où le centre de gravité de la région coïncide parfaitement avec le centre de rotation et de divergence. Donc, les paramètres de mouvement estimés devraient être égaux à ceux que nous avons choisis. La deuxième ligne des tableaux 4.1, 4.2, 4.3, 4.4 présente les valeurs estimées. Comme on peut le

constater, la différence entre les valeurs estimées et les valeurs exactes est très petite. Ce qui montre une très bonne convergence de l'algorithme d'optimisation. L'utilisation de la méthode Gauss-Newton est donc justifiée.

	a_1	a_2	b_{11}	b_{12}	b_{21}	b_{22}
ϕ	-3.5000	-3.5000	0.000	0.000	0.000	0.000
$\widehat{\phi}$	-3.4981	-3.4981	0.000	0.000	0.000	0.000

TAB. 4.1 - Valeurs exactes et estimées des paramètres de translation

	a_1	a_2	b_{11}	b_{12}	b_{21}	b_{22}
ϕ	0.000	0.000	0.004	-0.087	0.087	0.004
$\widehat{\phi}$	-0.039	0.044	0.004	-0.087	0.087	0.004

TAB. 4.2 - Valeurs exactes et estimées des paramètres de rotation

	a_1	a_2	b_{11}	b_{12}	b_{21}	b_{22}
ϕ	0.000	0.000	-0.048	0.000	0.000	-0.045
$\widehat{\phi}$	-0.024	-0.024	-0.047	0.000	0.000	-0.044

TAB. 4.3 - Valeurs exactes et estimées des paramètres de divergence

	a_1	a_2	b_{11}	b_{12}	b_{21}	b_{22}
ϕ	0.000	0.000	0.043	-0.091	-0.091	-0.043
$\widehat{\phi}$	-0.063	0.021	0.044	-0.090	-0.090	-0.040

TAB. 4.4 - Valeurs exactes et estimées des paramètres du mouvement divergent suivi d'une rotation

2. **Cas de la sur-segmentation**

Nous cherchons à vérifier par cette simulation la validité du choix du centre de gravité des régions comme origine du système de coordonnées dans le modèle de mouvement (voir l'équation (4.2)). Dans le cas de sur-segmentation, les centres de gravité des régions contenus dans l'image ne coïncident plus avec les centres de rotation et de divergence réels. Par conséquent, les paramètres

de mouvement des régions ne sont plus ceux que nous avons choisis. L'interprétation des résultats numériques n'est donc pas évidente. Nous choisissons de présenter ici une comparaison des champs de déplacement obtenus lorsque l'image est sur-segmentée et lorsque la segmentation est exacte (une seule région), car même si les résultats numériques sont différents, le déplacement des pixels doit être le même. Dans la figure 4.5, les images a, d, g, j représentent respectivement les champs de déplacement translationnel, rotationnel, divergent et divergent-rotationnel du cas d'une seule région ; b, e, h, k sont ceux du cas où l'image est sur-segmentée et c, f, i, l montrent la différence des déplacements pour chaque type de mouvement.

Cette comparaison nous permet de constater visuellement que, même si les centres de gravité des régions sont éloignés des centres de rotation ou de divergence, le déplacement estimé décrit très bien le mouvement des régions. La plupart des erreurs sont dues au fait que la taille des régions correspondantes est trop petite.

4.3.4.2 Résultats pour des mouvements naturels

Nous choisissons, dans cette expérience, d'estimer le mouvement naturel entre deux trames de chacune des trois séquences (sous-échantillonnée temporellement par trois) bien connues et typiques de la vidéo-conférence et du visiophone. Il s'agit des séquences "*Miss America*" , "*Carphone*" , "*Foreman*" (une description de ces séquence se trouve à la section 5.1). La première colonne des figures 4.6, 4.7 et 4.8 présente les résultats obtenus[1]. Ils montrent que qualitativement, les mouvements estimés décrivent assez bien les mouvements réels observés, et que les images prédites

1. Afin de mettre en évidence l'amélioration qu'apporte chacune des trois étapes de l'algorithme, les résultats de ces étape sont présentés en colonnes dans les mêmes figures.

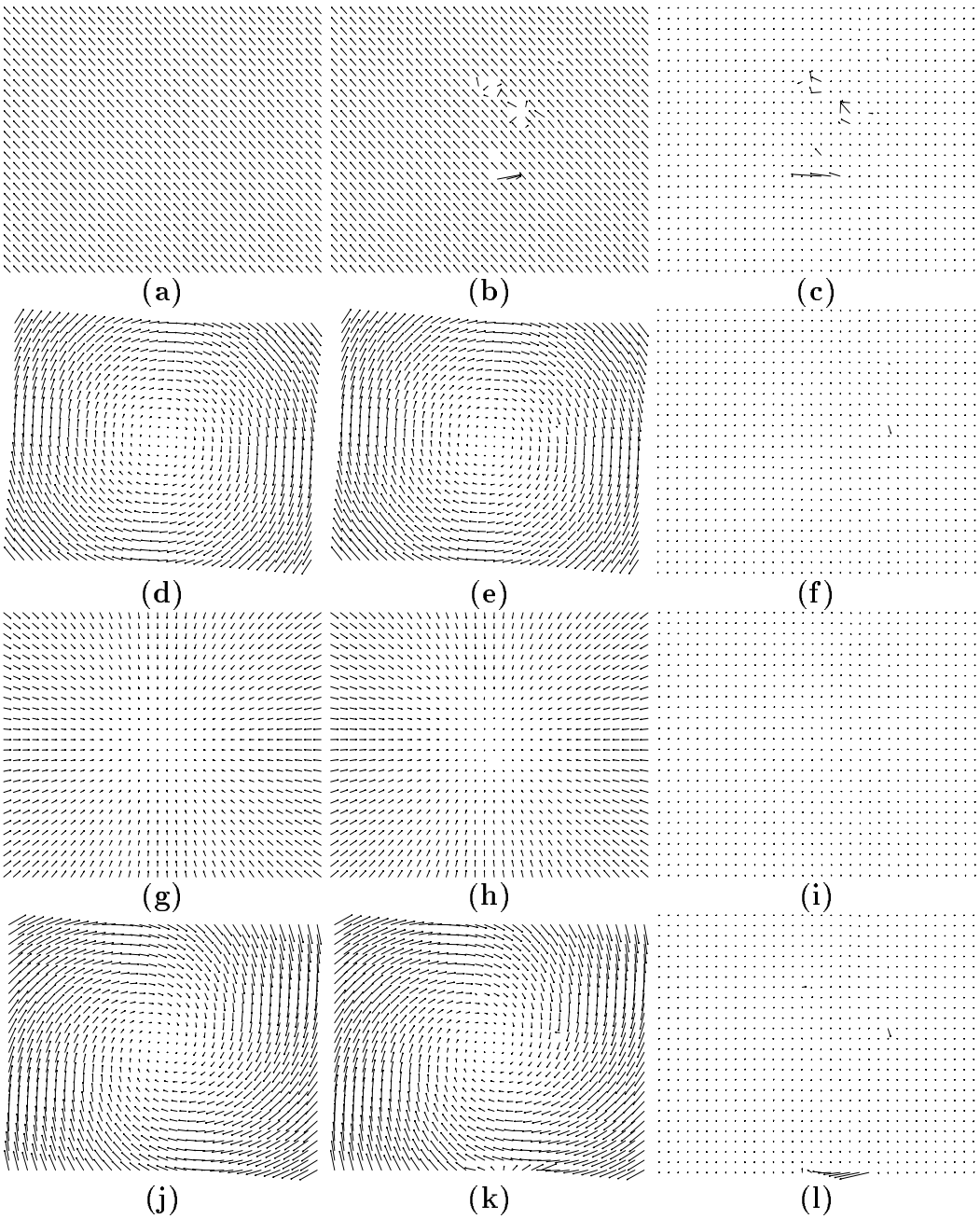


FIG. 4.5 - Comparaison des champs de déplacement obtenus lorsque la segmentation est exacte et quand l'image est sur-segmentée; a, d, g, j sont les champs de déplacement translationnel, rotationnel, divergent et divergent-rotationnel obtenus dans le cas où la segmentation est exacte; b, e, h, k correspondent au cas de la sur-segmentation; c, f, i, l sont les différences entre les deux cas; les champs ont été sous-échantillonnés par 4 dans chaque direction

sont bien compensées par le mouvement sauf pour les cas suivants:

- les régions de très petite taille où le problème d'estimation est sous-déterminé;
- les régions où la segmentation spatiale n'est pas capable de distinguer deux régions de même intensité (ou presque) mais de mouvement différent. Malgré des erreurs d'estimation, l'erreur de prédiction de ces régions est, comme prévu, très faible, puisque l'intensité y est presque uniforme (examiner la chevelure de "*Miss America*" et du personnage dans "*Carphone*");
- le cas où le mouvement est très fort et où il y a un manque de structure photométrique dans les régions, par exemple la séquence "*Foreman*" . Il est évident qu'ici, les erreurs sont dues essentiellement à des faiblesses bien connues de notre algorithme d'optimisation, qui utilise une solution initiale nulle et est basé sur le gradient.

4.4 Fusion des régions adjacentes

Nous avons déjà mentionné, lors de la description du modèle de région, que seul un certain nombre de contours d'intensité de l'image correspondent aux contours de mouvement, et qu'en conséquence, les régions d'intensités sont des sous-objets des objets physiques en mouvement dans la scène. Sous cette hypothèse, il est évident qu'une description du mouvement de tous les pixels de l'image par le mouvement des régions d'intensité contient encore beaucoup de redondances au sens du mouvement. L'objectif de cette étape est de réduire ces redondances en regroupant les régions adjacentes qui ont des mouvements semblables.

Ce regroupement correspond à une fusion des régions et présente plusieurs avantages. Premièrement, la fusion des petites régions pour former des régions plus grandes de mouvement cohérent, pourrait améliorer l'estimation des paramètres,

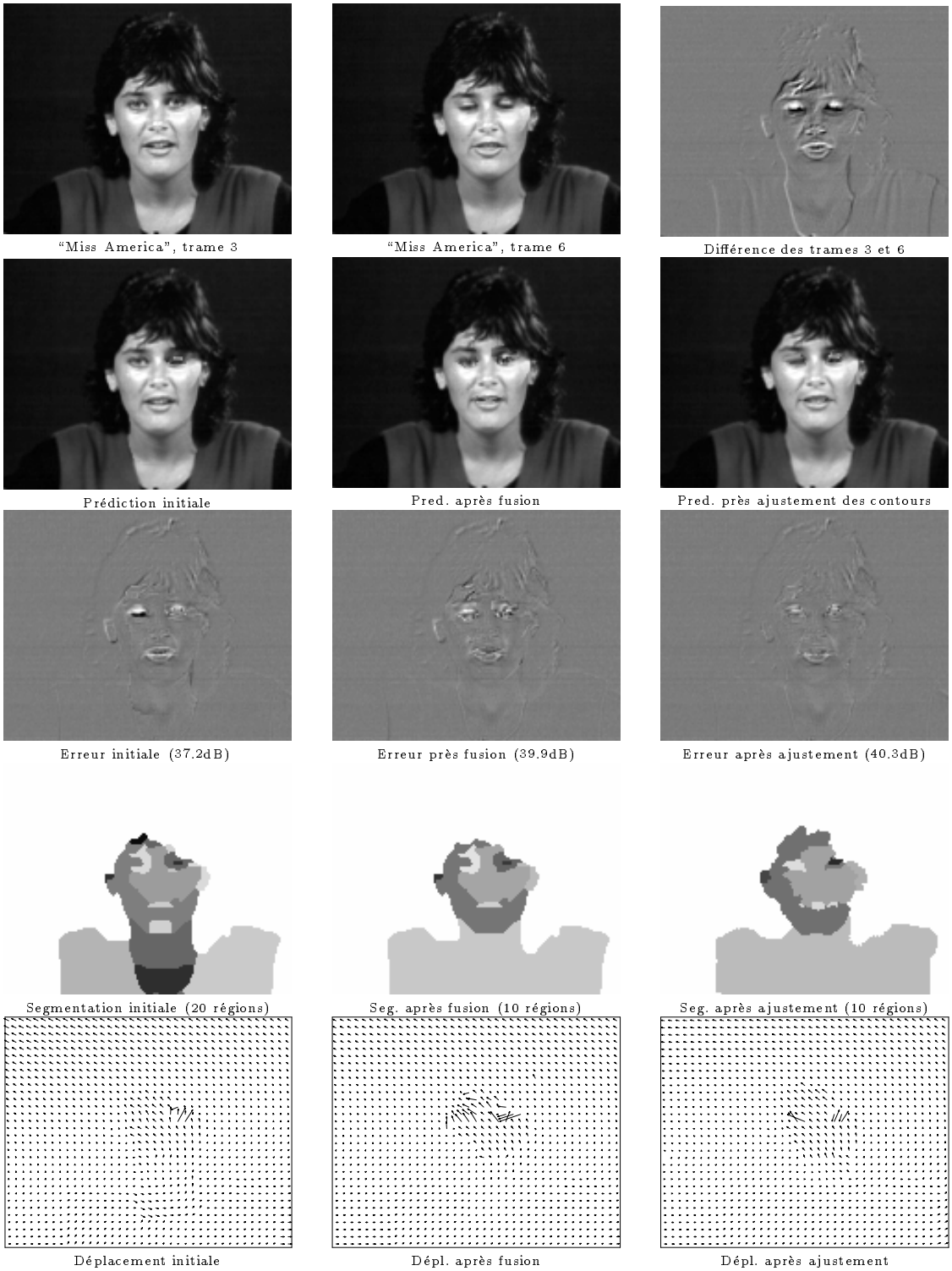


FIG. 4.6 - "*Miss America*" : images originales et les résultats de l'étape d'estimation du mouvement des régions d'intensité, de fusion et d'ajustement des contours; les champs de déplacement ont été sous-échantillonnés par 4 dans chaque direction; l'amplitude des vecteurs a été amplifiée 2 fois.

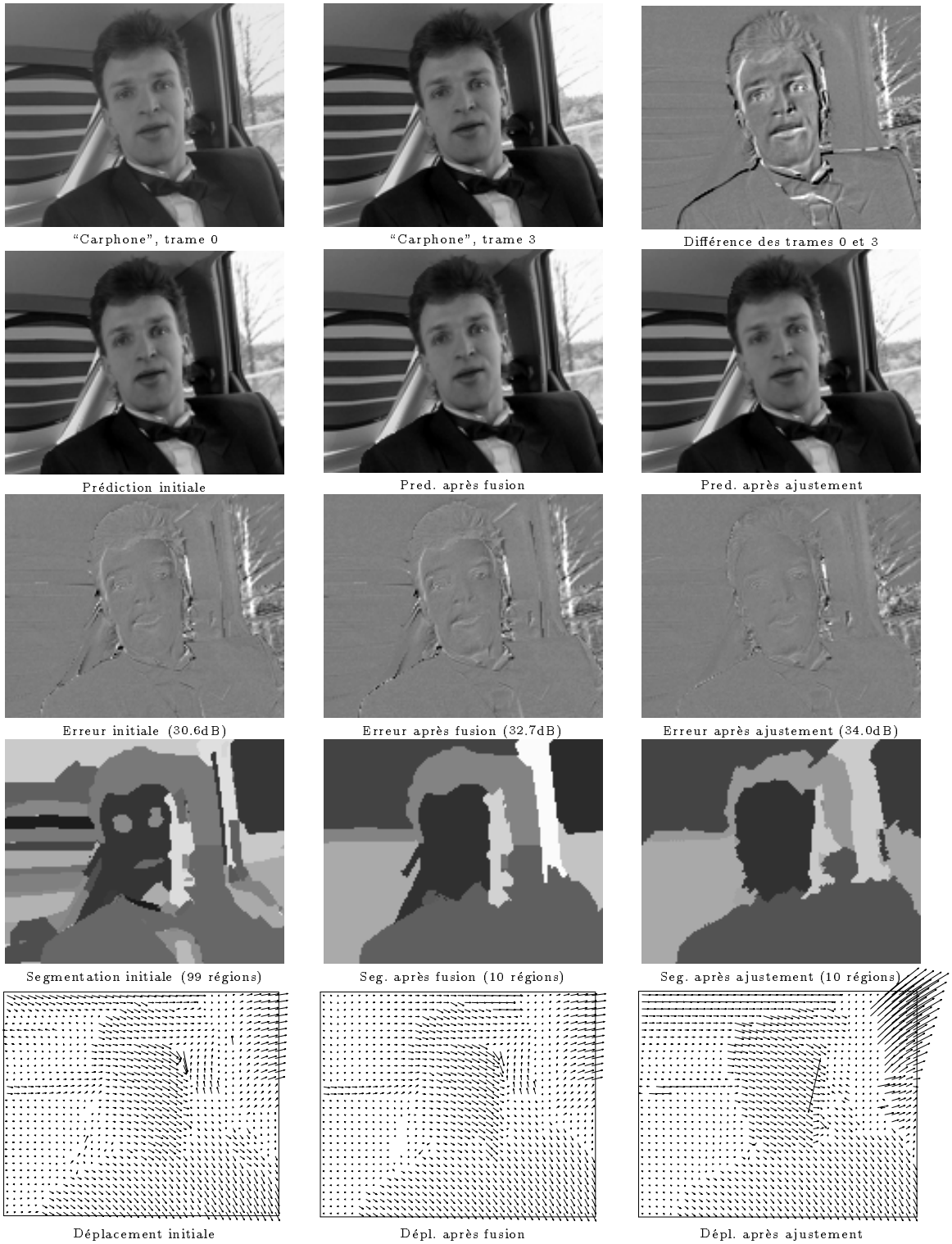


FIG. 4.7 - "*Carphone*" : images originales et les résultats de l'étape d'estimation du mouvement des régions d'intensité, de fusion et d'ajustement des contours; les champs de déplacement ont été sous-échantillonnés par 4 dans chaque direction; l'amplitude des vecteurs a été amplifiée 2 fois.

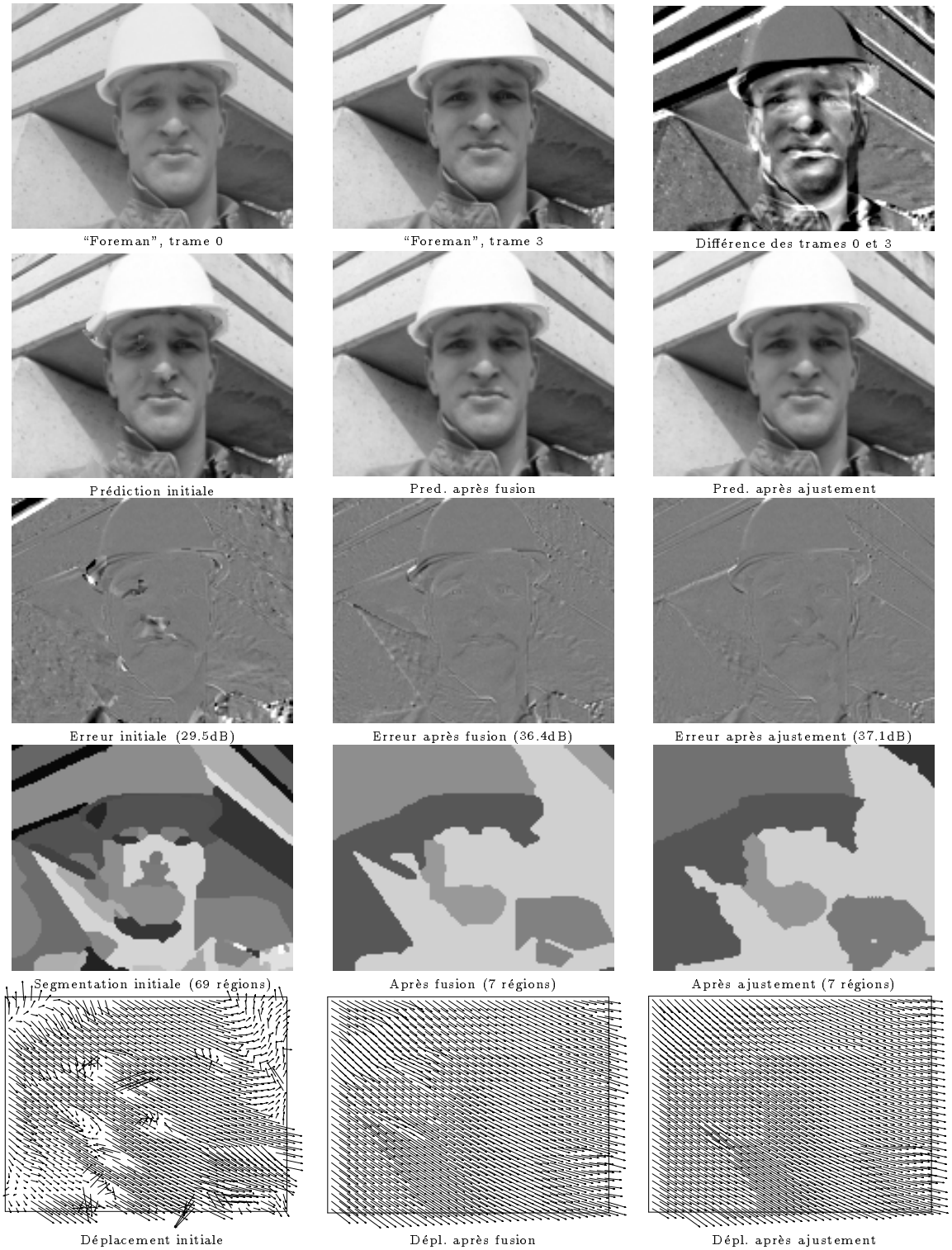


FIG. 4.8 - "*Foreman*" : images originales et les résultats de l'étapes d'estimation du mouvement des régions d'intensité, de fusion et d'ajustement des contours; les champs de déplacement ont été sous-échantillonnés par 4 dans chaque direction; l'amplitude des vecteurs a été amplifiée 2 fois.

puisque plus la taille d'une région est grande, plus elle procure de données pour la minimisation lors de l'étape d'estimation du mouvement; donc, par la fusion, la difficulté de l'estimation du mouvement des régions de petite taille pourrait être résolue. Deuxièmement, la fusion permettrait de simplifier la carte de segmentation de l'image; cela est très important pour le codage, car une partition contenant peu de contours nécessite moins de bits pour son codage, donc est plus économique pour sa transmission.

Nous pouvons modéliser le problème de fusion en considérant qu'une partition "optimale" au sens du codage doit être celle qui permet de minimiser en même temps l'erreur de prédiction compensée par le mouvement et le nombre de régions dans l'image. Par cette définition, la partition initiale basée sur l'intensité Ψ et leurs paramètres de mouvement peuvent être modifiés par le regroupement des régions voisines, afin de minimiser une fonction de coût définie par

$$\min_{\{N,\Phi\}} \sum_{n=1}^{N} \sum_{\mathbf{x}\in\psi_n} [g(\mathbf{x},t) - \tilde{g}(\mathbf{x} - h(\mathbf{x},\phi_n), t_-)]^2 + \lambda_1 N \tag{4.17}$$

Dans cette expression, le premier terme est l'erreur de prédiction compensée par le mouvement, qui permet de minimiser l'erreur résiduelle du codage, tandis que le second terme encourage la fusion des régions voisines ayant des paramètres de mouvement similaires. À la limite, lorsque $\lambda_1 \to \infty$, toutes les régions seront fusionnées pour former une région unique; alors que si $\lambda_1 = 0$, deux régions seraient fusionnées seulement au cas où l'erreur résiduelle de la région résultante (avec ses nouveaux paramètres de mouvement) est plus petite que la somme de leur erreur avant fusion.

4.4.1 Stratégie de minimisation

Le problème d'optimisation (4.17) est difficile, et il semble qu'il n'existe pas de méthodes de résolution permettant d'obtenir un optimum global. Nous proposons dans cette étude une méthode heuristique basée sur une représentation des régions

par un "graphe des voisins" qui est construit de la façon suivante:

- Chaque noeud du graphe représente une région de l'image; ce noeud contient les informations disponibles de la région, à savoir: l'erreur de prédiction, les paramètres de mouvement et la position de tous les pixels.

- Il y a un arc entre chaque paire de noeuds (i, j). Cet arc porte une mesure, notée $E(i+j)$, qui est définie de la manière suivante:

  - ◇ Si les deux noeuds d'extrémité représentent deux régions voisines, alors $E(i+j)$ est l'erreur quadratique de prédiction compensée par le mouvement de la nouvelle région, résultante d'une fusion de ces deux noeuds. Donc, cette erreur est calculée en supposant d'abord que ces deux régions sont fusionnées, puis en y appliquant l'algorithme d'estimation de mouvement défini par l'équation (4.6).

  - ◇ Si les deux noeuds d'extrémité ne sont pas deux régions voisines, alors $E(i+j) = \infty$.

La figure 4.9 donne l'exemple d'un graphe construit selon les règles établies ci-haut.

En se basant sur ce graphe, le processus de fusion est effectué itérativement. À chaque itération, les noeuds sont visités un à un. Lors de la visite d'un noeud, on calcule, pour chacun de ses voisins, le gain (ou perte) sur l'erreur de prédiction advenant l'hypothèse de leur fusion. En d'autres termes, on calcule

$$G_{i,j} = E(i) + E(j) - E(i+j)$$

où $G_{i,j}$ est le gain, $E(i)$, $E(j)$ sont les erreurs (quadratiques) résiduelles des régions ψ_i et ψ_j. La fusion aura lieu si ce gain est supérieur à un seuil défini par la valeur de λ_1; c'est à dire $G_{i,j} > -\lambda_1$, car:

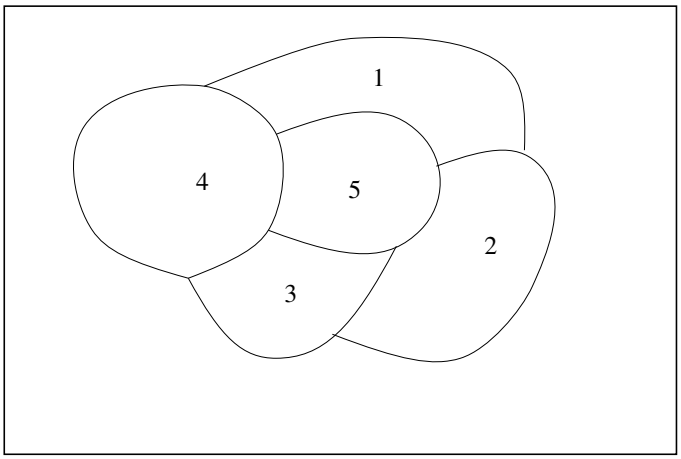
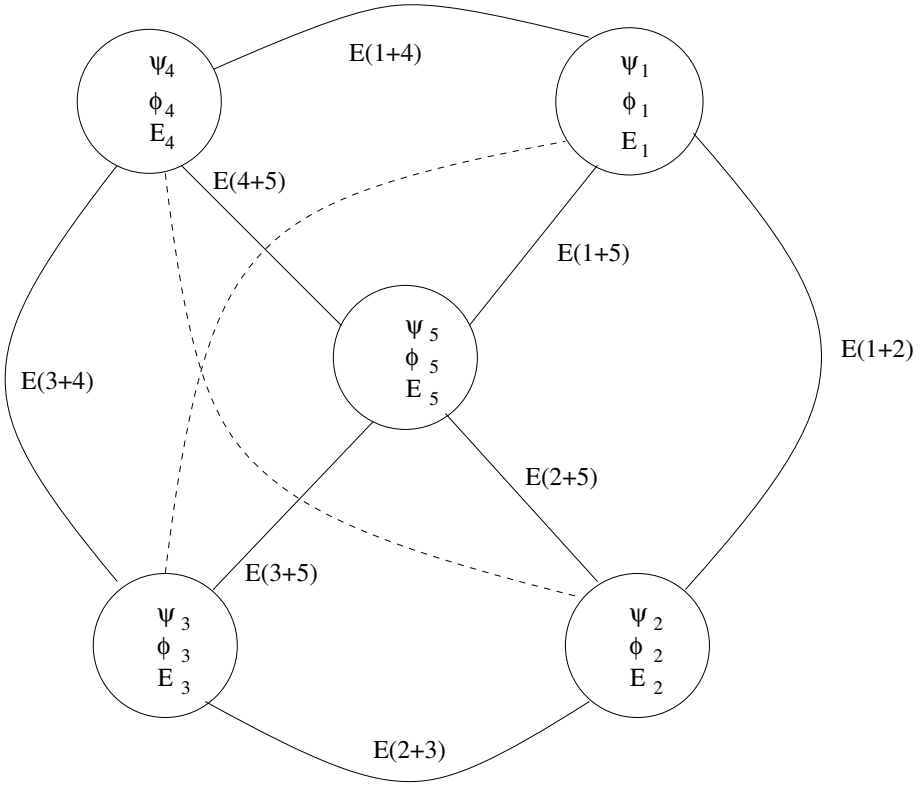


Image segmentée (5 régions)



Graphe "des voisins"

FIG. 4.9 - Exemple de construction d'un "graphe des voisins"

avant la fusion, le coût total est

$$C_{avant} = \sum_{n=1}^{N} \sum_{x \in \psi_n} DPD^2 + \lambda_1 N$$

après une fusion de deux régions, ce coût devient

$$C_{apres} = \sum_{n=1}^{N-1} \sum_{x \in \psi_n} DPD^2 + \lambda_1 (N-1)$$

d'où, la variation du coût est

$$\begin{aligned} \Delta C &= C_{avant} - C_{apres} \\ &= \sum_{n=1}^{N} \sum_{x \in \psi_n} DPD^2 + \lambda_1 N - \sum_{n=1}^{N-1} \sum_{x \in \psi_n} DPD^2 - \lambda_1 (N-1) \\ &= E(i) + E(j) - E(i+j) + \lambda_1 \\ &= G_{i,j} + \lambda_1 \end{aligned}$$

donc, pour faire diminuer la fonction de coût, il faut que $\Delta C > 0$; autrement dit, il faut que $G_{i,j} > -\lambda_1$.

Après chaque fusion, le graphe doit être mis à jour immédiatement. Le processus continu jusqu'à ce qu'il n'y ait plus de fusion possible.

Notons que cette méthode de résolution ne garantit pas l'obtention une carte de segmentation optimale. De plus, le résultat de la fusion peut dépendre de l'ordre dans lequel les noeuds sont visités; la résolution de ce problème n'est pas facile et dépasse le cadre de notre étude. Nous laissons au travaux futurs l'étude de ce problème.

4.4.2 Résultats

L'algorithme de fusion décrit ci-haut a été appliqué sur les cartes de segmentation spatiale des séquences "*Miss America*" , "*Carphone*" et "*Foreman*" utilisées lors de l'étape d'estimation du mouvement. La deuxième colonne des figures 4.6, 4.7, 4.8

montre les images prédites, les erreurs de prédiction, les cartes de segmentation ainsi que les champs de déplacement obtenus après cette étape de fusion. En comparant ces résultats avec ceux de l'étape précédente (les images de la première colonne de ces mêmes figures), on peut constater qu'il y a une réduction importante du nombre de régions. De plus, la qualité des images prédites et celle des champs de déplacement ont été nettement améliorées; D'autre part, les mesures de "peak prediction gain" (PPG –voir la définition à l'équation (5.1)) indiquent des gains de 2 à 6dB par rapport à ceux obtenus avant la fusion. Ce qui confirme les améliorations observées. Néanmoins, il reste quand même des erreurs assez importantes au voisinage des contours. La section suivante décrira le traitement destiné à faire diminuer ces erreurs.

4.5 Ajustement des contours

Après l'étape de fusion, la partition simplifiée Ψ ne contient plus qu'un petit nombre de régions ayant des mouvements homogènes. Les paramètres de mouvement sont aussi estimés avec plus de précision grâce à une augmentation de la taille des régions. Nous profitons de ces faits pour ajuster les contours d'intensité (qui sont considérés jusqu'à maintenant comme des contours de mouvement), afin qu'ils soient consistants avec le mouvement des régions (rappelons qu'il peut y avoir un certain écart entre les deux types de contour, dû à la discrétisation de l'image ou aux ombrages). Étant donné la nature de ces erreurs, nous ne cherchons pas à obtenir par cet ajustement un changement radical des contours existants, ni à créer de nouvelles régions. Notre objectif est de les modifier seulement de quelques pixels de l'un ou de l'autre côté de ces contours. Pour réaliser cet ajustement, nous proposons le modèle suivant:

$$\min_{k\in\Theta(\mathbf{x})} \quad [\tilde{g}(\mathbf{x}-\mathbf{d}(\mathbf{x},\phi_k),t_-)-g(\mathbf{x},t)]^2 + \lambda_2 \sum_{\mathbf{y}\in\eta_2(\mathbf{x})}[1-\delta(k-\mathcal{S}(\mathbf{y},t))], \quad \forall \mathbf{x}\in\mathcal{B}(\psi_n),\forall n \tag{4.18}$$

où:

- $\mathcal{S}(\mathbf{y},t)=i \quad$ si $\quad \mathbf{y}\in\psi_i$;
- $\Theta(\mathbf{x})=\{\mathcal{S}(\mathbf{x},t)\}\cup\{\mathcal{S}(\mathbf{y},t),\mathbf{y}\in\eta_1(\mathbf{x})\}$ est l'ensemble des étiquettes de régions voisines de la région contenant $\mathbf{x}$ et se trouvant dans $\eta_1(\mathbf{x})$, le voisinage de premier ordre de $\mathbf{x}$ (4 voisins les plus près);
- $\mathcal{B}(\psi_n)$ dénote la frontière de la région ψ_n;
- η_2 est le voisinage de second ordre de $\mathbf{x}$ (8 voisins les plus près).

Le deuxième terme de ce modèle décrit la complexité des contours et est inspiré des modèles MRF (Markov Random Field) très utilisés en estimation du mouvement [KD92]. Ce terme favorise les contours lisses.

4.5.1 Méthode de résolution

La minimisation de (4.18) se fait par la méthode de relaxation déterministe de Jacobi [GO93] et par une recherche exhaustive dans l'espace d'état $\Theta(\mathbf{x})$ pour chaque pixel de contour $\mathbf{x}$. Plus concrètement, la minimisation se déroule itérativement et chaque itération est réalisée de la manière suivante:

1. Après l'extraction des contours, on calcule, pour chaque pixel de contour $\mathbf{x}$ et pour toutes les étiquettes contenues dans $\Theta(\mathbf{x})$, les valeurs de la fonction d'énergie (4.18) (en utilisant les paramètres de mouvement de l'itération précédente); l'étiquette, qui donne une énergie la plus faible, sera gardée en mémoire

comme l'étiquette probable de ce pixel à l'itération suivante, car le changement d'étiquette peut ne pas avoir lieu; nous verrons les raisons ci-après.

2. Une fois que tous les pixels de contour ont été visités, les étiquettes de région seront mises à jour. Comme signalé, nous précisons ici un détail d'ordre pratique mais important concernant la mise à jour des étiquettes: puisqu'une région est définie dans ce travail comme l'ensemble des pixels portant la même étiquette de région, un point de frontière de deux régions se trouve donc entre deux pixels de différentes étiquettes; autrement dit, un point de contour (fictif) est toujours défini par une paire de pixels ayant différentes étiquettes de région (voir l'exemple montré à la figure 4.10); afin d'empêcher la création des pixels isolés par un échange des étiquettes entre deux pixels d'une paire, seul l'un de ces deux pixels peut changer d'étiquette (celui qui permet une plus grande réduction de la fonction d'énergie).

3. Les paramètres de mouvements des régions dont le contour a été modifié seront ré-estimés en utilisant l'algorithme d'estimation du mouvement. La valeur actuelle des paramètres est utilisée comme solution initiale.

4. Une autre itération commence jusqu'à ce que le nombre maximal d'itérations (fixé à 15) soit atteint.

4.5.2 Résultats

L'algorithme d'ajustement des contours décrit ci-haut a été appliqué sur les cartes de segmentation fusionnées des séquences "*Miss America*" , "*Carphone*" et "*Foreman*" . La troisième colonne des figures 4.6, 4.7, 4.8 montre les images prédites, les erreurs de prédiction, les cartes de segmentation ainsi que les champs de déplacement obtenus après cette étape. En comparant ces résultats avec ceux de

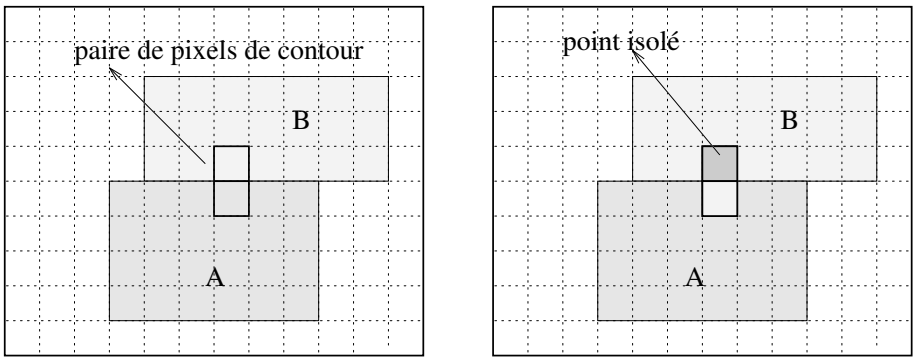


FIG. 4.10 - Illustration de la création des pixels isolés suite à un échange des étiquettes dans une paire de points de contour

l'étape de fusion (les images de la deuxième colonne de ces mêmes figures), on peut constater qu'il y a une réduction importante de l'erreur de prédiction, particulièrement au voisinage des contours. De plus, la qualité des images prédites et celle des champs de déplacement ont été nettement améliorées. Des gains de plus de 1dB par rapport à ceux obtenus lors de l'étape de fusion confirment d'ailleurs les améliorations observées. Il est intéressant de remarquer que cette étape a permis d'obtenir de bonnes compensations de mouvement dans les régions où le mouvement est très complexe ou très fort; par exemple, les yeux et la bouche de "*Miss America*" ou la fenêtre de la voiture dans "*Carphone*" .

4.6 Discussion sur la complexité de l'algorithme

Dans cette étude, nous nous concentrons nos efforts sur l'aspect modélisation et sur la validation de notre approche. Des problèmes d'ordre pratique, telle l'optimisation des calculs et de la programmation, ne sont pas du tout abordés et sont laissés à des travaux futurs. Néanmoins, nous discutons brèvement dans cette section de la complexité intrinsèque de notre algorithme puisque cela joue un rôle important dans l'application visée (visiophone).

Dans l'ensemble, l'algorithme proposé est simple et facile à implanter. Comme

nous pouvons le constater d'après les sections précédents, le coeur de notre algorithme est le module d'estimation du mouvement, puisqu'il est utilisé à la fois dans l'estimation initiale et dans la mise à jours des paramètres de mouvements lors de chacune des trois étapes de l'algorithme. Ce module occupe donc la majeure partie du temps total de fonctionnement requis; d'autres calculs et traitements dans les étapes de fusion et de l'ajustement des contours sont minimes et ne nécessitent que très peu de temps supplémentaire.

En ce qui concerne l'estimation du mouvement, c'est l'interpolation de la fonction luminance et la détermination de son gradient qui nécessitent le plus de calculs, car on doit effectuer ces deux opérations pour chaque pixel de l'image. À titre d'exemple, une implantation (non optimale) de ces deux opérations (convolutions- voir la section 4.3.1.1) requiert 110 additions et 106 multiplications pour chaque pixel; le temps de calcul moyen pour estimer et segmenter une image au format QCIF est environ 15 minutes (sur une machine *DECstation 3100*). Il est utile de préciser que dans la version implantée, le champ de déplacement prend des valeurs réelles; il faut alors recalculer les coefficients des filtres avant chaque opération d'interpolation de la luminance et de son gradient, ce qui explique le nombre élevé des calculs. En pratique, les valeurs possibles des déplacements sont limitées (par exemple en utilisant une résolution de 1/2 pixel comme dans les standards H.261 ou MPEG); on peut donc calculer les coefficients des filtres une seule fois et les stocker dans des tableaux; par conséquent, la quantité de calculs peut être largement réduite.

Notons toutefois que l'algorithme proposé est fortement parallélisable. En effet, l'estimation du mouvement est totalement indépendante d'une région à l'autre; l'interpolation de la luminance et calcul du gradient des pixels sont aussi des opérations indépendantes; il en est de même pour le calcul de la fonction d'énergie lors de l'étape d'ajustement des contours. Une réduction du temps de calcul est donc possible avec une implantation parallèle.

Chapitre 5

Résultats pour les séquences d'images

La méthode d'estimation et de segmentation du mouvement présentée dans le chapitre précédent a été implantée sur une station de travail *DECstation 3100* en langage C. Des simulations ont été effectuées sur des séquences d'images à mouvement naturel. Les résultats sont observés sur le poste de visualisation *Viewstore 6000* disponible dans le laboratoire du Groupe Communications Visuelles de l'INRS-Télécommunications. Nous allons présenter et analyser ces résultats dans ce chapitre. De plus, pour mettre en valeur la performance de la méthode proposée, nous comparons ces résultats avec ceux de deux méthodes les plus connues: la méthode d'appariement de blocs et la méthode "basée pixel". Par ailleurs, dans le but de minimiser le débit de transmission, un codage avec perte des cartes de segmentation a été proposé dans [Cha95]. Afin de vérifier cette proposition, nous étudierons l'impact de ce codage sur la qualité des images prédites.

5.1 Séquences de test

Afin de réaliser les simulations, nous avons choisi trois séquences vidéo au format QCIF(176×144): "*Miss America*" , "*Carphone*" et "*Foreman*" . Ce sont des séquences vidéo bien connues et typiques de la vidéo-conférence et du visiophone. Les figures 5.1, 5.2 et 5.3 donnent en exemple quelques trames de chacune de ces trois séquences. Pour chaque séquence, les trames suivantes ont été utilisées lors des simulations:

- "*Miss America*" : 3, 6, 9, ..., 57
- "*Carphone*" : 3, 6, 9, ..., 57, 120, 123, 126, ..., 177
- "*Foreman*" : 3, 6, 9, ..., 57, 150, 153, 156, ..., 207

Les caractéristiques concernant le mouvement de ces séquences sont les suivantes:

- La séquence "*Miss America*" présente des mouvements de faible amplitude. En général, le mouvement y est relativement facile à estimer. Cependant, dans la région du visage de "*Miss America*" –particulièrement au niveau de la bouche et des yeux– le mouvement est rapide et complexe; ces régions sont très souvent des zones d'occlusion (donc, ne peuvent être prédites). D'autre part, la segmentation spatiale des images de cette séquence contient beaucoup de régions où l'hypothèse de coïncidence des contours d'intensité et de mouvement n'est pas respectée; par exemple, la chevelure et les manchettes de la chemise ont été confondues avec le fond qui est stationnaire.
- La séquence "*Carphone*" montre l'intérieur d'une voiture en mouvement (filmé par une caméra embarquée) où il y a un homme en train de parler. Le mouvement des trames 3–57 est de moyenne amplitude, sauf la région de la fenêtre où il est très fort (le paysage se trouvant à l'extérieur recule très rapidement en raison du déplacement de la voiture). Dans les trames 120–177, étant donné

la courte distance qui sépare le personnage de la caméra, chaque mouvement du personnage vers la caméra se traduit par un fort mouvement de divergence (effet de zoom). Des changements d'illumination globale sont aussi observés dans cette séquence.

– Les trames 3–57 de la séquence "*Foreman*" comporte principalement un mouvement de la caméra de forte amplitude. Ce mouvement est combiné avec un fort mouvement de la tête et des déformations du visage du personnage dans les trames 150–207. Les objets de cette scène sont très peu texturés, de sorte que la segmentation spatiale n'a pas été capable de distinguer les objets de même intensité, mais se déplaçant dans différentes directions; par exemple, le casque et le mur.

FIG. 5.1 - Exemple de quelques trames de la séquence "*Miss America*" .

5.2 Critères d'évaluation

Pour évaluer la qualité de notre algorithme, nous examinons les résultats selon des critères quantitatifs et qualitatifs.

FIG. 5.2 - Exemple de quelques trames de la séquence "*Carphone*" .

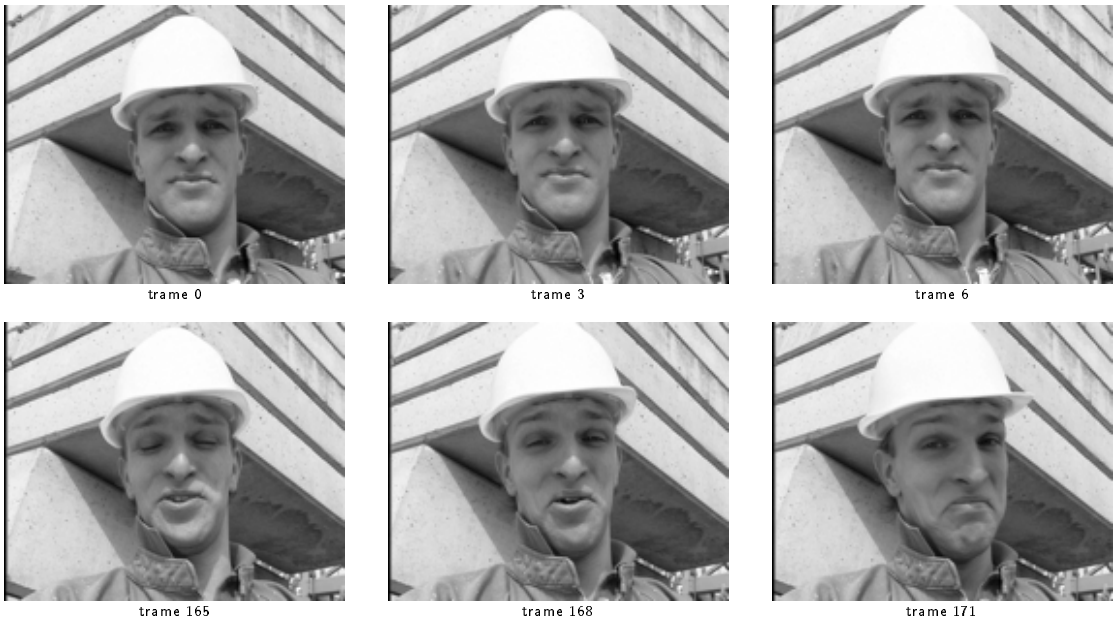


FIG. 5.3 - Exemple de quelques trames de la séquence "*Foreman*"

5.2.1 Critère quantitatif

Comme nous l'avons mentionné dans la section 4.3.1, il n'existe pas de mesures parfaitement adéquates permettant de quantifier la qualité des images reconstruites, puisque la perception visuelle d'un défaut dépend non seulement de son amplitude, mais également de sa position dans l'image. Néanmoins, il est nécessaire d'avoir une mesure objective pour évaluer la performance d'une méthode de codage et surtout pour comparer la performance de différentes méthodes. Dans cette étude, nous choisissons une mesure de distorsion entre l'image prédite et l'originale. Cette mesure, dénotée PPG (Peak Prediction Gain), est définie par

$$PPG = 10 \log_{10}(\frac{g_{max}^2}{EQM}), \tag{5.1}$$

où g_{max} est l'intensité maximale de l'image; dans notre cas $g_{max} = 255$; EQM est l'erreur quadratique moyenne entre l'image prédite $\hat{g}(\mathbf{x}, t)$ et l'originale $g(\mathbf{x}, t)$

$$EQM = \frac{1}{N} \sum_{\mathbf{x}} (g(\mathbf{x}, t) - \hat{g}(\mathbf{x}, t))^2 \tag{5.2}$$

où N est le nombre de pixels dans l'image. Soulignons que le PPG (en dB) ne donne qu'une idée sur l'ordre de grandeur de la distorsion; une valeur élevée de PPG ne signifie pas nécessairement que l'image prédite a une bonne qualité visuelle. Par conséquent, son interprétation doit prendre en considération le critère qualitatif présenté ci-après.

5.2.2 Critère qualitatif

Ce critère est en fait une opinion subjective que l'on peut avoir en comparant visuellement les images prédites avec les originales, (ou avec les images prédites de différentes méthodes lorsqu'il s'agit d'une comparaison) ou en examinant les images d'erreur et les champs de déplacement. Cette évaluation subjective permet

d'apprécier la qualité visuelle des images prédites, la visibilité des erreurs et la qualité du mouvement estimé [1].

Notons que pour fin d'affichage des résultats que nous allons présenter plus loin, les images d'erreur sont amplifiées par 2 et sont centrées au niveau de gris 128. Plus précisément, un pixel $\mathbf{x}$ de l'image d'erreur $e(\mathbf{x}, t)$ est construit selon la formule

$$e(\mathbf{x}, t) = 2 \cdot (g(\mathbf{x}, t) - \widehat{g}(\mathbf{x}, t)) + 128. \tag{5.3}$$

De plus, pour mieux faire apparaître les défauts, étant donné une résolution insuffisante de l'imprimante, les images d'erreur sont normalisées dans l'intervalle d'intensité [70 170]; autrement dit, l'intensité normalisée e_{norm} d'un point $\mathbf{x}$ de $e(\mathbf{x}, t)$ est calculée par

$$e_{norm}(\mathbf{x}, t) = \frac{e(\mathbf{x}, t)]_{70}^{170} - 70}{170 - 70} \tag{5.4}$$

où $e(\mathbf{x}, t)]_{70}^{170}$ définit la troncation de $e(\mathbf{x}, t)$ dans l'intervalle d'intensité [70 170]. Donc, l'amplitude des erreurs montrées est beaucoup plus forte qu'elle ne l'est en réalité. Conséquemment, ces images d'erreur ne doivent être interprétées qu'en les comparant avec celles des autres méthodes.

5.3 Résultats et discussion

Nous présentons dans cette section les résultats de simulation pour les trois séquences de test. L'analyse de ces résultats se fait en se basant sur les critères exposés dans la section précédente. De plus, pour mettre en valeur la performance de notre algorithme, nous comparons ses résultats avec ceux des méthodes d'appariement de blocs ("basée bloc") et "basée pixel". Mais avant de présenter les résultats, nous donnons quelques détails sur ces deux méthodes.

1. Théoriquement, au point de vue codage, un bon champ de déplacement ne correspond pas nécessairement au déplacement physique des objets de la scène, mais c'est celui qui produit la plus petite erreur de compensation de mouvement. En pratique, un champ de déplacement qui décrit bien le mouvement physique des objets de la scène produit également une bonne compensation de mouvement.

Méthode d'appariement de blocs

Il s'agit de la méthode adoptée dans les standards H.261 et MPEG (voir la section 3.1.2.1). Les simulations de cette méthode ont été réalisées avec les paramètres suivants:

- mode de recherche: recherche exhaustive basée sur un critère minimisant l'erreur quadratique
- dimension des blocs: 16×16;
- résolution: 1/2 pixel;
- dimension de la fenêtre de recherche: 32×32.

Méthode "basée pixel"

Nous avons choisi la méthode présentée dans [KD91] dont le modèle d'estimation du mouvement est composé de DPD et de contrainte de lissage (voir section 3.1.2.2). L'optimisation utilise l'approche multi-résolutions (3 niveaux de résolution) et se fait par la méthode de relaxation déterministe Gauss-Seidel.

Notons qu'avec la méthode "basée pixel", l'on peut obtenir des erreurs résiduelles très petites en diminuant le poids de la contrainte de lissage, mais cela rend le champ de déplacement beaucoup moins lisse (peu corrélé) qui nécessitera un débit élevé pour sa transmission. En principe, la comparaison des trois méthodes doit se faire à débits égaux. Mais dans le cadre de cette étude, ceci n'est pas possible, puisque nous n'avons pas encore de schéma de codage précis pour la méthode "basée région". Pour que la comparaison soit équitable, nous ajustons le poids de la contrainte de lissage de telle façon que le champ de déplacement soit visuellement acceptable et comparable à celui de la méthode "basée région". Cet ajustement est réalisé sur quelques trames typiques de la séquence; par la suite, la valeur du poids ainsi obtenue sera gardée constante pour toute la séquence.

Analyse des résultats

Examinons et comparons maintenant les résultats obtenus.

Les figures 5.4, 5.5, 5.6 montrent les mesures de PPG des trois séquences testées en utilisant les trois méthodes: "basée région", "basée bloc" et "basée pixel". Nous pouvons y constater qu'en moyenne, la méthode "basée région" [1] a un gain de 3 à 4dB supérieur à la méthode "basée bloc"; mais en même temps, elle est surpassée d'environ 2dB par rapport à la méthode "basée pixel" pour les séquences "*Miss America*" , "*Carphone*" , et la première partie de "*Foreman*" (de la trame 3 à 57); c'est tout à fait ce à quoi nous nous attendions. Par contre, pour les trames de 150 à 207 de "*Foreman*" , la méthode "basée région" a un gain d'environ 1dB comparativement à la méthode "basée pixel". De prime abord, ce résultat semble surprenant. Mais cela s'explique par le fait que le mouvement de cette séquence est presque translationnel et est très fort (mouvement de la caméra principalement); pour bien l'estimer, nous avons dû fournir à la méthode "basée pixel" une forte contrainte de lissage; cependant, lorsque le mouvement devient plus complexe, –c'est le cas des trames 150-207– cela provoque un sur-lissage important à travers des frontière de mouvement; ce qui explique les erreurs élevées mesurées.

Afin de permettre une appréciation subjective des résultats obtenus, nous présentons ici quelques exemples les plus significatifs. Les figures 5.7, 5.8, 5.9, 5.10, 5.11 montrent respectivement les résultats des trames: 3 de "*Miss America*" , 3 et 171 de "*Carphone*" , 36 et 156 de "*Foreman*" . Chaque jeu de résultats contient:

- l'image précédente et l'image de référence entre lesquelles le mouvement est estimé, la segmentation finale produite par la méthode "basée région" (après l'ajustement des contours);

- les images prédites par les trois méthodes: "basée bloc", "basée région" et

1. Examiner seulement les courbes marquées "sans perte", celles qui sont marquées "avec perte" sont obtenues avec un codage avec perte des cartes de segmentation que nous allons décrire et discuter dans la section suivante

"basée pixel";

- les images d'erreur (multipliées par 2 et normalisées);
- les champs de déplacement (sous-échantillonnés par 4 dans chaque direction, l'amplitude des vecteurs est amplifiée 2 fois).

L'examen de ces résultats nous permet d'en tirer des conclusions suivantes:

- Même si les segmentations finales ne sont pas parfaites, elles correspondent raisonnablement bien, autant à la forme qu'au nombre des objets physiques contenus dans l'image.
- Les champs de déplacement de la méthode "basée région" décrivent assez bien les mouvements physiques, même dans les régions où le mouvement est complexe ou très fort; par exemple, la bouche, les yeux des personnages dans les trois séquences, ou encore la fenêtre de la voiture dans "*Carphone*" . Notons que, dans ces régions, la méthode d'appariement de blocs n'arrive jamais à estimer correctement le mouvement; tandis que les champs de déplacement de la méthode "basée pixel" montrent clairement les effets de sur-lissage au voisinage des frontières de mouvement.
- La plupart des erreurs de la méthode "basée région" se concentrent au voisinage des contours d'intensité où elles sont très peu visibles, grâce à l'effet de masque du SVH. Dans le cas de la méthode "basée bloc", les erreurs sont beaucoup plus fortes et provoquent des "effets de bloc" très visibles sur les images prédites, en particulier dans les régions de la bouche et des yeux. Nous croyons que ces distorsions resteront visibles sur les images décodées si l'on utilise peu de bits pour coder l'erreur résiduelle. En ce qui concerne la méthode "basée pixel", les erreurs sont généralement petites et peu visibles; ce qui concorde avec les mesures élevées de PPG que nous avons présentées plus

tôt. Cependant, au cas où l'estimation du mouvement est mauvaise, on peut observer d'importantes distorsions; par exemple, la fenêtre de la voiture dans la trame 3 et 171 de "*Carphone*" ou le visage de l'homme dans la figure 5.9.

- En général, la qualité visuelle des images prédites par la méthode "basée région" est très bonne; elle dépasse de loin celle de la méthode d'appariement de blocs. De plus, les observations des résultats sur le poste de visualisation du système *Viewstore* démontrent que la qualité visuelle de la méthode "basée régions" est équivalente à celle de la méthode "basée pixel" (pour des estimations du mouvement de qualité comparable), malgré la supériorité de cette dernière en terme de PPG, ceci s'explique par une concentration des erreurs au voisinage des contours d'intensité, comme nous l'avons mentionné plut haut.

5.3.1 Impact du codage avec perte des contours sur la qualité des images prédites

La transmission de l'information de mouvement nécessite le codage de ses deux composantes: la carte de segmentation et les paramètres de mouvement. Pour pouvoir se concentrer sur le problème d'estimation et de segmentation du mouvement, nous laissons le codage des paramètres à des travaux futurs, alors que le codage des contours a été étudié dans [Cha95]. Afin de minimiser le débit de transmission, un codage avec perte des contours de mouvement y est proposé. Ce procédé consiste à combiner deux techniques de codage de contours. La première fait l'approximation des régions de grande taille par des polygones. La seconde encode sans perte le contour des petites régions par la méthode "chain coding" (voir [Cha95] pour plus de détails). Notons que l'approximation a été réalisée en considérant uniquement des propriétés géométriques des contours; donc, elle n'est pas optimale au sens du mouvement.

Dans cette section, nous étudions l'impact de cette technique de codage sur la qualité des images prédites. Pour cela, nous avons ré-estimé le mouvement des régions codées avec perte. Le nouveau champ de déplacement (ou les nouveaux paramètres de mouvement, qui doivent être envoyés au décodeur) a été utilisé ensuite pour reconstruire l'image prédite par compensation de mouvement. Nous présentons maintenant les résultats obtenus et les comparons avec ceux du cas sans perte.

Les courbes marquées "avec perte" dans les figures 5.4, 5.5, 5.6, montrent les mesures de PPG obtenues avec des approximations jugées "haute fidélité" qui requièrent un débit de transmission allant de 3 à 5 Kb/s pour une fréquence de trames de 10 Hz/s [Cha95] (les images "segmentation avec perte" des figures 5.12, 5.13, 5.14 en sont quelques exemples). En moyenne, pour les trois séquences testées, le codage avec perte présente une perte de gain d'environ 1dB par rapport au cas sans perte. Néanmoins, nous constatons qu'il n'y a pas de dégradation notable de la qualité visuelle des images prédites (voir les figures 5.12, 5.13, 5.14). Ceci s'explique par le fait que les approximations sont appliquées seulement sur les régions de grande taille; donc, le nombre de pixels en erreur de segmentation est très petit par rapport à la taille des régions; par conséquent, l'hypothèse de mouvement dominant s'applique encore très bien; ce qui implique que les erreurs de compensation de mouvement se concentrent surtout au voisinage des contours d'intensité où elles sont peu visibles.

En nous appuyant sur ces résultats, nous pouvons conclure que, pour une "bonne" approximation de la carte de segmentation, le codage avec perte des contours de mouvement est envisageable sans que cela n'entraîne une perte excessive de la qualité visuelle des images prédites. Toutefois, nous pensons qu'il serait plus prudent d'effectuer d'autres études plus poussées pour déterminer les seuils critiques de l'approximation, avant de se prononcer définitivement sur la validité d'un codage avec perte des cartes de segmentation.

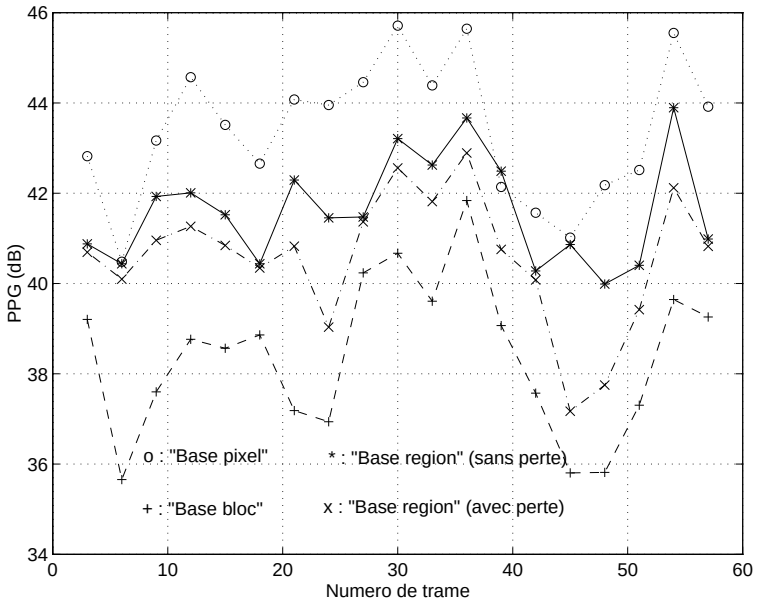
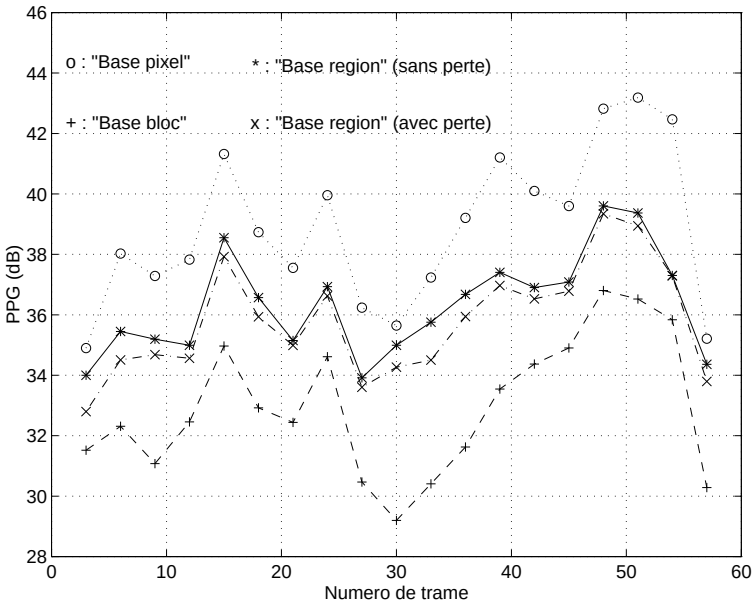
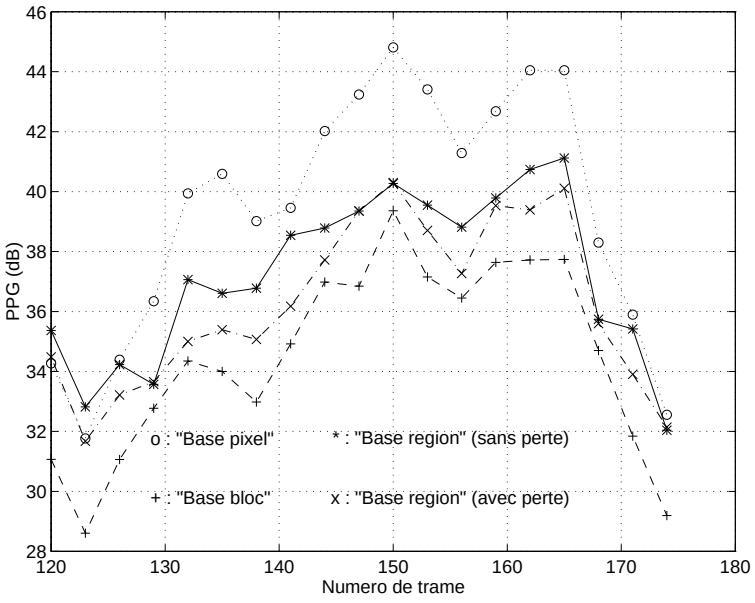


FIG. 5.4 - "*Miss America*" : comparaison des mesures de PPG pour les méthodes "basée bloc", "basée-région" avec une transmission sans perte des cartes de segmentation, "basée-région" avec une transmission avec perte des cartes de segmentation et "basée pixel".

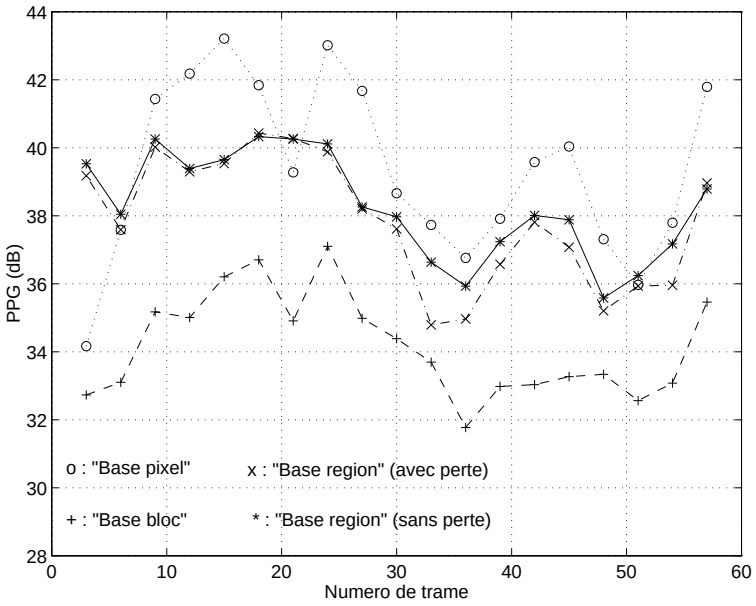


"*Carphone*" trames 3–57

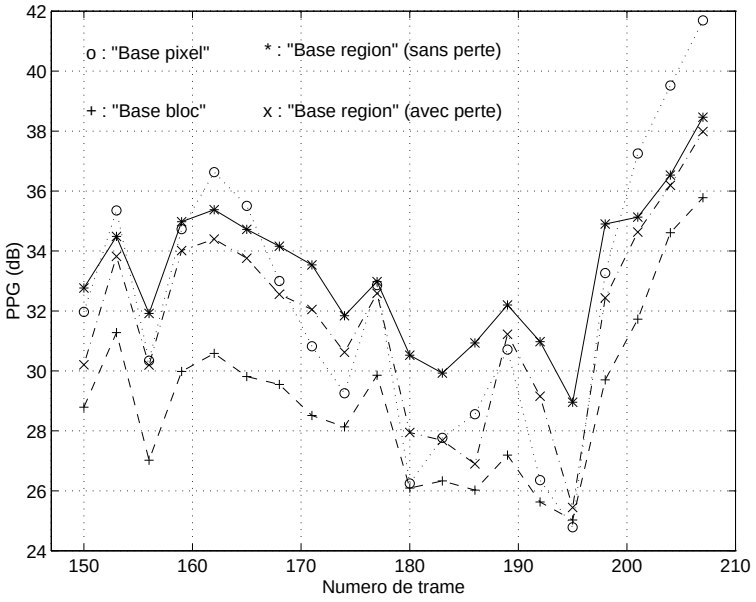


"*Carphone*" trames 120–177

FIG. 5.5 - "*Carphone*" : comparaison des mesures de PPG pour les méthodes "basée bloc", "basée-région" avec une transmission sans perte des cartes de segmentation, "basée-région" avec une transmission avec perte des cartes de segmentation et "basée pixel".



"*Foreman*" trames 3–57



"*Foreman*" 150–207

FIG. 5.6 - "*Foreman*" : comparaison objective des méthodes "basée bloc", "basée-région" avec une transmission sans perte des cartes de segmentation, "basée-région" avec une transmission avec perte des cartes de segmentation et "basée pixel".

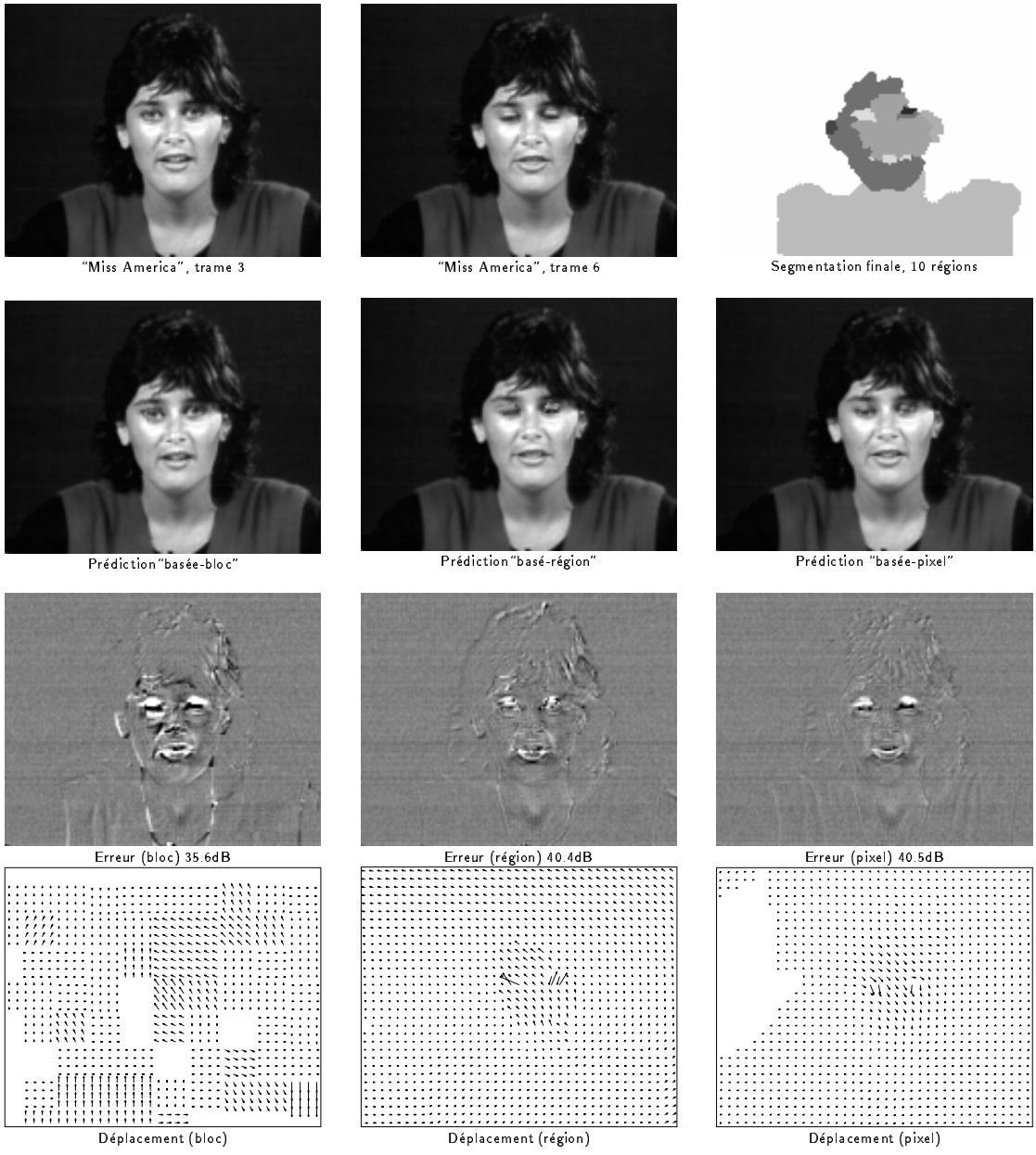


FIG. 5.7 - "*Miss America*" trame 6: comparaison subjective de trois méthodes "basée bloc", "basée région" et "basée pixel": images originales; segmentation finale; images prédites, erreurs et champs de déplacement pour les trois méthodes.

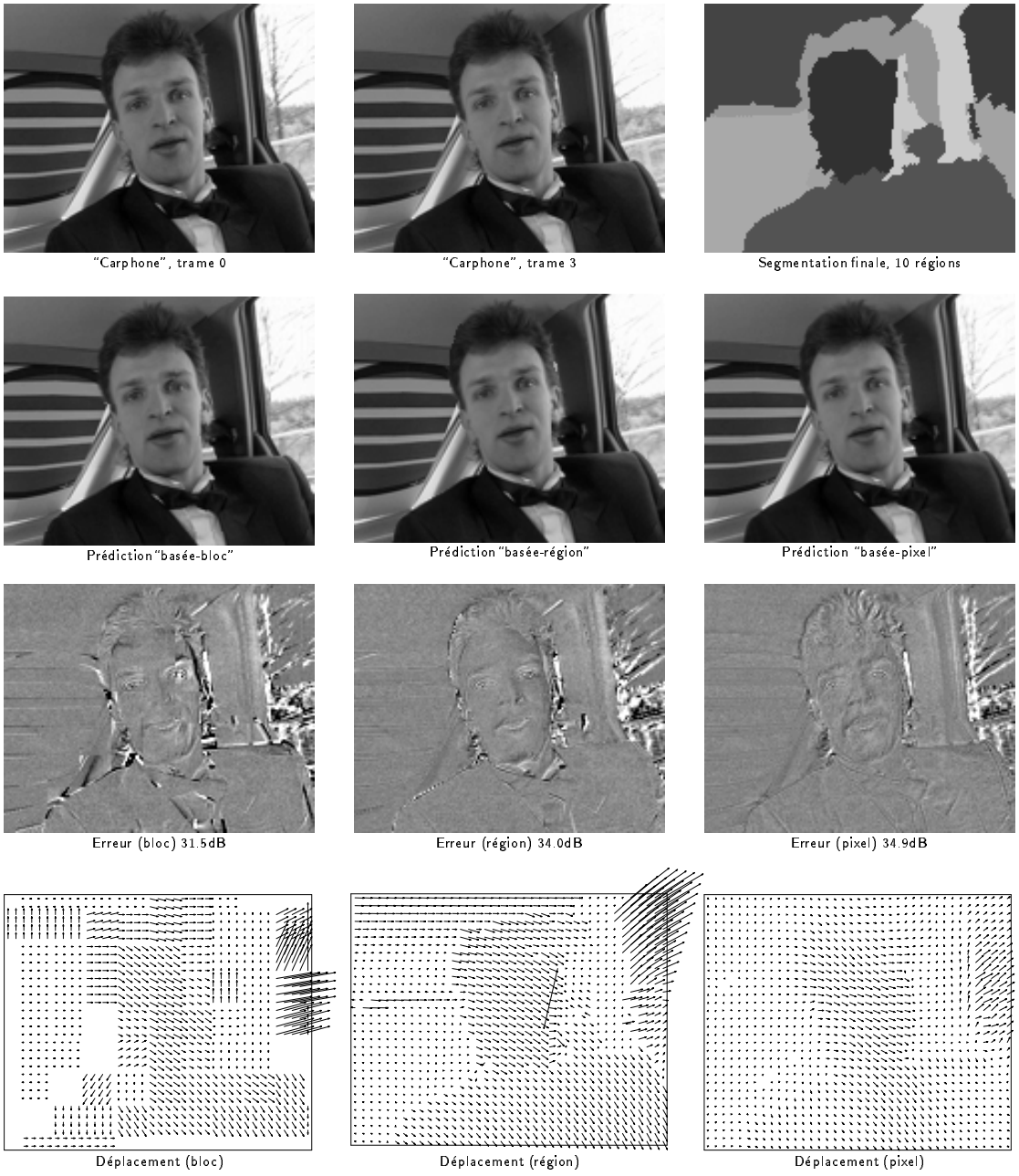


FIG. 5.8 - "*Carphone*" trame 3: comparaison subjective de trois méthodes "basée bloc", "basée région" et "basée pixel": images originales; segmentation finale; images prédites, erreurs et champs de déplacement pour les trois méthodes.

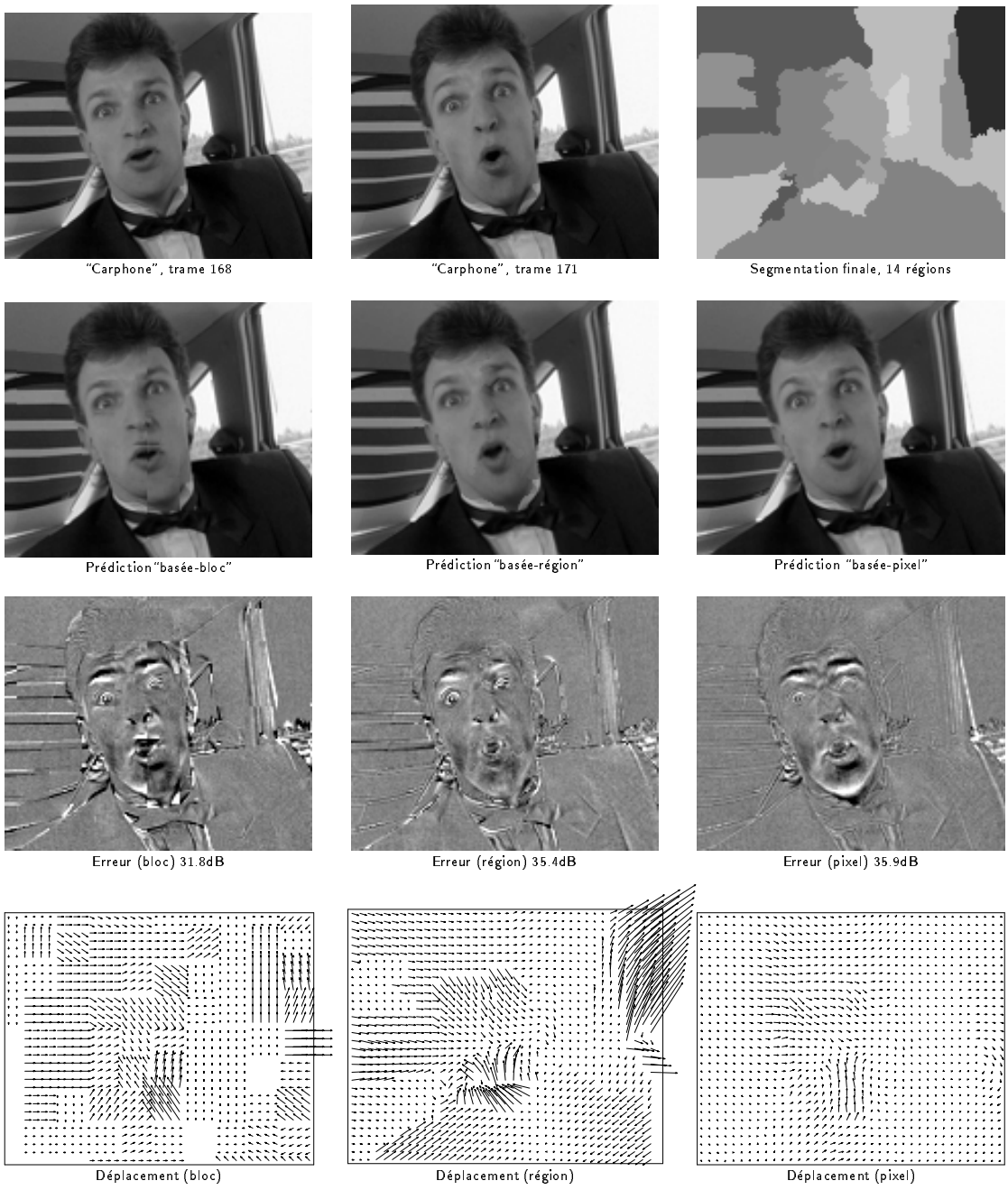


FIG. 5.9 - "*Carphone*" trame 171: comparaison subjective de trois méthodes "basée bloc", "basée région" et "basée pixel"; images originales; segmentation finale; images prédites, erreurs et champs de déplacement pour les trois méthodes.

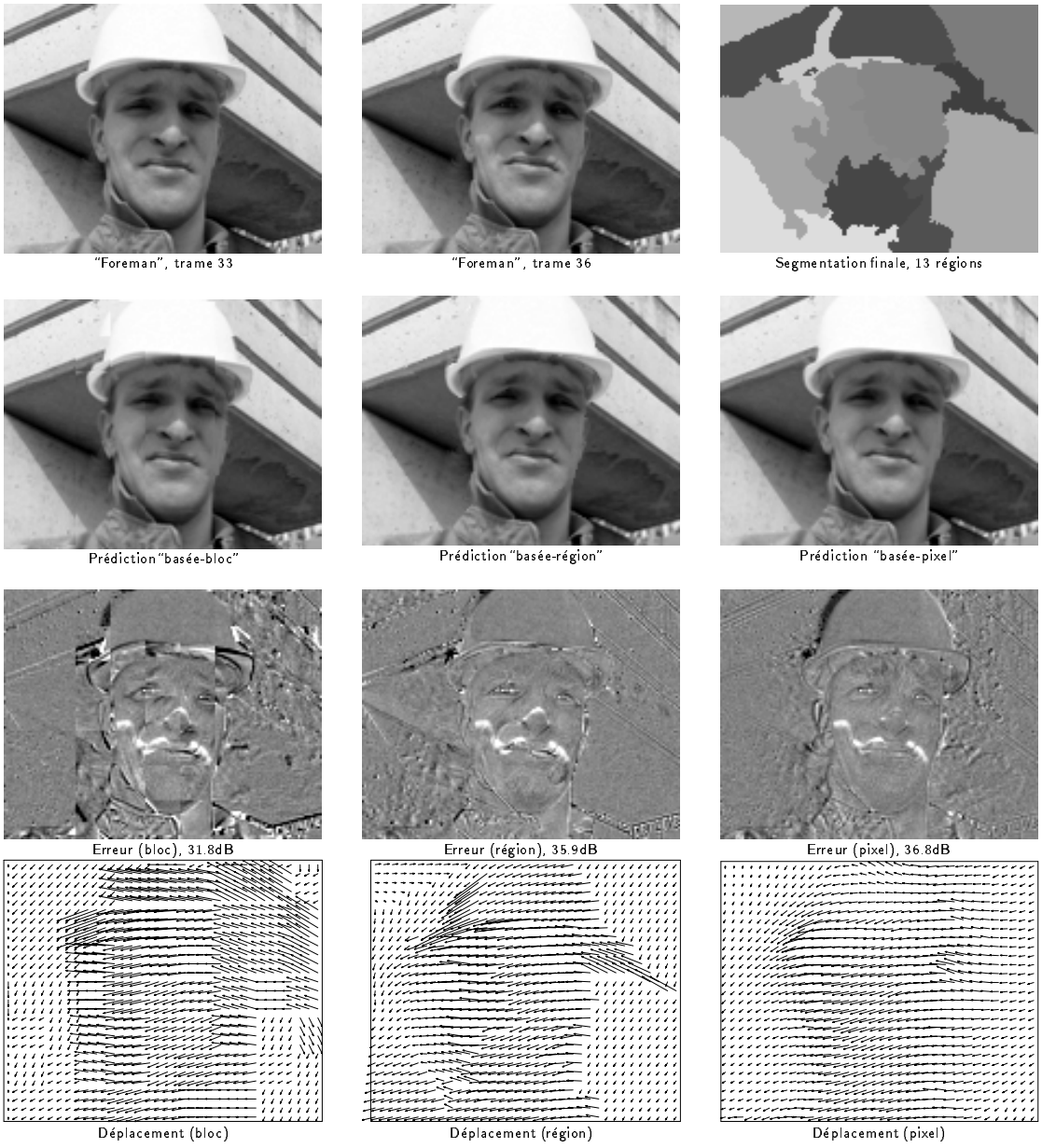


FIG. 5.10 - "*Foreman*" trame 36: comparaison subjective de trois méthodes "basée bloc", "basée région" et "basée pixel"; images originales; segmentation finale; images prédites, erreurs et champs de déplacement pour les trois méthodes.

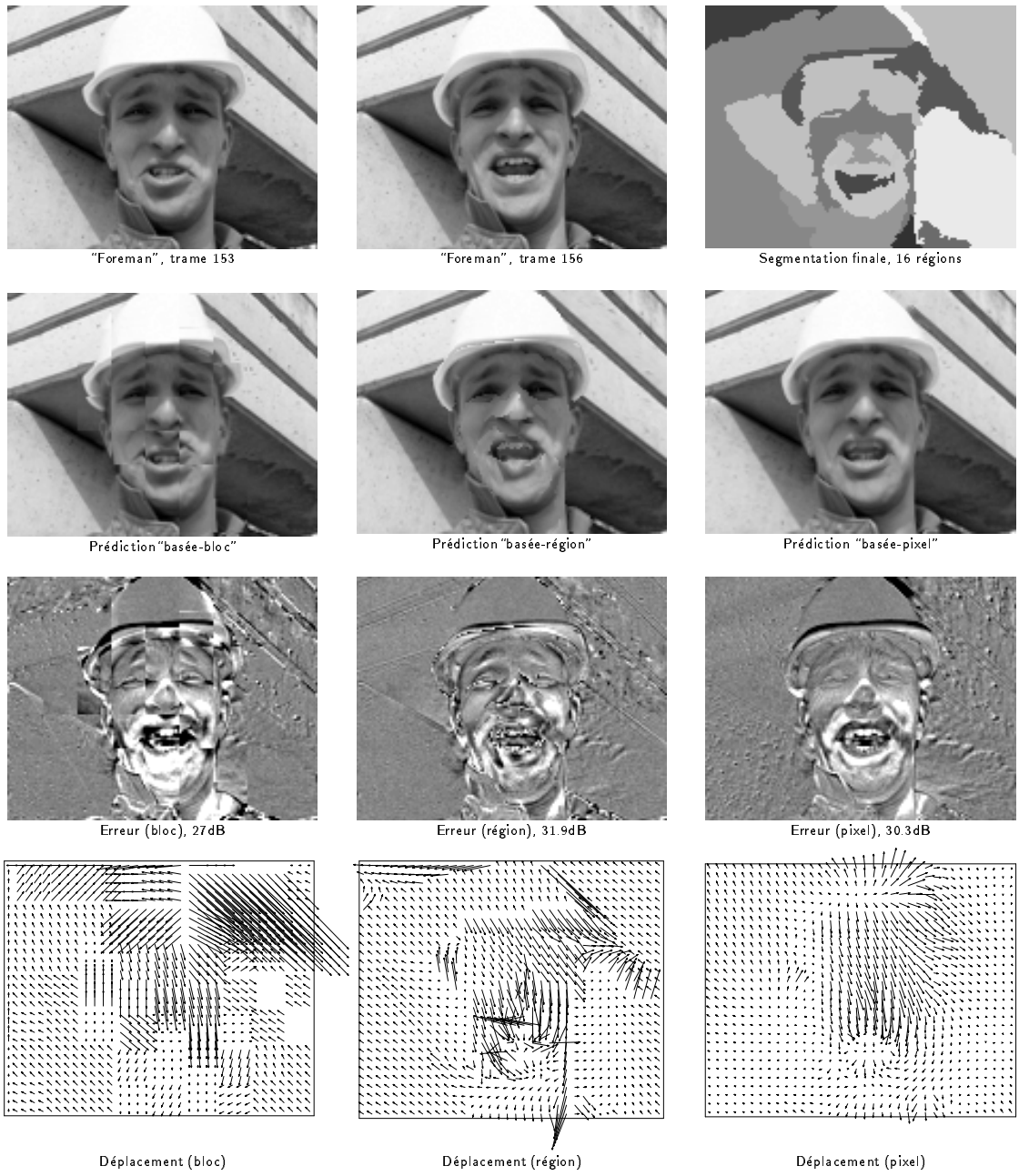


FIG. 5.11 - "*Foreman*" trame 156: comparaison subjective de trois méthodes "basée bloc", "basée région" et "basée pixel"; images originales; segmentation finale; images prédites, erreurs et champs de déplacement pour les trois méthodes.

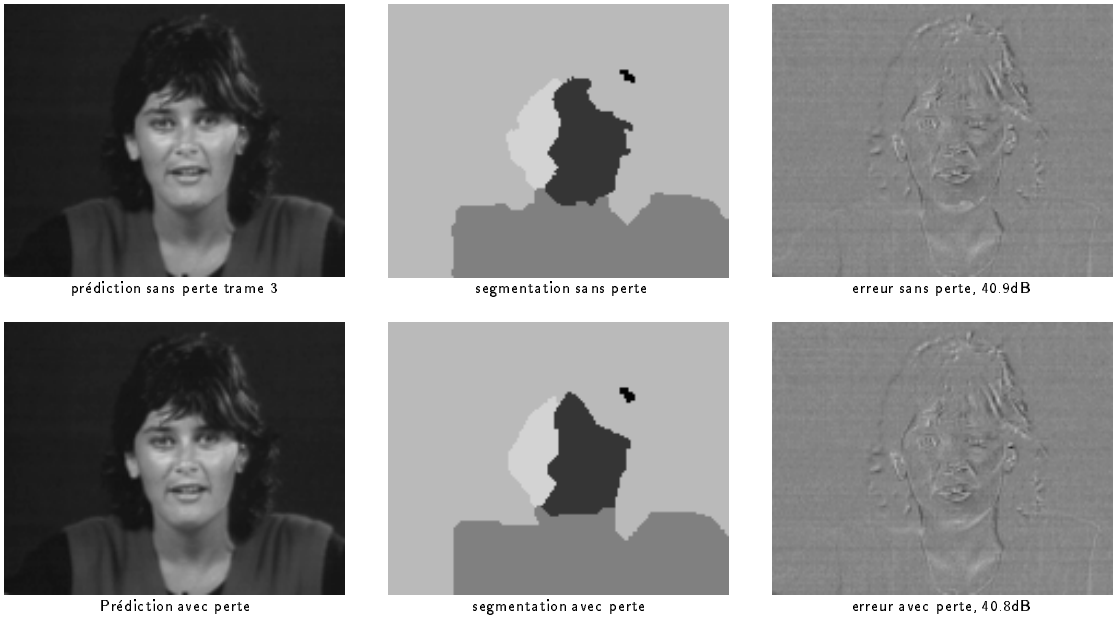


FIG. 5.12 - "*Miss America*" trame 3: comparaison de qualité pour le codage sans perte et avec perte des contours.

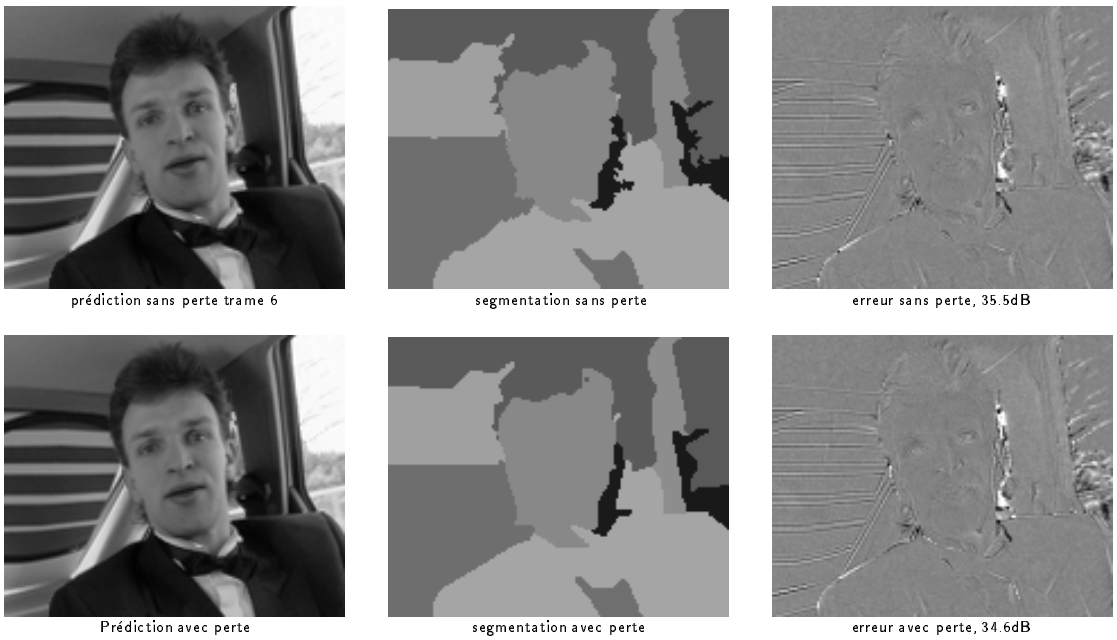


FIG. 5.13 - "*Carphone*" trame 6: comparaison de qualité pour le codage sans perte et avec perte des contours.

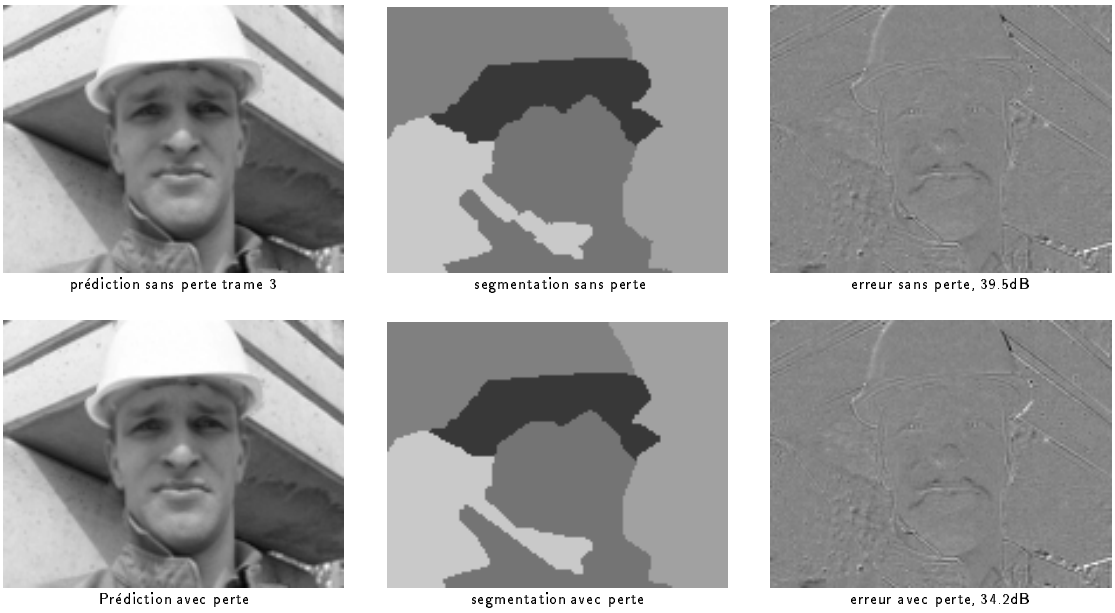


FIG. 5.14 - "*Foreman*" trame 3: comparaison de qualité pour le codage sans perte et avec perte de contours.

Chapitre 6

Conclusion

Ce mémoire décrit une méthode d'estimation et de segmentation du mouvement, susceptible d'être utilisée dans un schéma de codage à très bas débit de séquences d'images numériques par compensation de mouvement, dans lequel on souhaite transmettre l'information de mouvement sous la forme de régions et paramètres de mouvement.

Le principe général qui a orienté et guidé cette étude découle d'une observation déjà utilisée dans d'autres études [HKLM88, KD89]: les contours d'intensité coïncident presque toujours avec les contours de mouvement; conséquemment, les régions d'intensité sont des sous-objets des objets en mouvement. Ceci nous a conduit à concevoir et implanter un algorithme qui exploite les cartes de segmentation basées sur l'intensité, pour résoudre le problème d'estimation et de segmentation du mouvement en trois étapes successives:

1. estimer le mouvement des régions d'intensités;

2. fusionner les régions adjacentes ayant des mouvements similaires;

3. ajuster les contours d'intensité afin qu'ils soient consistants au sens du mouvement.

Dans la première étape, nous estimons le mouvement des régions d'intensité fournies par la méthode MDL, en utilisant le modèle de mouvement affine dans la minimisation d'une fonction, qui mesure l'erreur quadratique entre une région et sa prédiction compensée par le mouvement. La méthode Gauss-Newton a été adoptée pour résoudre ce problème d'optimisation.

L'étape de fusion a pour but principal de simplifier la carte de segmentation spatiale afin qu'elle soit la plus simple possible au sens du mouvement. D'autre part, la fusion permet d'améliorer l'estimation des paramètres de mouvement, car plus la taille des régions est grande, plus elle procure des données à la minimisation. Pour réaliser cette étape, nous avons proposé un modèle qui minimise à la fois l'erreur de prédiction compensée par le mouvement et le nombre de régions contenues dans l'image. Une méthode heuristique basée sur une représentation des régions par "graphe des voisins" a été introduite pour résoudre le problème d'optimisation; l'idée consiste à considérer chaque paire de régions voisines, et à les fusionner si la région résultante donne un gain en erreur de prédiction supérieur à un seuil donné.

Enfin, dans la dernière étape, nous remettons en cause l'hypothèse de coïncidence des contours d'intensité et de mouvement, puisqu'en pratique, il peut y avoir un petit écart entre les deux types de contour dû à la discrétisation de l'image ou aux ombrages des objets sur leurs voisins. Ceci nous a conduit à ajuster les contours d'intensité afin qu'ils soient consistants au sens du mouvement. Pour cela, nous avons proposé de minimiser, pour chaque pixel de contour, une fonction d'énergie composée de deux termes. Le premier terme mesure l'erreur de prédiction compensée par le mouvement. Le deuxième terme décrit la complexité du contour et est inspiré des modèles MRF, très utilisés en estimation du mouvement. Le problème d'optimisation qui en résulte a été résolu en utilisant la méthode de relaxation déterministe Jacobi et par une recherche exhaustive de l'étiquette de région donnant la plus petite énergie.

De nombreuses simulations ont été effectuées sur des séquences d'images typiques

de la vidéo-conférence et du visiophone afin d'évaluer la qualité des prédictions et de valider ainsi l'approche proposée. Les résultats obtenus sont très favorables. Nous avons pu observer une amélioration très nette des mouvement estimés et des images prédites après chacune des étapes de l'algorithme; les cartes de segmentation finales correspondent raisonnablement bien, autant à la forme qu'au nombre des objets réels en mouvement de la scène. Par ailleurs, la comparaison des résultats de la méthode proposée avec ceux des méthodes d'appariement de blocs et "basée pixel" ont montré que la méthode "basée région" a un gain relatif en terme de "peak prediction gain" (PPG) de 3 à 4dB par rapport à la méthode d'appariement de blocs, tandis que par rapport à la méthode "basée pixel", elle est inférieure d'environ 2dB pour une qualité comparable des mouvements estimés. Malgré une plus faible performance en terme de PPG, les comparaisons subjectives réalisées par visualisation des images prédites sur l'écran du système *Viewstore* ont montré que la qualité visuelle donnée par la méthode "basée région" est équivalente à celle de la méthode "basée pixel" et dépasse de loin celle de la méthode d'appariement de blocs. Cette caractéristique, qui constitue certainement l'originalité de notre méthode, s'explique par le fait que la plupart des erreurs se trouvent à proximité des contours d'intensité, où elles sont moins visibles grâce à l'effet de masque du système de vision humain.

Cette étude a montré l'intérêt d'une utilisation judicieuse des cartes de segmentation spatiale pour résoudre le problème d'estimation et de segmentation du mouvement dans des séquences d'images. La méthode proposée a permis d'obtenir une bonne qualité des images prédites qui ne souffrent pas de défauts des méthodes traditionnelles tels l'effet de bloc ou le "mosquito noise". Elle a permis d'obtenir aussi une description assez compacte de l'information du mouvement en régions avec leurs paramètres de mouvement. Nous croyons que notre algorithme représente une approche très prometteuse. Il reste cependant à démontrer que le codage du mouvement (régions et ses paramètres) est possible à très bas débit.

Travaux futurs

Il reste bon nombre de questions qui n'ont pas été abordées ou qui n'ont été traitées que partiellement dans cette étude, notamment les problèmes concernant le codage de l'information de mouvement. Nous achevons cette étude en donnant quelques directions de recherche que les travaux futurs pourront exploiter, en vue d'intégrer la méthode proposée dans un schéma de codage vidéo complet ou tout simplement pour l'améliorer.

- *Quantification des paramètres de mouvement*

  Pour leur transmission, les paramètres de mouvement doivent être quantifiés. Il faut donc choisir la méthode de quantification et le nombre de bits à utiliser afin d'obtenir un meilleur compromis entre la qualité d'image décodée et le débit de transmission.

- *Suivi temporel de la segmentation*

  Jusqu'à présent, la segmentation a été effectuée indépendamment d'une image à l'autre. Cette façon de faire est la plus facile, mais elle n'est pas judicieuse en codage, car dépendant du mouvement des objets de la scène, les cartes de segmentation des images dans une séquence peuvent être très différentes même si les objets physiques sont les mêmes (ou une même région est étiquetée par différentes étiquettes au fil des images), ce qui rend impossible le codage prédictif (compensé par le mouvement) des contours ou des paramètres de mouvement. Or, ce type de codage est très souhaitable à très bas débit puisqu'il permet de réduire le volume d'information de mouvement. Il faut donc introduire un lien temporel permettant de suivre la trajectoire des objets au fil des images. Ce problème a été déjà traité dans [GLB94]. Mais dans le le cadre de notre algorithme, nous pensons qu'on pourrait explorer les possibilités suivantes:

  – Introduction d'un modèle de trajectoire de mouvement dès la première

étape (estimation du mouvement).

- Si une sur-segmentation était tolérable, alors on pourrait diviser les régions concaves en petites régions convexes. De cette façon, le centre de gravité des régions pourrait être utilisé dans un suivi temporel des régions, car la position du centre de gravité d'un objet ne varie que très peu d'une trame à l'autre.

- *Contrôle du nombre de régions*

  Dans l'étape de fusion, λ_1 joue le rôle de seuil de fusion. On peut agir sur ce paramètre pour contrôler le nombre de régions dans l'image ou le débit de transmission.

- *Amélioration de l'étape de fusion*

  La méthode heuristique que nous avons proposée pour résoudre le problème d'optimisation lors de cette étape ne garantit pas l'obtention d'une carte de segmentation optimale. De plus, puisque le seuil de fusion a été fixé intuitivement par essaie-erreur, il est toujours possible que deux régions de mouvement différent soient fusionnées par erreur. Cette situation doit être évitée à tout prix, car elle est difficile à corriger par l'étape d'ajustement des contours (voire quasiment impossible si la taille des régions en erreur est grande). Il est donc souhaitable d'utiliser une meilleur méthode d'optimisation ou de modéliser le seuil de fusion en considérant la taille ou d'autres propriétés statistiques des régions.

- *Utilisation de la couleur*

  Tout au long de cette étude, seule la luminance a été utilisée dans l'estimation et dans la segmentation du mouvement. Nous pensons que l'utilisation des composantes couleurs dans la segmentation permettrait d'obtenir des régions plus consistantes, car la redondance chromatique est souvent très forte à l'intérieur des objets de la scène mais très faible entre les objets de mouvement différent.

Références

[Ana84] Anandan (P.). – Computing dense displacement fields with confidence measures in scenes containing occlusion. – In *Proc. SPIE Intelligent Robots and Computer Vision*, pp. 184–194, 1984.

[Bes86] Besag (J.). – On the statistical analysis of dirty pictures. *Royal Statistical Society*, vol. 48, serie B, No. 3, 1986, pp. 259–302.

[BPT88] Bertero (M.), Poggio (T.) et Torre (V.). – Ill-posed problems in early vision. *Proc. IEEE*, vol. 76, n° 8, août 1988, pp. 869–889.

[CCI90] CCITT. – *Recommendation H.261: Video codec for audiovisual services at $p \times 64$ kbits/s*, 1990.

[Cha95] Chartier (J. B.). – *Représentation compacte des cartes de segmentation dans les séquences d'images: étude comparative.* – Mémoire de maîtrise, INRS-Télécommunications, mars 1995.

[CK95] Chahine (M.) et Konrad (J.). – Motion-compensated interpolation using trajectories with acceleration. – In *Proc. IS&T/SPIE Symp. Electronic Imaging Science and Technology, Digital Video Compression: Algorithms and Technologies 1995*, pp. 152–163, février 1995.

[CST94] Chang (M.), Sezan (M. I.) et Tekalp (A. M.). – An algorithm for simultanous motion estimation and scene segmentation. – In *Proc. IEEE Int. Conf. Acoustics Speech Signal Processing*, pp. V.221–V.224, avril 1994.

[DB91] Driessen (J. N.) et Biemond (J.). – Motion field estimation for complex scenes. – In *Proc. SPIE Visual Communications and Image Process.*, pp. 511–521, 1991.

[DD92] Depommier (R.) et Dubois (E.). – Motion estimation with detection of occlusion areas. – In *Proc. IEEE Int. Conf. Acoustics Speech Signal Processing*, pp. III.269–III.272, mars 1992.

[Die91] Diehl (N.). – Object-oriented motion estimation and segmentation in image sequences. *Signal Process., Image Commun.*, vol. 3, n° 1, février 1991, pp. 23–56.

[DK93] Dubois (E.) et Konrad (J.). – Estimation of 2-D motion fields from image sequences with application to motion-compensated processing. *In: Motion Analysis and Image Sequence Processing*, éd. par Sezan (M.I.) et Lagendijk (R.L.), chap. 3, pp. 53–87. – Kluwer Academic Publishers, 1993.

[DMMN94] Dufaux (F.), Moccagatta (I.), Moscheni (F.) et Nicolas (H.). – Vector quantization-based motion field segmentation under the entropy criterion. *J. Vis. Commun. Image Represent.*, vol. 5, n° 4, décembre 1994, pp. 356–369.

[GG84] Geman (S.) et Geman (D.). – Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-6, n° 6, novembre 1984, pp. 721–741.

[GLB94] Garcia-Garduño (V.), Labit (C.) et Bonnaud (L.). – Temporal linking of motion-based segmentation for object-oriented image sequence coding. – In *Signal Process. VII: Theories and Applications (Proc. Seventh European Signal Process. Conf.)*, pp. 147–150, septembre 1994.

[GMW81] Gill (P.E.), Murray (W.) et Wright (M. H.). – *Pratical Optimisation*, pp. 133–154. – Academic press, 1981.

[GO93] Golub (G.) et Ortega (J. M.). – *Scientific Computing, An Introduction with Parallel Computing*, chap. 8. – Academic press, 1993.

[HB90] Heitz (F.) et Bouthemy (P.). – Multimodal motion estimation and segmentation using Markov random fields. – In *Proc. IEEE Int. Conf. Pattern Recognition*, pp. 378–383, juin 1990.

[HKLM88] Hutchinson (J.), Koch (Ch.), Luo (J.) et Mead (C.). – Computing motion using analog and binary resistive networks. *Computer*, vol. 21, n° 3, mars 1988, pp. 52–63.

[HS81] Horn (B.K.P.) et Schunck (B.G.). – Determining optical flow. *Artificial Intelligence*, vol. 17, 1981, pp. 185–203.

[Jai89] Jain (A. K.). – *Fundamentals of Digital Image Processing*, chap. 3. – Prentice-Hall, 1989.

[JW87] Jacobson (L.) et Wechsler (H.). – Derivation of optical flow using a spatiotemporal-frequency approach. *Comput. Vis. Graph. Image Process.*, vol. 38, 1987, pp. 29–65.

[KD88] Konrad (J.) et Dubois (E.). – Estimation of image motion fields: Bayesian formulation and stochastic solution. – In *Proc. IEEE Int. Conf. Acoustics Speech Signal Processing*, pp. 1072–1075, avril 1988.

[KD89] Konrad (J.) et Dubois (E.). – Bayesian estimation of discontinuous motion in images using simulated annealing. – In *Proc. Conf. Vision Interface VI'89*, pp. 51–60, juin 1989.

[KD91] Konrad (J.) et Dubois (E.). – Comparison of stochastic and deterministic solution methods in Bayesian estimation of 2D motion. *Image Vis. Comput.*, vol. 9, n° 4, août 1991, pp. 215–228.

[KD92] Konrad (J.) et Dubois (E.). – Bayesian estimation of motion vector fields. *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-14, septembre 1992, pp. 910–927.

[Key81] Keys (R.G.). – Cubic convolution interpolation for digital image processing. *IEEE Trans. Acoust. Speech Signal Process.*, vol. ASSP-29, n° 6, décembre 1981, pp. 1153–1160.

[Lec89] Leclerc (Y.). – Constructing simple stable descriptions for image partitioning. *Intern. J. Comput. Vis.*, vol. 3, 1989, pp. 73–102.

[LeG91] LeGall (D.). – MPEG: A video compression standard for multimedia applications. *Communications ACM*, vol. 34, n° 4, avril 1991, pp. 46–58.

[LN91] Labit (C.) et Nicolas (H.). – Compact motion representation based on global features for semantic image sequence coding. – In *Proc. SPIE Visual Communications and Image Process.*, pp. 697–708, 1991.

[MB87] Murray (D.W.) et Buxton (B.F.). – Scene segmentation from visual motion using global optimization. *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-9, n° 2, mars 1987, pp. 220–228.

[MHO89] Musmann (H. G.), Hötter (M.) et Ostermann (J.). – Object-oriented analysis-synthesis coding of moving images. *Signal Process., Image Commun.*, vol. 1, octobre 1989, pp. 117–138.

[Nic92] Nicolas (H.). – *Hiérarchie de modèles de mouvement et méthodes d'estimation associées. Application au codage de séquences d'images.* – Thèse de PhD, l'Université de Rennes I, septembre 1992.

[RC94] Reader (C.) et Chiariglione (L.). – MPEG-4 Proposal Package Description. *ISO/IEC JTC1/SC29/WG11*, Nov. 1994.

[San94] Sanson (H.). – Region-based motion analysis for video coding at low bitrates. *MPEG-4, Paris*, Mars 1994.

[Sch93] Schaphorst (R.). – Report of the ITU-TS WP 15/1 Special Rapporteur for Very-Low Bitrate Visual Telephony. *ISO/IEC JTC1/SC29/WG11*, Sept. 1993.

[SH93] Stiller (C.) et Hürtgen (B.). – Combined displacement estimation and segmentation in image sequences. – In *Int. Symp. on Fibre Optic Networks and Video Comm. EUROPTO*, pp. 276–287, avril 1993.

[SS92] Stiller (C.) et Suntrup (R.). – Parametric object-motion estimation. – In *Proc. Int. Symp. on Information Theory and its Applications*, pp. 633–637, novembre 1992.

[Sti94] Stiller (C.). – Object oriented video coding employing dense motion fields. – In *Proc. IEEE Int. Conf. Acoustics Speech Signal Processing*, pp. V.273–V.276, avril 1994.

[TH81] Tsai (R.Y.) et Huang (T.S.). – Estimating three-dimensional motion parameters of a rigid planar path. *IEEE Trans. Acoust. Speech Signal Process.*, vol. ASSP-29, n° 6, décembre 1981, pp. 1147–1152.

[Tho87] Thomas (G.A.). – *Television motion measurement for DATV and other applications.* – Rapport technique n° 11, BBC Reasearch Dept., septembre 1987.