

## Estimation robuste multiéchelle de modèles paramétrés de mouvement sur des scènes complexes

---

### *Robust Multiresolution Estimation of Parametric Motion Models in Complex Image Sequences*

par Jean-Marc ODOBEZ, Patrick BOUTHEMY

IRISA/INRIA, Campus Universitaire de Beaulieu, 35042 Rennes Cedex.  
Adresse électronique : bouthemy@irisa.fr. Tél. 99 84 71 00

#### Résumé

Nous présentons dans cet article une méthode d'estimation de modèles paramétrés de mouvement. L'intérêt de formuler un problème d'analyse de mouvement par l'identification de tels modèles est double. Il s'agit d'une part d'une représentation compacte qui s'avère adéquate et pertinente dans des contextes différents et nombreux (mesure, segmentation, suivi ou caractérisation du mouvement). D'autre part, une estimation peu coûteuse en temps calcul peut en être obtenue. Le point crucial par contre est d'en établir une estimation fiable et précise, et de gérer correctement la présence de plusieurs mouvements dans l'image. Pour répondre de façon bien fondée et efficace à ces problèmes, nous proposons deux versions d'un estimateur robuste, du type  $M$ -estimateur, défini dans un schéma multirésolution. Cette technique a permis d'obtenir des résultats tout-a-fait satisfaisants sur des images représentant des scènes complexes.

**Mots clés :** Séquence d'images, mouvement, modèles paramétrés, estimateur robuste, mesure multiéchelle, segmentation.

#### Abstract

*This paper describes a parametric motion model estimation algorithm. Motivations for the use of such models are on the one hand their compactness, which has proved to be efficient in numerous contexts such as estimation, segmentation, tracking and interpretation of motion, and on the other hand, the low computational cost of its estimations. However, it is important to have the best accuracy for the estimated parameters, and to take into account the problem of multiple motion. We therefore developed two  $M$ -estimator-like estimators in a multiresolution framework. Numerical results support this approach, as demonstrated by the use of these algorithms on complex sequences.*

**Key words :** Dynamic scene analysis, parametric models, motion estimation, robust estimation, multiresolution, segmentation.

---

## 1. Introduction

L'analyse du mouvement dans une séquence d'images représente un des sujets centraux en vision par ordinateur et est motivée par de nombreuses applications ([1, 7, 25]). Certaines d'entre elles (météorologie, bio-médical, ...) s'intéressent à des phénomènes physiques naturels et donc traitent des scènes comprenant des objets aux formes souvent peu structurées et aux comportements dynamiques complexes. Des situations non triviales se rencontrent aussi dans des problèmes de navigation d'un véhicule en environnement extérieur quelconque ou de télésurveillance de sites, dans le domaine des transports par exemple, où des mouvements de personnes doivent être pris en compte.

Il est clair que dans de tels contextes, l'analyse du mouvement présent dans la séquence d'images acquises est une tâche délicate requérant une approche bien fondée et efficace. Un premier élément de réponse réside dans la formulation de ce problème par l'identification de modèles paramétrés 2D du champ des vitesses, en l'occurrence des modèles polynomiaux fonction des coordonnées  $(x, y)$  des points dans l'image, soit des modèles affines (polynômes du premier degré en  $x$  et  $y$ ), ou quadratiques, voire des fonctions rationnelles (transformées homographiques). Ce choix s'est déjà avéré judicieux aussi bien pour segmenter l'image en régions homogènes au sens du mouvement apparent, [8, 31], pour extraire et coder l'information temporelle dans des schémas de codage à compensation de mouvement, [14, 27], pour fournir une mesure du mouvement apparent, [32], pour suivre des objets en mouvement le long de la séquence, [24], ou pour caractériser le mouvement 3D des objets dans

la scène, que ce soit qualitativement, [8], ou quantitativement, [26]. Soulignons qu'une telle approche par modèle a aussi été appliquée avec succès sur un problème voisin, à savoir pour la phase de mise en correspondance entre deux images dans un schéma de stéréovision, [2]. L'intérêt d'une telle approche est double. Il s'agit d'une part d'une représentation compacte et qui s'est avérée adéquate dans des contextes différents et nombreux comme l'ont montré les études citées précédemment. D'autre part, une estimation peu coûteuse en temps calcul peut en être obtenue.

Le point crucial par contre est d'en obtenir une estimation fiable et précise. Nous proposons dans cet article une méthode d'estimation robuste multiéchelle de modèles de mouvement paramétrés.

Il est désormais acquis en analyse du mouvement dans une séquence d'images par des approches différentielles, c.à.d. utilisant les gradients spatio-temporels de l'intensité, que la mesure du mouvement est grandement améliorée si l'on adopte une technique d'estimation multirésolution. On ne considère pas seulement la séquence d'images à sa résolution d'acquisition, mais on construit à partir de chaque image une pyramide d'images successivement filtrées et sous-échantillonnées. Ceci a été principalement étudié et validé pour la détermination de champs denses de vitesses apparentes, [3, 12, 20]. On obtient ainsi des mesures correctes même en présence de mouvements de grande amplitude, ou de distribution d'information de gradients d'intensité irrégulière dans l'image. Plus récemment, des techniques d'estimation aux moindres-carrés multiéchelle de modèles de mouvement ont été proposées dans [4] et [23]. Cependant cela ne suffit pas, car pour être opérationnel, ce calcul multirésolution doit être effectué sur des zones de taille suffisante. Or, les modèles de mouvement utilisés ne peuvent rendre compte que d'un seul mouvement, et dans les scènes complexes qui nous intéressent, plusieurs mouvements peuvent alors être simultanément présents. L'estimation globale sur toute l'image (ou sur des blocs de taille convenable) d'un modèle de mouvement s'en trouve très vite perturbée. Dans [4], il est postulé que le mouvement global perçu résulte du déplacement du capteur, les projections des objets mobiles dans la scène ne couvrant qu'une portion très faible de l'image. Une latitude sensiblement plus grande sur ce point est autorisée dans [5] pour des situations de transparence, mais ne mettant en œuvre que des modèles constants. Dans [23], l'estimation est effectuée par région, ces régions étant issues d'une phase préalable de segmentation au sens du mouvement garantissant la présence d'un mouvement unique dans chacune des régions considérées.

La classe des estimateurs robustes, [17, 28], récemment popularisée dans le domaine de l'analyse d'image, [22, 6, 19], offre une voie d'investigation intéressante. Toutefois, la définition d'un estimateur robuste bien adapté au problème qui nous intéresse, suppose que l'on tienne compte de trois aspects importants : les données considérées sont bruitées, les modèles introduits sont inexacts, le coût des calculs doit être le plus faible possible. Notons qu'une première tentative intéressante a été menée dans [11]. Cependant cette méthode ne prend en compte qu'un modèle de translation 3D et suppose la profondeur constante, soit en fait un

modèle de mouvement 2D du premier ordre mais réduit à trois paramètres; de plus elle n'opère qu'en mono-résolution. Notons également que récemment des techniques d'estimation robuste ont été exploitées pour la mesure de champs de vitesse 2D dans [6].

Le plan de l'article est le suivant. Dans la section 2, nous définissons le type de modèles de mouvement 2D considéré. La section 3 contient la technique d'estimation aux moindres-carrés multiéchelle à laquelle sera comparée la technique originale que nous avons développée. La section 4, après quelques rappels sur les estimateurs robustes, décrit notre méthode d'estimation de modèles paramétrés de mouvement à l'aide d'un estimateur robuste multiéchelle (en fait nous en présenterons deux variantes). Plusieurs résultats sur des exemples synthétiques et réels seront montrés et commentés dans la section 5.

## 2. Modèle de mouvement

Nous considérons la classe des modèles de mouvement 2D polynomiaux. En utilisant des notations matricielles, ces modèles peuvent s'écrire sous la forme générale suivante (linéaire en les paramètres  $(a_j)$  du polynôme) :

$$V_A(X_i) = \begin{bmatrix} u_A(X_i) \\ v_A(X_i) \end{bmatrix} = B(X_i)A \quad (1)$$

où  $A^t = (a_j)$ ,  $X_i = (x_i, y_i)$  désigne un point de l'image (ou du bloc considéré),  $V_A(X_i)$  le vecteur vitesse au point  $X_i$  associé au modèle paramétré par  $A$ .  $B$  est une matrice dont la forme dépend du modèle choisi, mais dont les coefficients ne dépendent que des coordonnées du point considéré.

Dans notre approche, tout type de modèles de cette classe peut être envisagé. On peut utiliser un simple modèle constant,  $V_A(X_i) = (a_0, a_1)$ , aussi bien que des modèles affines et quadratiques, complets ou partiels. Nous nous intéresserons en fait dans cet article plus particulièrement au modèle affine complet défini par :

$$\begin{cases} u_A(X_i) = a_1 + a_2 x_i + a_3 y_i \\ v_A(X_i) = a_4 + a_5 x_i + a_6 y_i \end{cases} \quad (2)$$

On a dans ce cas :

$$B_i = B(X_i) = \begin{bmatrix} 1 & x_i & y_i & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x_i & y_i \end{bmatrix}$$

Ce modèle représente en fait un bon compromis entre représentativité et complexité. Il peut rendre compte d'une large gamme de mouvement 2D (translation, rotation, similitude plane, déformation) et même si le projeté 2D d'un mouvement rigide 3D est au moins quadratique, ce modèle contient quand même l'essentiel de l'information, [8, 26].

### 3. Estimation multirésolution aux moindres-carrés

Nous allons tout d'abord décrire dans ses grandes lignes la technique d'estimation moindres-carrés multirésolution introduite dans [23, 4], à laquelle nous avons ajouté l'estimation d'un paramètre supplémentaire  $\xi$  qui permet de rendre compte (dans une certaine limite toutefois) de variations d'illumination. Elle est basée sur l'algorithme de Gauss-Newton, qui conduit à une estimation incrémentale du modèle de mouvement. Pour prendre en compte les grands déplacements, une stratégie descendante classique ("coarse-to-fine") à travers les niveaux de résolution d'une pyramide gaussienne des images est utilisée.

#### 3.1. CRITÈRE DE MINIMISATION

Pour chaque point  $X_i$ , on peut écrire sous forme vectorielle :

$$\frac{dI}{dt}(X_i, t) = \vec{V}(X_i, t) \cdot \vec{\nabla}I(X_i, t) + I_t(X_i, t) \quad (3)$$

où  $\vec{V} = (\frac{dx}{dt}, \frac{dy}{dt})$  désigne le vecteur vitesse d'un point,  $\vec{\nabla}I = (I_x, I_y)$  et  $I_t$  représentent respectivement le gradient spatial de la fonction intensité  $I$  et sa dérivée temporelle partielle.  $\frac{dI}{dt}(X_i, t)$  est la dérivée totale de  $I$  par rapport au temps  $t$ , soit la variation d'intensité instantanée du point  $X_i$  sur sa trajectoire 2D. L'hypothèse d'invariance de l'intensité d'un point sur sa trajectoire, soit  $\frac{dI}{dt}(X_i, t) = 0$ , [16], conduit à l'équation bien connue de contrainte du mouvement apparent :

$$\vec{V}(X_i, t) \cdot \vec{\nabla}I(X_i, t) + I_t(X_i, t) = 0.$$

Cependant, des changements globaux d'illumination peuvent survenir, par exemple sur des scènes d'extérieur, ou bien sur des séquences d'images satellites (où la fréquence d'acquisition des images est de une image toutes les demi-heures pour Météosat par exemple), aussi bien dans le canal du visible que dans le canal infrarouge (dans ce dernier cas, le changement global est dû à des variations diurnes et interdiurnes de température d'illumination, [29]). Pour pouvoir en tenir compte, nous avons choisi d'autoriser une variation globale d'intensité sur la zone de calcul considérée, soit :

$$\frac{dI}{dt}(X_i, t) = -\xi \quad (4)$$

où  $\xi$  représente donc un paramètre supplémentaire à estimer. On introduit la variable  $r_i$  suivante (en supprimant la variable temps  $t$  pour simplifier les notations) :

$$\begin{aligned} r_i &= \frac{dI}{dt}(X_i) + \xi = \vec{V}(X_i) \cdot \vec{\nabla}I(X_i) + I_t(X_i) + \xi \\ &= I_x(X_i)u(X_i) + I_y(X_i)v(X_i) + I_t(X_i) + \xi \end{aligned} \quad (5)$$

En considérant non pas une fonction  $\vec{V}$  quelconque, mais le modèle de mouvement  $\vec{V}_A$ , on obtient finalement l'expression suivante de  $r_i$  en utilisant cette fois des notations matricielles :

$$r_i = \mathcal{X}_i \Theta - \mathcal{Y}_i \quad (6)$$

$$\text{où } \begin{cases} \Theta^t = (A^t, \xi) \\ \mathcal{Y}_i = -I_t(X_i) \quad \text{et} \\ \mathcal{X}_i = \mathcal{X}(X_i, \nabla I(X_i)) = (\nabla I(X_i))^t B_i, 1) \end{cases}$$

Cependant, l'équation  $r_i = 0$ , avec  $r_i$  donné par (5), ne traduit bien l'hypothèse (4) que lorsque l'amplitude du mouvement réel est compatible avec les fréquences spatiales de l'intensité. Plus l'image contient de hautes fréquences spatiales, plus l'amplitude de mouvement mesurable est petite. Il faut alors lisser l'image (exploiter les basses fréquences) pour appréhender les mouvements de grande amplitude. Ceci sera effectué dans l'approche multirésolution proposée plus loin. Néanmoins, lorsque les déplacements sont importants, il est préférable d'utiliser une formulation discrète de l'hypothèse  $\frac{dI}{dt}(X_i, t) = -\xi$ . Considérons deux images successives aux instants  $t$  et  $t + \delta t$ . En posant  $\delta t = 1$  pour simplifier les notations, on cherche alors à minimiser la fonction d'erreur suivante :

$$E(\Theta) = \sum_{X_i \in F} e_i^2 \quad (7)$$

où  $F$  désigne le support d'estimation considéré (par exemple toute l'image ou un bloc), et  $e_i$  désigne l'expression suivante :

$$e_i = I(X_i + B_i A, t + 1) - I(X_i, t) + \xi \quad (8)$$

Cependant, le critère (7) n'est plus linéaire vis-a-vis des paramètres à estimer. Pour minimiser  $E(\Theta)$ , nous allons faire appel à la méthode de Gauss-Newton, qui conduit à une estimation incrémentale des paramètres.

#### 3.2. ESTIMATION INCRÉMENTALE

Cette approche incrémentale s'applique aussi bien au passage d'un niveau de résolution au suivant, qu'à l'estimation des paramètres au sein d'un niveau de résolution donné. Pour minimiser  $E$ , supposons que l'on ait une estimation courante  $\hat{\Theta}_k^t = (\hat{A}_k^t, \hat{\xi}_k^t)$  de  $\Theta$ . Dans ce cas, on peut écrire :

$$\Theta = \hat{\Theta}_k + \Delta\Theta_k, \text{ soit } \begin{cases} A = \hat{A}_k + \Delta A_k \\ \xi = \hat{\xi}_k + \Delta \xi_k \end{cases}$$

et l'on a donc :

$$\delta X_i = B_i A = B_i \hat{A}_k + B_i \Delta A_k.$$

Pour chacune des expressions  $e_i$ , on effectue un développement limité au premier ordre de  $I$  à l'instant  $t + 1$  au point  $X_i + B_i \hat{A}_k$  et on obtient alors une fonction d'erreur approchée de  $E$  :

$$E_2(\Delta\Theta_k) = \sum_{X_i \in F} r_i'^2 \quad (9)$$

$$\begin{aligned} \text{avec } r'_i &= I(X_i + B_i \widehat{A}_k, t + 1) - I(X_i, t) + \widehat{\xi}_k \\ &\quad + \nabla I^t(X_i + B_i \widehat{A}_k, t + 1) B_i \Delta A_k + \Delta \xi_k \\ &= \mathcal{X}'_i \Delta \Theta_k - \mathcal{Y}'_i \end{aligned} \quad (10)$$

où ici :

$$\begin{cases} \mathcal{Y}'_i = I(X_i, t) - I(X_i + B_i \widehat{A}_k, t + 1) - \widehat{\xi}_k \\ \mathcal{X}'_i = \mathcal{X}(X_i, \nabla I(X_i + B_i \widehat{A}_k, t + 1)) \end{cases} \quad (11)$$

La minimisation de  $E_2$  par rapport à  $\Delta \Theta_k$  donne la solution évidente suivante :

$$\widehat{\Delta \Theta}_k = \left[ \sum_{X_i \in F} \mathcal{X}'_i{}^t \mathcal{X}'_i \right]^{-1} \sum_{X_i \in F} \mathcal{X}'_i{}^t \mathcal{Y}'_i \quad (12)$$

Les dérivées de la fonction intensité sont calculées selon la technique décrite dans [30], et dans les expressions de (11), les valeurs de  $I$ ,  $I_x$  et  $I_y$  aux points de coordonnées non entières sont calculées par interpolation bilinéaire.

### 3.3. STRATÉGIE DESCENDANTE COMPLÉTÉE

Ces estimations incrémentales sont itérées selon une stratégie multirésolution descendante, soit du niveau de résolution le plus grossier au plus fin. Nous utilisons une pyramide gaussienne de chaque image de  $L$  niveaux, obtenue selon le schéma de [9], à partir de la résolution initiale d'acquisition, niveau 0. Ainsi les déplacements entre deux images successives sont chaque fois divisés par deux en passant au niveau grossier suivant.

L'estimation se déroule alors de la manière suivante. Au niveau de résolution le plus grossier,  $L$ , nous ne disposons pas d'estimation initiale de  $\Theta$ . On minimise alors la fonction suivante :

$$E_1(\Theta) = \sum_{X_i \in F} r_i^2 = \sum_{X_i \in F} (\mathcal{X}_i \Theta - \mathcal{Y}_i)^2 \quad (13)$$

où les différents termes sont donnés par les relations (5) et (6). A ce niveau de résolution, les déplacements sont petits, ce qui reconditionne correctement l'équation de contrainte de mouvement. Une première estimation de  $\Theta^L$  est ainsi obtenue. Puis des raffinements sont opérés à l'aide de (12) au même niveau. Lorsque l'incrément estimé  $\widehat{\Delta \Theta}^L$  devient trop faible ou qu'un nombre prédéfini d'itérations ont été effectuées, les paramètres  $\widehat{\Theta}^L$  sont transmis au niveau directement inférieur, où le processus de raffinement reprend avec, comme estimée initiale, la valeur de  $\widehat{\Theta}^L$  projetée au niveau  $L-1$ . Ceci est répété de niveau  $l$  en niveau  $l-1$ . L'estimée finale de  $\Theta$ ,  $\widehat{\Theta}_{est}$ , est alors la valeur  $\widehat{\Theta}^0$  obtenue après la dernière itération au niveau 0, le niveau le plus fin. Cet algorithme est présenté sur la figure 1.

Dans cet algorithme, la "norme" utilisée ( $\|\widehat{\Delta \Theta}^l\|$ ) correspond en fait à une somme pondérée (en fonction de la géométrie de la région) des variations  $\Delta a_j^l$  des paramètres.  $\|\widehat{\Delta \Theta}^l\|$  est alors

```

 $\widehat{\Theta}^L \leftarrow$  Moindres-Carrés (résiduel (6))
POUR  $l =$  niveau  $L$  au niveau 0 FAIRE
  iter  $\leftarrow$  0
  FAIRE
     $\widehat{\Delta \Theta}^l \leftarrow$  Moindres-Carrés (résiduel(10))
     $\widehat{\Theta}^l \leftarrow \widehat{\Theta}^l + \widehat{\Delta \Theta}^l$  et iter  $\leftarrow$  iter + 1
  TANT QUE (iter <  $\lambda$  et  $\|\widehat{\Delta \Theta}^l\| > \frac{d}{2^l}$ )
  Si  $l \neq 0$  :  $\widehat{A}^{l-1} \leftarrow P(\widehat{A}^l)$ ,  $\widehat{\xi}^{l-1} \leftarrow \widehat{\xi}^l$ 
FIN POUR
 $\widehat{\Theta}_{est} \leftarrow \widehat{\Theta}^0$ 
    
```

Figure 1. – Algorithme moindres-carrés multirésolution (MCMR)

assimilable à un déplacement, et le critère d'arrêt des itérations teste en fait si le nouvel incrément correspond à une modification significative du champ des vitesses. Le dénominateur rend le critère homogène aux différents niveaux de la pyramide. De même, dans (12), les différents termes sont considérés au niveau auquel on se trouve :  $I$  devient  $I^l$ ,  $X_i$  devient  $X_i^l$ , la zone  $F$  devient  $F^l$ , etc... Quant à l'opérateur de projection  $P$ , il permet de transformer les paramètres de mouvement d'un niveau donné au niveau directement inférieur. Il s'agit d'un changement d'échelle. Reconsidérons le modèle des déplacements et les unités des termes  $a_j$  : les termes constants sont homogènes à des déplacements, les termes linéaires sont sans unité, (les termes quadratiques, si on les avait introduits, seraient homogènes à l'inverse d'une distance, ...). Lorsque l'on descend d'un niveau, l'unité de distance est multipliée par deux. La transformation  $P$  se résume donc en :

$$P : \begin{cases} a_{const}^{l-1} \leftarrow 2a_{const}^l \\ a_{lin}^{l-1} \leftarrow a_{lin}^l \\ a_{quad}^{l-1} \leftarrow \frac{1}{2}a_{quad}^l \end{cases} \quad (14)$$

## 4. Estimation robuste multiéchelle

En analyse statistique, l'estimation robuste a pour but de trouver les paramètres  $\Theta$  qui ajustent au mieux un modèle  $M(X_i, \Theta)$  aux observations  $y_i$ , dans le cas où des données ne correspondent pas à la statistique du modèle d'erreur choisi, c'est-à-dire dans le cas où un certain nombre de données se comportent comme des "outliers".

## 4.1. RAPPEL SUR LES ESTIMATEURS ROBUSTES

Les estimateurs robustes sont souvent caractérisés par trois indices : leur efficacité, c.à.d. leur capacité à atteindre les estimées optimales étant donné certaines hypothèses sur les distributions, leur robustesse, que l'on peut définir grossièrement comme le taux d'"outliers" maximal que peut tolérer l'estimateur, et le coût calcul. Nous présentons ci-dessous brièvement les deux types d'estimateurs robustes les plus fréquemment utilisés, [17, 28].

– M-estimateur : Le principe de cet estimateur consiste à minimiser une somme de résiduels :

$$\hat{\Theta} = \operatorname{argmin}_{\Theta} \sum_i \rho(y_i - M(\Theta, X_i), \sigma) \quad (15)$$

La fonction  $\rho$  qui intervient dans cette fonction d'erreur est appelée M-estimateur car cette estimation correspond à une estimation suivant le maximum de vraisemblance, si l'on interprète  $\rho$  comme étant l'opposé de la log-vraisemblance des observations conditionnellement au modèle.

Pour analyser la robustesse des M-estimateurs, on peut se baser sur les courbes d'influence introduites dans ([13]). Ces courbes caractérisent le biais ou l'influence que peut introduire une erreur sur l'estimation, et correspond, dans le cas d'un estimateur continu, à la dérivée  $\psi$  de cet estimateur. Si l'on considère par exemple l'estimateur des moindres carrés, on a :  $\rho(x) = x^2$ ,  $\psi(x) = 2x$ . Dans ce cas, l'influence des "outliers" augmente sans limite. Si l'on remplace la norme quadratique  $L_2$  par la norme absolue  $L_1$  :  $\rho(x) = |x|$ ,  $\psi(x) = \operatorname{sign}(x)$ , l'influence des erreurs est certes plus faible, mais le point de rupture asymptotique reste tout de même 0. Comme l'on souhaite totalement éliminer les contributions des points aberrants, l'utilisation d'un estimateur fortement "redescendant" est nécessaire. C'est le cas de l'estimateur "biweight" de Tukey (voir figure 2) dont la fonction  $\psi$  est donnée par :

$$\psi(x, C) = \begin{cases} x(C^2 - x^2)^2 & \text{si } |x| < C, \\ 0 & \text{sinon} \end{cases} \quad (16)$$

L'utilisation d'estimateurs fortement redescendants permet d'obtenir des points de rupture strictement supérieurs à 0, mais ils atteignent au maximum  $\frac{1}{p+1}$ , où  $p$  est le nombre de paramètres du modèle à estimer.

– Estimateur des moindres carrés médians : Il est obtenu par :

$$\hat{\Theta} = \operatorname{argmin}_{\Theta} \operatorname{Med}_i (y_i - M(\Theta, X_i))^2 \quad (17)$$

Son avantage principal est bien sûr sa très grande robustesse théorique, l'estimateur restant fiable jusqu'à un taux de 50% d'"outliers". Il présente plusieurs défauts, notamment :

- si le modèle est non linéaire par rapport aux paramètres, il devient difficile voire impossible de le calculer;
- si le modèle est linéaire, l'estimation requiert un temps de calcul très important, grandissant très vite avec le nombre de

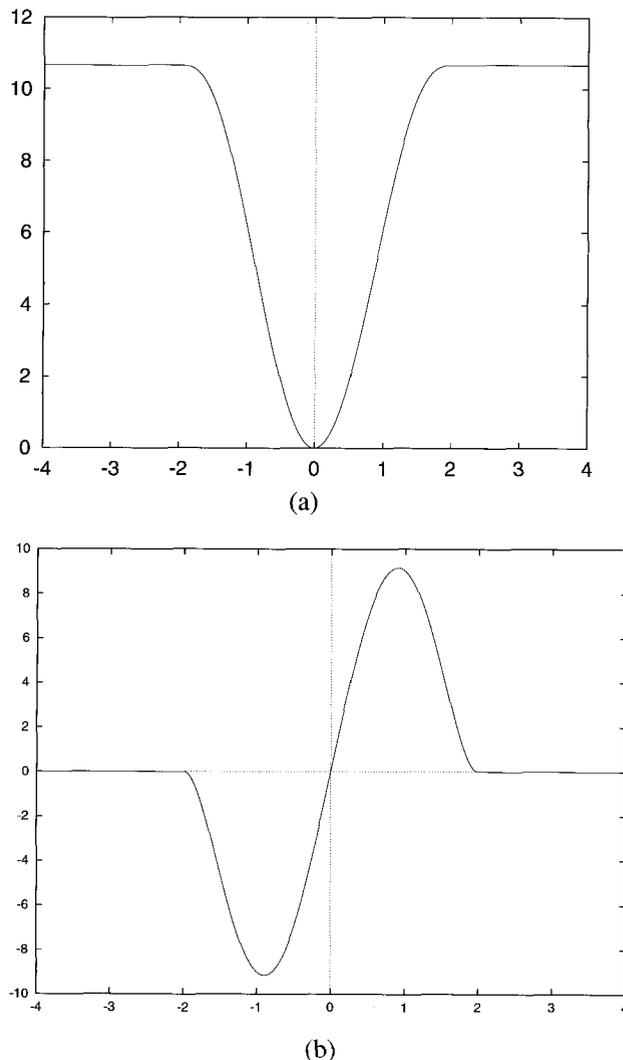


Figure 2. – Estimateur "biweight" de Tukey ( $C = 2$ ). a) Estimateur (fonction  $\rho$ ). b) Fonction d'influence  $\psi(\cdot)$ .

données considérés, même si une technique de Monte Carlo est utilisée, [22];

- son efficacité en cas de bruit gaussien sur les données est faible puisque à chaque itération, un nombre de données égal au nombre de paramètres est tiré pour fournir une estimée des paramètres.

Dans notre cas, les données (gradients spatio-temporels de l'intensité) peuvent être relativement bruitées, le modèle affine introduit ne constitue qu'une approximation du mouvement réel et l'hypothèse de conservation de l'intensité permettant de dériver l'équation de mesure n'est que rarement vérifiée en pratique. Ce sont des points auxquels l'estimateur des moindres carrés médian est plus sensible que le M-estimateur [22]. De plus, comme le nombre de données dans le support de calcul  $F$  est en général très grand, l'estimateur des moindres carrés médian serait très coûteux. Nous avons donc choisi de privilégier la méthode des M-estimateurs et avons retenu l'estimateur de Tukey.

## 4.2. MÉTHODES D'ESTIMATION ROBUSTE MULTIÉCHELLE PROPOSÉES

Il est bien connu que la résolution d'un problème d'estimation robuste peut être reformulée comme une estimation suivant les moindres carrés pondérés, en posant :

$$\sum_i \rho(r_i) = \sum_i \frac{1}{2} w_i r_i^2 \quad \text{avec} \quad r_i = y_i - M(\Theta, X_i)$$

Minimiser le critère revient à calculer les dérivées partielles par rapport à chaque paramètre  $\Theta_j$  et à les évaluer à 0, soit :

$$\sum_i \psi(r_i) \frac{\partial r_i}{\partial \Theta_j} = \sum_i w_i r_i \frac{\partial r_i}{\partial \Theta_j} = 0 \quad (18)$$

On obtient ainsi les coefficients  $w_i$  de pondération du moindres-carré :

$$w_i = \frac{\psi(r_i)}{r_i} \quad (19)$$

Le principe de la méthode des moindres carrés pondérés itérés (MCPI) [15] est alors le suivant. Si l'on dispose d'une estimation initiale des paramètres, les poids  $w_i$  sont tout d'abord évalués pour caractériser l'influence de chaque donnée; sinon, ils sont pris égaux à 1. Ensuite, une estimation du vecteur de paramètres  $\Theta$  est obtenue en résolvant le problème des moindres-carrés pondérés pour les valeurs courantes des poids  $w_i$ . Dans le cas où le modèle est linéaire par rapport aux paramètres, la résolution est immédiate. Les deux phases –calcul des poids et estimation– sont alors itérées jusqu'à atteindre la convergence.

Dans la section 3, la fonction  $E$  que l'on cherchait à minimiser était  $\sum e_i^2$ , avec  $e_i$  donné par (8). Il est donc naturel de la remplacer par :

$$E_r(\Theta) = \sum_{X_i \in F} \rho(e_i) \quad (20)$$

où la fonction  $\rho$  est la fonction de Tukey évoquée plus haut. Pour minimiser  $E_r$ , nous avons conservé le schéma d'estimation multirésolution et incrémentale de la section 3. La minimisation associée à l'estimation de chaque incrément est alors mise en œuvre à l'aide des MCPI. Nous nous plaçons tout-d'abord au niveau le plus grossier de la pyramide où à la toute première itération, on applique la méthode des MCPI au résiduel  $r_i$  donné dans l'équation (6). Puis, nous mettons à profit les niveaux plus fins successifs pour affiner itérativement et incrémentalement l'estimé de  $\Theta$ . Plus précisément, pour le calcul d'un incrément, on remplace l'expression de  $E_r$  par une expression approchée  $E'_r$  :

$$E'_r(\Delta\Theta_k) = \sum_{X_i \in F} \rho(r'_i) \quad (21)$$

où  $r'_i$  est donné par la formule (10). Ce résiduel étant linéaire par rapport à l'incrément, la méthode des MCPI s'applique facilement. Comme à l'itération  $k$ ,  $\Theta_k$  est supposé être proche

de la solution optimale, la valeur 0 peut constituer une estimée initiale de  $\Delta\Theta_k$  et peut être utilisée pour évaluer les poids initiaux dans la procédure des MCPI.

Dans ce qui précède, nous n'avons pas évoqué l'estimation de la variance du bruit (des données conformes au modèle). Celle-ci se fait habituellement parallèlement à l'estimation des paramètres et l'on utilise souvent comme estimée la valeur médiane de la déviation absolue [22] :

$$\hat{\sigma} = 1.48 \times \text{Med}_i(|r_i - \text{Med}_j(r_j)|) \quad (22)$$

Cette variance peut être reliée au terme  $C$  qui paramétrise la fonction  $\psi$  (16). Par exemple, dans [15], il est recommandé de prendre un facteur de proportionnalité proche de 4.7 entre  $C$  et  $\sigma$  pour assurer une meilleure efficacité en cas de bruit gaussien. Dans notre cas, puisque le résiduel correspond à une différence d'image déplacée,  $C$  est donc directement identifiable à une variation d'intensité entre deux images. De manière générale, un écart de 2 à 5 niveaux de gris est considéré comme correct dans un recalage global (avec un seul modèle de mouvement). Comme  $C$  représente la valeur limite du résiduel, au delà de laquelle la contribution d'un point devient nulle, on retiendra pour  $C$  des valeurs au-delà desquelles le recalage est considéré comme douteux, c'est-à-dire de l'ordre de 5 à 20. En fait, la valeur de ce paramètre dépend surtout de l'adéquation du modèle de vitesse au mouvement réel dans l'image : si le modèle utilisé est bien approprié,  $C$  peut être assez faible, sinon il est préférable de le prendre plus élevé. Il peut aussi dépendre de la nature des images, selon qu'elles présentent des variations localement faibles d'intensité, ou qu'elles sont au contraire très texturées. De manière plus détaillée, la constante  $C$  est choisie très large à basse résolution, puis est diminuée après chaque itération pour atteindre à la dernière minimisation, soit une valeur prédéfinie (typiquement 8), soit une valeur estimée de manière robuste. Plus précisément, dans ce dernier cas, à l'issue de l'estimation incrémentale au niveau le plus grossier de la pyramide, la formule (22) est utilisée pour calculer  $\hat{\sigma}$ , et l'on prend comme valeur finale à atteindre pour  $C$ ,  $4.7 \hat{\sigma}$  comme nous le notons au début de ce paragraphe. Des essais menés avec ces deux options de valeur finale pour le paramètre  $C$  ont donné des résultats similaires, légèrement meilleurs pour la technique robuste, mais avec un coût calcul beaucoup plus élevé dans ce cas. Ces essais ont également montré qu'une certaine latitude existe pour fixer la valeur finale de  $C$ .

Dans la plupart des cas, la valeur du nombre maximal  $\lambda$  d'itérations effectuées à un même niveau dans l'algorithme RMR, présenté dans la figure 3 sera faible (égale à 5 ou 6). De la même façon, les MCPI se réduiront à quelques itérations (5 ou 6).

Nous avons dérivé un deuxième algorithme robuste, qui reprend globalement les mêmes principes. Il repose sur l'application directe de la méthode des MCPI à la fonction  $E_r$ . Plus précisément, nous avons donc :

$$E_r(\Theta) = \sum_{X_i \in F} \rho(e_i) = \sum_{X_i \in F} w_i e_i^2 \quad (23)$$

```

C ← Max(|ItL|)
Θ̂L ← MCPI(résiduel (6), C)
POUR l = niveau L au niveau 0 FAIRE
  iter ← 0
  FAIRE
    C ← f(C)
    Δ̂Θl ← MCPI( résiduel(10), C )
    Θ̂l ← Θ̂l-1 + Δ̂Θl et iter ← iter + 1
  TANT QUE (iter < λ et ||Δ̂Θl|| >  $\frac{d}{2^l}$  )
  Si l ≠ 0 : Âl-1 ← P(Âl), ξ̂l-1 ← ξ̂l
FIN POUR
Θ̂est ← Θ̂0

```

Figure 3. – Algorithme Robuste Multirésolution (RMR).

Comme au départ nous ne disposons pas d'estimée du vecteur  $\Theta$ , tous les poids sont fixés à 1. La fonction  $E_r(\Theta)$  est alors identique à  $E(\Theta)$  (relation (7)), et peut donc être minimisée avec l'algorithme multirésolution MCMR. On obtient alors l'estimée  $\hat{\Theta}$ , à l'aide duquel on calcule les poids  $w_i$  suivant la formule (19), le résiduel étant ici la quantité  $\tilde{e}_i$  :

$$\tilde{e}_i = I(X_i + B_i \hat{A}, t + 1) - I(X_i, t) + \hat{\xi} \quad (24)$$

La pondération est propagée à tous les niveaux de façon appropriée (on utilise les mêmes filtrage gaussien et sous-échantillonnage que ceux utilisés pour construire la pyramide des données), et on relance alors une nouvelle estimation à travers la pyramide. La structure de l'algorithme de minimisation à chaque descente reste du type de celle du MCMR, mais désormais en considérant des moindres carrés pondérés. Cette version permet d'éviter le calcul des MCPI pour chaque estimation d'un incrément, mais elle nécessite d'effectuer plusieurs descentes de la pyramide. Nous l'avons dénommée *PSM* (Pseudo M-estimateur). Elle est résumée à la figure 4.

```

wi ← 1.0, k ← 0 et (Θ̂est)0 ← 0
FAIRE
  (Θ̂est)k+1 ← MCMR(wi)
  wi ←  $\frac{\psi(\tilde{e}_i, C)}{\tilde{e}_i}$  et k ← k + 1
TANT QUE ||(Θ̂est)k - (Θ̂est)k-1|| > d' et k < λ
Θ̂est ← (Θ̂est)k

```

Figure 4. – Algorithme Pseudo M-estimateur (PSM).

### 4.3. ETAPES COMPLÉMENTAIRES

Nous n'avons pas tenu compte dans les paragraphes précédents de certains problèmes de l'estimation de mouvement, notamment celui des données. Ces problèmes spécifiques, qui produisent des effets indésirables dans les algorithmes que nous avons définis précédemment, sont de trois ordres :

1. le mouvement réel que l'on souhaite estimer pourrait être décrit par moins de paramètres que le nombre retenu dans notre modèle; de manière équivalente, il peut arriver que le mouvement réel et la distribution spatiale des gradients d'intensité ne contraignent pas suffisamment l'estimation de tous les paramètres du modèle.
2. la minimisation initiale est basée sur les moindres carrés; alors, les mouvements des objets secondaires peuvent être utilisés pour contraindre suffisamment les degrés de liberté laissés indéterminés ou mal conditionés par les observations correspondant au mouvement dominant. Cet effet peut être d'autant plus marqué que les régions supportant le mouvement dominant sont très peu texturées, et qu'à l'opposé, les régions correspondant aux mouvements secondaires le sont fortement.
3. notre estimation est grandement dépendante des paramètres d'estimation initiaux; si l'un des cas précédents se produit, notre algorithme sera susceptible de converger vers un minimum local éloigné de la solution optimale.

La figure 5 présente un exemple type. La figure 5a est censé nous montrer les vecteurs vitesses réels dans l'image, qui ne sont tracés que pour des pixels ayant un gradient d'intensité important, tandis que la figure 5b contient le champ estimé que l'on pourrait obtenir dans une telle situation. Comme on peut le constater, les vecteurs de mouvement aux points de mesure possibles sont en fait peu distordus, bien que les deux translations soient "fusionnées" au sein d'un même modèle de rotation qui n'a pas grand chose à voir avec les mouvements réels. Une première possibilité serait de faire confiance à un estimateur plus robuste, du moins pour la première itération, comme les moindres carrés médian. Cette solution souffrira cependant des inconvénients évoqués au paragraphe 4.1.. En fait, une solution plus simple et plus efficace à ce problème et qui fonctionne dans un grand nombre de cas, consiste à commencer le processus d'estimation en considérant un modèle de translation, et à introduire les modèles plus complexes uniquement par la suite, après quelques itérations. La figure 6 présente l'algorithme RMR modifié dans ce sens : nous commençons par estimer un modèle constant du niveau  $L$  au niveau  $L_a$  inclus, puis nous considérons du niveau  $L_a$  au niveau 0 le modèle complet retenu, qui est fréquemment le modèle affine. Dans nos applications, nous choisissons généralement pour  $L_a$  une valeur de 2, alors que le nombre de niveaux  $L$  est de 3 ou 4. Notons que dans le cas d'une séquence, il est possible à l'aide des estimations effectuées dans les images précédentes de déterminer une valeur opportune du niveau  $L_a$  auquel il est souhaitable de commencer l'estimation des paramètres linéaires, voire quadratiques.

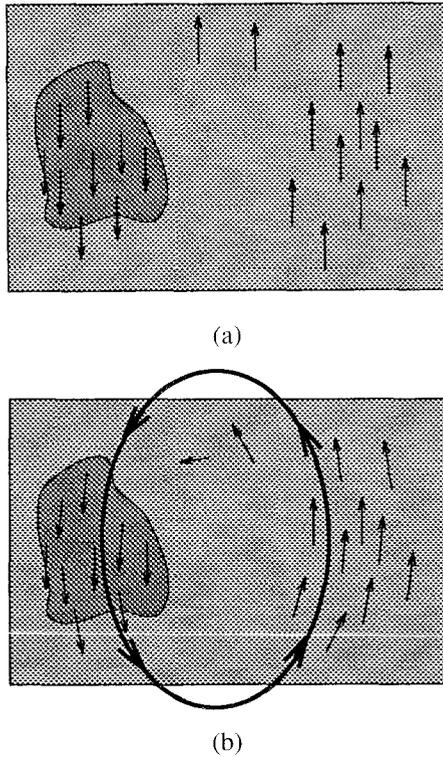


Figure 5. – a) Exemple de problème d'estimation ambiguë : l'objet dominant se déplace vers le haut, tandis que le second objet est animé d'une translation vers le bas (les vecteurs de mouvement ne sont tracés qu'aux points de gradient d'intensité supposé significatif). b) le mouvement estimé résultant peut être une rotation.

L'avantage essentiel de cette méthode repose sur l'algorithme d'estimation incrémentale qui est très efficace lorsque un modèle constant est utilisé (voir [10]), et facilite la discrimination entre les mouvements de différents objets ou entre le fond et des objets mobiles.

L'algorithme PSM peut également être modifié suivant ce schéma, en estimant un mouvement constant lors de la première estimation multirésolution. Ce nouvel algorithme (voir figure 7) est similaire à la méthode présentée dans [18]. Cependant, dans cet article, après chaque estimation multirésolution, un algorithme de détection explicite (plus complexe qu'une simple procédure de pondération) est utilisé : il élimine les points pour lesquels le mouvement calculé n'est pas satisfaisant, et attribue un poids de 1 pour tous les autres. Cette décision binaire peut être trop catégorique lorsqu'il n'existe pas de carte de détection explicite ou dans le cas d'entités "floues", comme dans les images météorologiques). De plus, après l'estimation du modèle constant, cette décision peut aussi déboucher sur le choix d'un nombre limité de points formant une région qui sera vraiment susceptible d'avoir un mouvement purement translationnel. Les itérations suivantes avec un modèle affine n'auront d'autre possibilité que de rester bloquées dans une telle configuration. Il ne sera donc pas facile de traiter des situations impliquant des mouvements plus complexes.

Dans notre cas, le calcul des poids après l'estimation du modèle de mouvement constant se fait avec une constante  $C$  dans la fonction

```

Nombre de paramètres ← 2
C ← Max(|ItL|)
ΘL ← MCPI(résiduel (6), C)
TANT QUE l ≥ 0 FAIRE
  iter ← 0
  FAIRE
    C ← f(C)
    ΔΘl ← MCPI(résiduel(10), C)
    Θl ← Θl + ΔΘl et iter ← iter + 1
  TANT QUE (iter < λ et ||ΔΘl|| >  $\frac{d}{2^l}$ )
  SI l = La et Nombre de paramètres=2
    Nombre de paramètres ← n
  SINON
    SI l ≠ 0 : Al-1 ← P(Al), ξl-1 ← ξl, l ← l - 1
FIN TANT QUE
Θest ← Θ0
    
```

Figure 6. – Algorithme Robuste Multirésolution modifié (RMRmod).

```

Nombre de paramètres ← 2, wi ← 1.0 et k ← 0
(Θest)0 ← MCMR(wi)
wi ←  $\frac{\psi(\bar{e}_i, 2 \times C)}{\bar{e}_i}$ 
Nombre de paramètres ← n
FAIRE
  (Θest)k+1 ← MCMR(wi)
  wi ←  $\frac{\psi(\bar{e}_i, C)}{\bar{e}_i}$  et k ← k + 1
TANT QUE ||(Θest)k - (Θest)k-1|| > d' et k < λ
Θest ← (Θest)k
    
```

Figure 7. – Algorithme Pseudo M-estimateur modifié (PSMmod).

$\Psi$  deux fois plus importante que lors des itérations suivantes. Ceci permet de prendre en compte le fait que le modèle constant est sans doute trop fruste. De cette façon, nous conserverons suffisamment de points pour faire une estimation des paramètres affines dans les phases suivantes. Notons qu'il serait également possible de faire varier  $C$  après chaque itération, comme pour l'algorithme RMR. Enfin, mentionnons ici que dans les deux algorithmes, un pixel dont le poids est nul à une itération donnée n'est pas définitivement écarté, et peut donc de ce fait récupérer un poids non nul si l'affinement de l'estimation joue en "sa faveur". Ceci nous permet d'estimer des mouvements qui ne sont pas nécessairement cons-

tants, comme le montreront les résultats présentés dans la section suivante.

## 5. Résultats

Des expériences ont été menées sur des images réelles animées de mouvements synthétiques pour obtenir une évaluation quantitative précise du comportement des algorithmes. Des séquences réelles ont été également traitées pour montrer l'intérêt et l'efficacité des algorithmes pour trois scènes complexes, les deux premières correspondant à des scènes d'extérieur, la dernière à des images prises par un satellite météorologique.

### 5.1. EXPÉRIMENTATIONS SUR UNE IMAGE RÉELLE AVEC DES MOUVEMENTS SYNTHÉTIQUES

Nous avons construit cet exemple de la façon suivante. Nous avons pris l'image de la figure 11.a. Nous lui avons appliqué le mouvement synthétique de la figure 8.a pour construire une

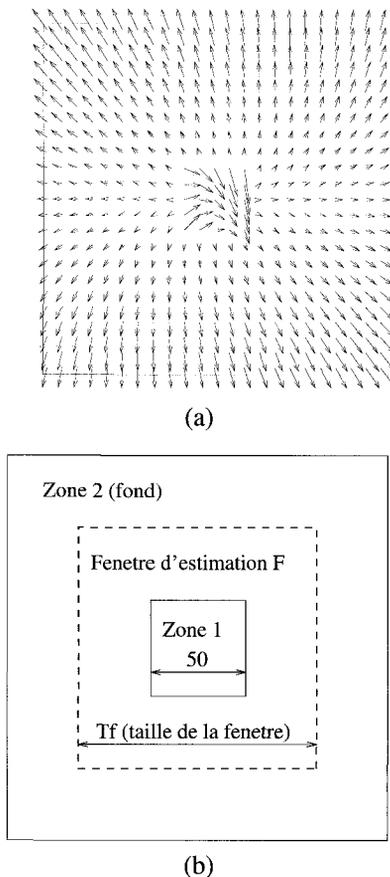


Figure 8. – a) champ synthétique des vitesses appliqué sur l'image; b) schéma d'expérimentation

seconde image (en utilisant une interpolation bilinéaire pour les points obtenus de coordonnées non entières). Ce mouvement est en fait composé de deux modèles affines différents, l'un appliqué sur une fenêtrée carrée de l'image (zone 1,  $Z_1$ , (voir figure 8.b)), l'autre sur le reste de l'image (zone 2,  $Z_2$ ). Les valeurs des paramètres des deux modèles sont données dans le tableau 1.

Tableau 1. – Valeurs des paramètres des modèles affines  $A_{th}$  utilisés dans l'exemple synthétique.

Paramètres	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$
Modèle zone 1	1.0	-0.03	0.0	1.0	0.08	-0.06
Modèle zone 2	0.0	0.01	0.005	0.0	0.0	0.02

L'expérimentation consiste simplement à étudier le comportement de l'estimateur lorsque la proportion des deux zones présentes dans la fenêtrée –ou dans le support– d'estimation  $F$  varie. Pour cela, les six paramètres du modèle affine ont été estimés sur un support carré  $F$  de largeur  $Tf$  variable (Fig. 8.b). Nous avons retenu comme indice d'adéquation des paramètres estimés dans chaque zone, l'erreur moyenne donnée par la relation suivante ( $n = 1, 2$ ) :

$$err_n = \frac{1}{card(F \cap Z_n)} \sum_{X_i \in (F \cap Z_n)} \|\vec{V}_{A_{est}}(X_i) - \vec{V}_{A_{th}}(X_i)\|$$

L'étude d'autres erreurs (comme l'erreur angulaire) ou même l'étude directe de l'évolution des paramètres donne des résultats similaires et permet d'aboutir aux mêmes conclusions. Dans toutes les expériences, la valeur de  $C$  est de 5 pour l'algorithme RMR et de 9 pour l'algorithme PSM. Les graphiques de la figure 9 présentent les résultats obtenus. Comme prévu, l'estimateur des moindres carrés multirésolution MCMR effectue un moyennage entre les deux mouvements (cela est encore plus évident sur la figure 10), alors que les estimateurs robustes RMR et PSM, même lorsqu'un fort pourcentage de points ne subit pas le mouvement dominant, fournissent une estimation correcte. Pour ces deux derniers, la plage de transition (très réduite) séparant les tailles de fenêtrées pour lesquelles le mouvement principal estimé est celui de la zone 1, et celles pour lesquelles il s'agit de celui de la zone 2, se trouve toutefois aux alentours de  $Tf = 60$ , alors que la zone 1 représente encore 70% de la surface. On aurait pu imaginer que le point de rupture se trouve près de 50%. Ceci s'explique par le fait que la mesure est basée sur l'équation de contrainte du mouvement, dans laquelle le gradient de l'intensité joue un rôle notable. Les zones uniformes, qui occupent une surface non négligeable, ne procurent aucune information de mouvement, et la transition se produit en fait théoriquement pour une fenêtrée  $F$  telle que :

$$\sum_{F \cap Z_1} \|\nabla I\| = \sum_{F \cap Z_2} \|\nabla I\| \quad (25)$$

Dans ce premier exemple, la zone 1 est située sur une zone de feuillage où il y a peu de forts gradients d'intensité comparativement à l'environnement proche (voitures, panneau). Le calcul du

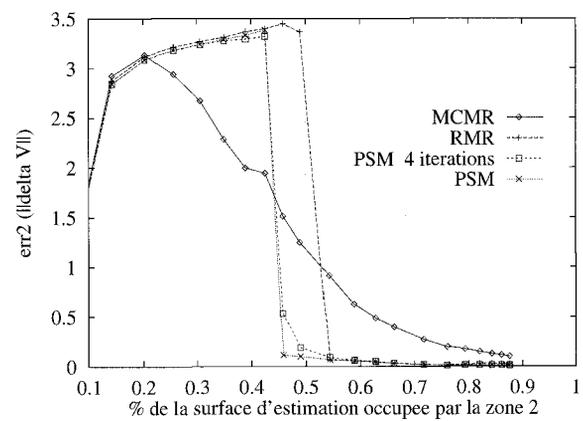
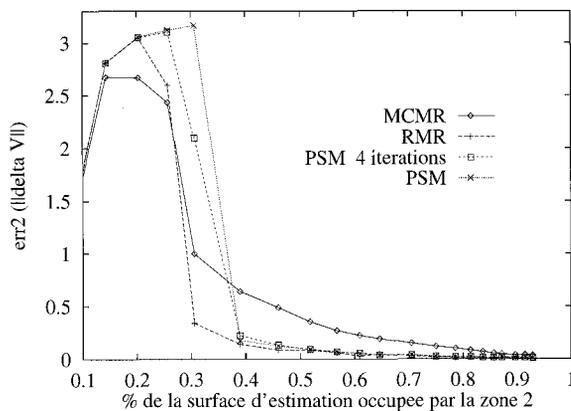
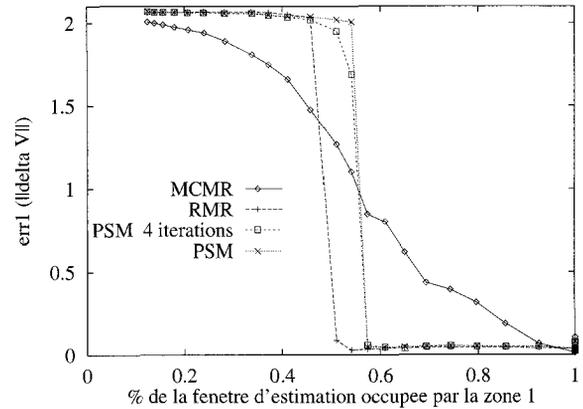
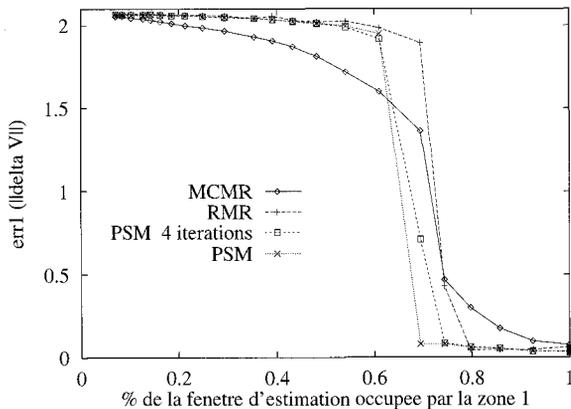


Figure 9. – Erreur  $err_n$  en fonction du taux d'occupation du support d'estimation par la zone  $n$ ; à gauche, pour la zone 1; à droite pour la zone 2. Cas du carré placé au centre de l'image

Figure 10. – Erreur  $err_n$  en fonction du taux d'occupation du support d'estimation par la zone  $n$ ; à gauche, pour la zone 1; à droite pour la zone 2. Cas du carré placé sur la voiture au premier plan

point d'équilibre de la relation (25) donne (au niveau 0)  $Tf = 63$ , soit 63% du support occupé par la zone 1. Pour corroborer ces remarques, nous avons recommencé l'expérience mais en centrant cette fois-ci la zone 1 sur une portion de la voiture au premier plan. Les mêmes modèles de mouvement que dans l'exemple précédent sont appliqués. Les résultats sont présentés en figure 10. Le point d'équilibre (25) est atteint cette fois-ci lorsque 48% de la fenêtre d'estimation est occupée par la zone 1, ce qui concorde avec les résultats montrés par les diagrammes de la figure 10.

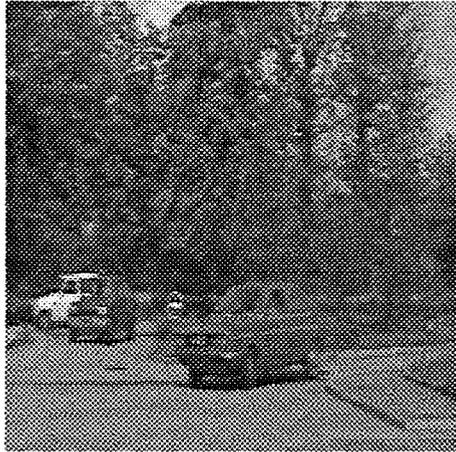
## 5.2. EXPÉRIMENTATIONS AVEC DES SÉQUENCES RÉELLES

### Séquence "voitures"

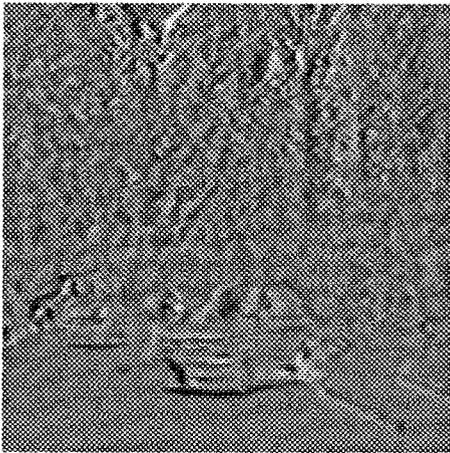
Dans de nombreux cas d'analyse dynamique, il est utile et nécessaire de déterminer ou de compenser dans un premier temps le mouvement de la caméra pour ensuite détecter les objets mobiles. La figure 11.a montre la première image de la séquence réelle traitée, dont le contenu dynamique est souligné par l'image (Fig. 11.b) des différences temporelles entre les deux images considérées, auxquelles un "offset" de 128 a été ajouté (une valeur

grise correspond ainsi à une différence nulle, et, plus noir ou blanc est le point, suivant le signe de la différence, plus cette dernière est importante). Ce contenu dynamique comprend trois composantes essentielles : un panoramique de la caméra de la droite vers la gauche induisant une translation apparente vers la droite dans l'image, un mouvement un peu erratique du feuillage dû au vent surtout au centre, et enfin les déplacements de deux voitures. Pour illustrer plusieurs cas typiques, l'image a été divisée en quatre blocs. Sur chacun d'eux, on estime le mouvement principal et on calcule les résiduels  $\tilde{e}_i$  correspondants (donnés par (24)). Le nombre de niveaux dans la pyramide est de quatre (nous utilisons un critère simple pour fixer le nombre de niveaux en fonction de la taille du support d'estimation  $F$ ).

La figure 12 présente les résultats obtenus avec les algorithmes MCMR et PSM modifié. Dans les deux blocs du haut, les mouvements du feuillage sont globalement incohérents et faibles. Le mouvement dominant est donc clairement le panoramique que l'algorithme MCMR et l'algorithme PSM modifié sont en mesure d'estimer comme le montrent les champs  $\vec{V}_{A_{est}}$  de la figure 13. Les vecteurs de vitesse sont tracés aux points où le mouvement dominant est considéré comme adéquat selon la valeur de  $w_i$  qui est fonction de  $\tilde{e}_i$ . En revanche, dans le bas, les mouvements



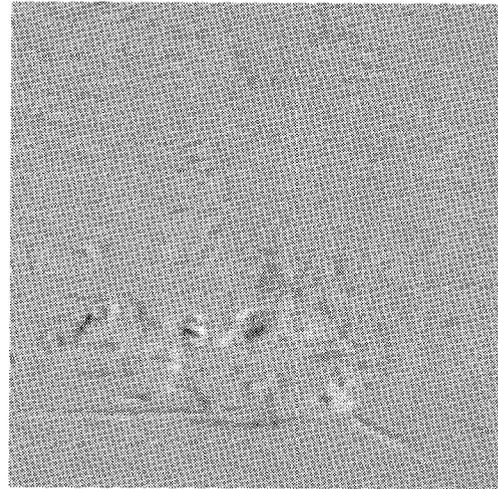
(a)



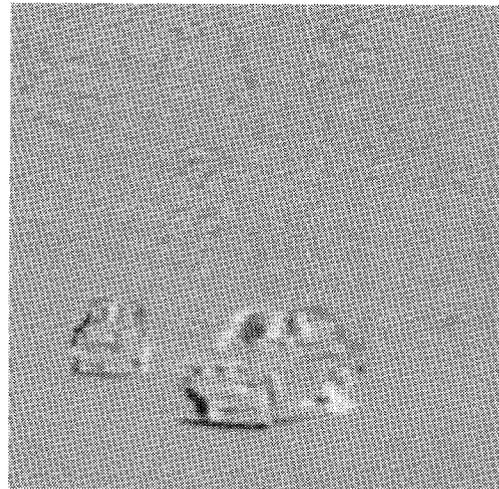
(b)

Figure 11. – Séquence “voitures” : a) première image; b) image des différences temporelles entre les deux images considérées

rigides des deux voitures constituent des perturbations cohérentes importantes, et l'image d'erreur 12.a ainsi que le mouvement estimé résultant 13.a mettent clairement en évidence l'effet de moyennage de l'algorithme MCMR. L'erreur  $\bar{e}_i$  n'est petite ni sur la partie fixe de la scène, ni sur les voitures en mouvement et les mouvements estimés sont très erronés. Par contre, les figures 12.b et 13.b montrent bien que le mouvement panoramique a été parfaitement déterminé par l'algorithme PSM modifié alors que les voitures couvrent une surface non négligeable dans chaque bloc. Non seulement le mouvement panoramique de la caméra est très bien compensé, aucun élément de la partie fixe de la scène n'apparaissant dans l'image d'erreur de la figure 12.b, mais simultanément les zones correspondant à des objets mobiles dans la scène sont beaucoup mieux mises en évidence que dans le cas de l'algorithme MCMR. On peut bien sûr envisager une seconde étape, où pour les zones ainsi détectées, on relance l'algorithme PSM modifié pour déterminer les mouvements secondaires dans l'image, et ainsi de suite.



(a)



(b)

Figure 12. – Image des différences d'image compensées  $\bar{e}_i$  : a) MCMR; b) PSM modifié ( $C = 9$ );

### Séquence ROND-POINT

Les figures 14a, 14c et 14e représentent trois images de la séquence ROND-POINT. Ici, les déplacements dominants dans l'image sont dus au mouvement de la caméra, qui est montée sur le côté gauche d'une voiture approchant un rond-point, et qui pointe latéralement par rapport au déplacement de ce véhicule. Cependant, comme les différences entre les profondeurs des objets statiques dans la scène sont importantes, le modèle du mouvement dominant correspond uniquement au mouvement du fond (c.a.d., les maisons principalement).

Dans cette expérience, nous avons estimé un modèle affine entre chaque couple d'images consécutives avec les algorithmes MCMR, RMR et PSM modifiés. Nous avons utilisé quatre niveaux dans la pyramide ( $L = 3$ ) et retenue une valeur de 2 pour  $L_a$ . Nous avons choisi 8 pour valeur finale de  $C$ . Les modèles calculés sont

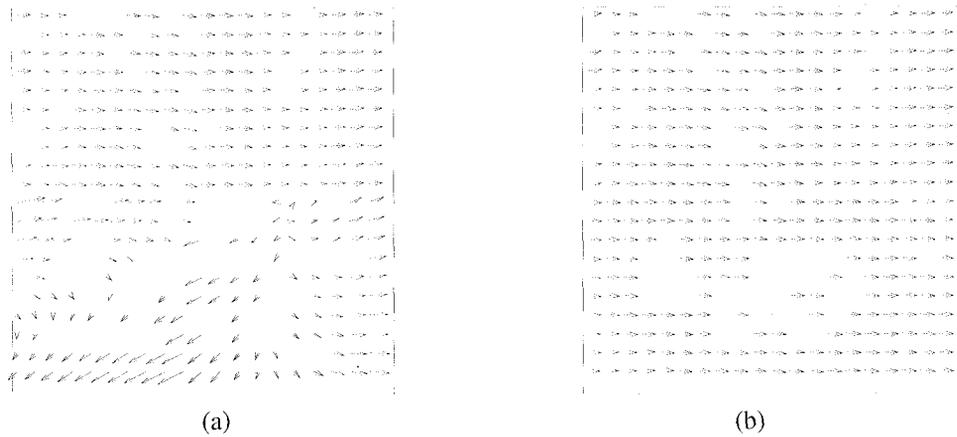


Figure 13. – Champ de vitesses associé au modèle de mouvement estimé : a) MCMR; b) PSM modifié( $C = 9$ );

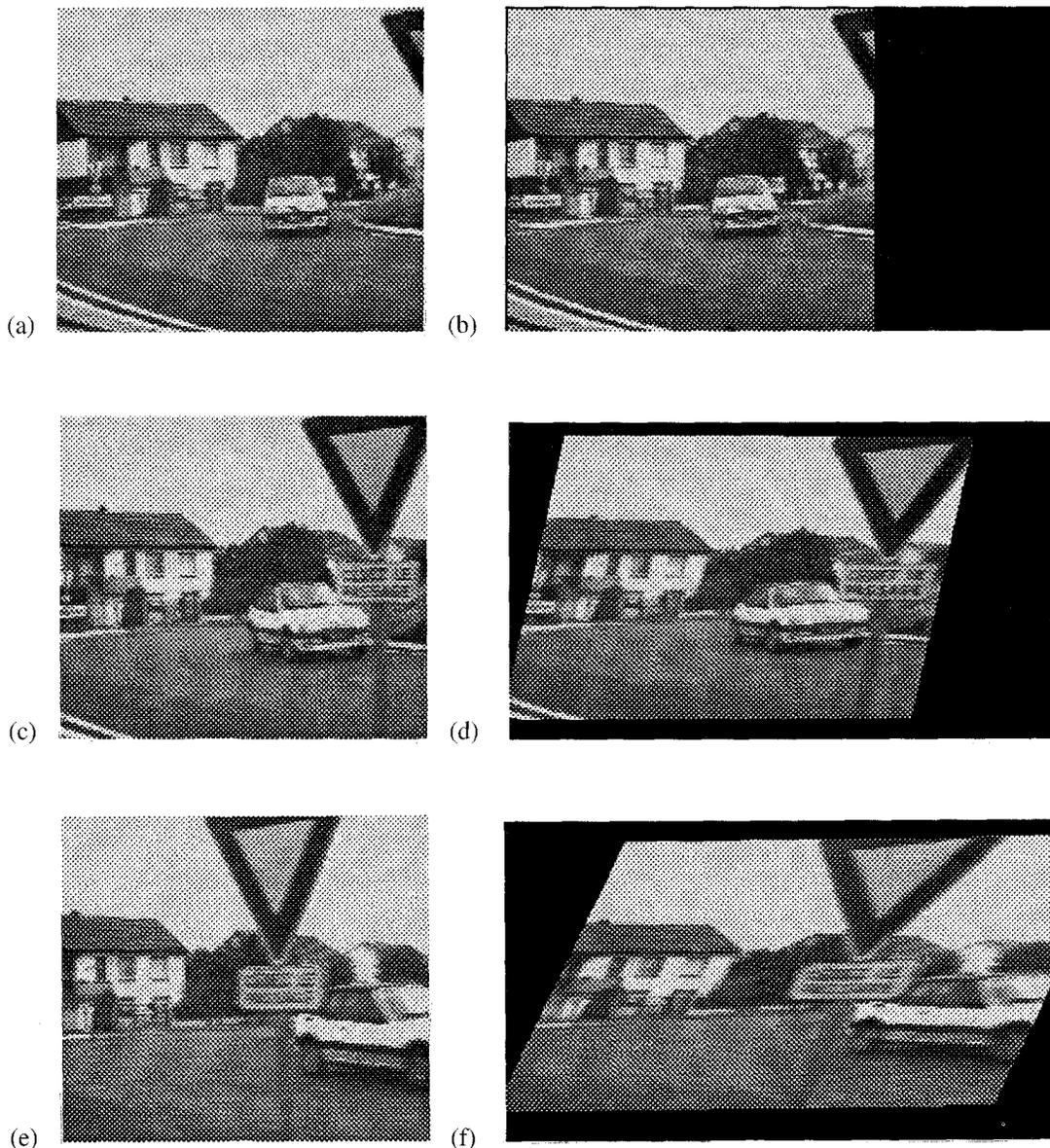


Figure 14. – a) c) e) Images aux instants  $t_{62}$ ,  $t_{67}$  et  $t_{72}$  compensées avec les modèles de mouvement calculés par l'algorithme RMR modifié.  
 b) d) f) Images aux instants  $t_{62}$ ,  $t_{67}$  et  $t_{72}$  compensées avec les modèles de mouvement calculés par l'algorithme PSM modifié.

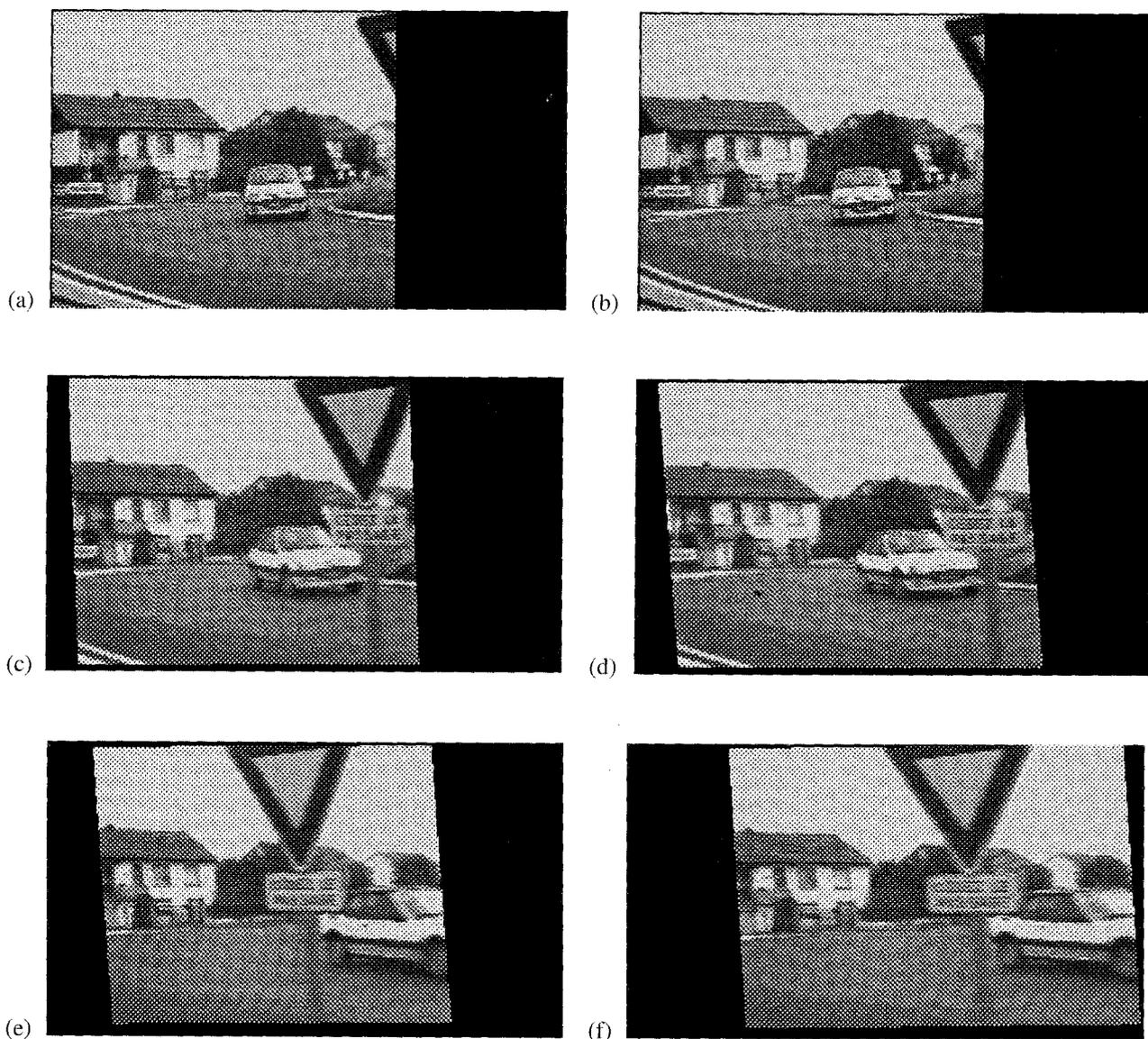


Figure 15. - a) c) e) Images aux instants  $t_{62}$ ,  $t_{67}$  et  $t_{72}$  compensées avec les modèles de mouvement calculés par l'algorithme RMR modifié.  
 b) d) f) Images aux instants  $t_{62}$ ,  $t_{67}$  et  $t_{72}$  compensées avec les modèles de mouvement calculés par l'algorithme PSM modifié.

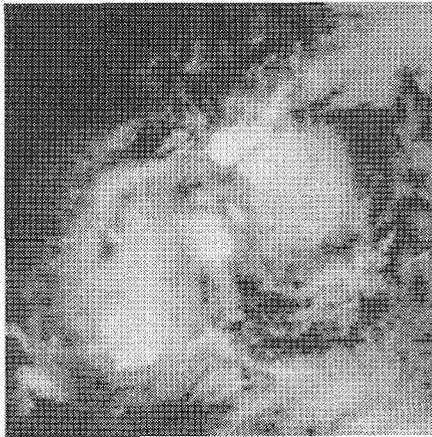
alors utilisés pour générer les séquences compensées 14b-d-f, 15a-c-e et 15b-d-f. Si le déplacement d'une région est bien compensé par le mouvement estimé, celle-ci doit alors rester au même endroit dans les images successives.

Une fois encore, les images 14b-d-f indiquent clairement l'effet de moyennage de l'algorithme MCMR, puisque toute l'image se déforme au cours du temps. A l'opposé, les figures 15a-c-e montrent que l'algorithme RMR modifié estime de manière cohérente le mouvement apparent du fond sur toute la durée de la séquence : les maisons restent fixes dans la séquence compensée. Enfin, l'algorithme PSM modifié estime au départ correctement le mouvement du fond (de  $t_{62}$  à  $t_{63}$ ), passe par une phase où il

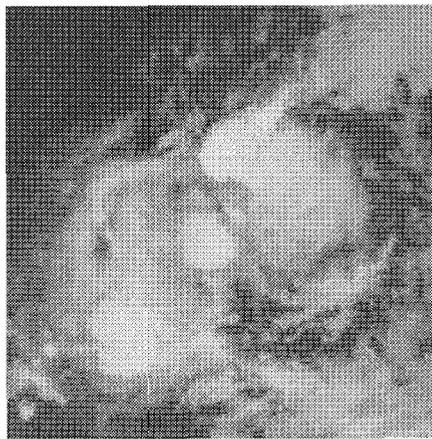
effectue un moyennage à l'image 64 (voir figures 15b-d-f : entre 15b et 15d, les maisons ne sont pas fixes; le panneau se déplace à droite), et ensuite se cale sur le mouvement du panneau de  $t_{65}$  à  $t_{72}$ . La différence entre les performances des deux algorithmes sera commentée à l'issue de cette section.

#### Séquence météo

Les figures 16.a et 16.b correspondent à deux images météorologiques successives d'une zone équatoriale, prises à une demi-heure d'intervalle dans le canal infrarouge. La scène est constituée principalement d'un développement ascensionnel de forte convection entraînant un mouvement de rotation-spirale dont le centre



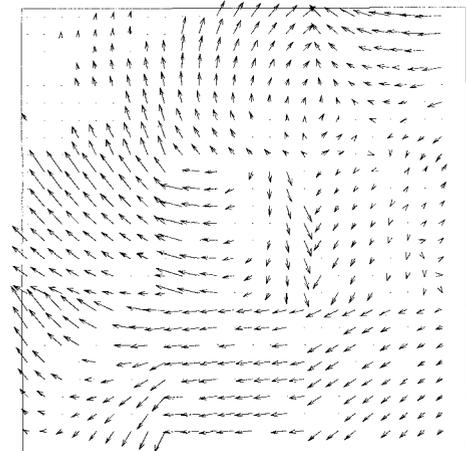
(a)



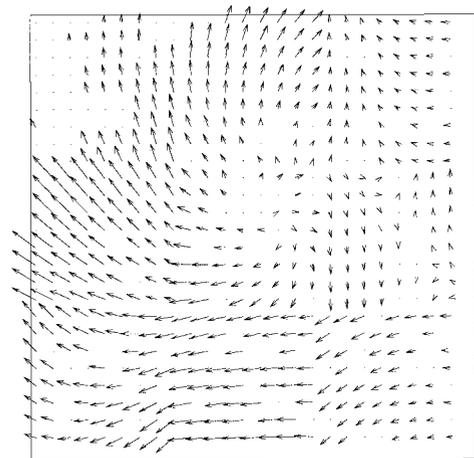
(b)

Figure 16. – Séquence météo : a) première image; b) deuxième image.

se situe au cœur du développement (coin supérieur droit du bloc central). Cependant, sur la droite de l'image se trouve des nuages de moyenne altitude tournant globalement en sens inverse, et un second développement dans le bas à gauche dévie certains nuages vers le bas. De plus, de nombreux phénomènes de transparence entre couches sont présents, l'augmentation de la température de l'atmosphère entre les deux acquisitions dissipe une partie des nuages (surtout de la couche moyenne) et l'ascension des nuages provoque des variations de niveau de gris importants. L'image a été divisée en neuf blocs de  $75 \times 75$  sur lesquels le modèle affine de mouvement peut constituer une approximation à peu près valable. Cette approximation étant cependant plus lâche que dans les cas précédents, on utilise ici une variance plus élevée ( $C = 18$ ). Les champs de vitesse  $\vec{V}_{Est}$  obtenus sont présentés tels quels, sans post-traitement pour éliminer l'effet de bloc. L'algorithme MCMR obtient des résultats qui sont dans l'ensemble convenables (Fig. 17.a), mais montre ses limites dans le bloc central du fait de la présence du mouvement ascensionnel. En revanche, l'algorithme PSM modifié, de même que l'algorithme RMR modifié qui donne des résultats similaires, retrouve tout à fait le mou-



(a)



(b)

Figure 17. – Séquence météo : a) première image; b) deuxième image.

vement de rotation dans ce bloc central et l'on peut remarquer que les liaisons entre blocs sont plus cohérentes. Les valeurs du paramètre  $\xi$  sont montrées dans le tableau 2, et prouvent qu'il est opportun d'introduire ce terme complémentaire dans le modèle. Un  $\xi > 0$  indique en fait une diminution moyenne des niveaux de gris sur le bloc. Les valeurs estimées le sont presque toutes, ce qui est cohérent avec la dissipation constatée de certains nuages sur la séquence complète.

## 6. Comparaison entre les algorithmes RMR et PSM modifiés

Ces deux algorithmes procurent généralement des résultats très similaires, particulièrement lorsqu'il existe de manière évidente une région dominante dont le mouvement peut se décrire à l'aide du modèle paramétrique choisi. Cependant, certaines différences,

**Tableau 2.** – Valeurs du paramètre  $\xi$  estimées par bloc.

0.5	2.0	-1.1
3.4	2.3	2.6
0.7	-0.6	4.0

expliquant en partie les différents comportements observés sur la séquence ROND-POINT, peuvent être identifiées :

1. supposer que le mouvement est constant uniquement sur les niveaux de résolution les plus grossiers est une hypothèse plus faible que supposer cela sur une première estimation multirésolution complète. Cela implique que l'algorithme RMR modifié permet de récupérer une classe plus large de modèles affines que l'algorithme PSM modifié.
2. avec l'algorithme PSM, tous les points sont considérés de manière équivalente lors de la première estimation multirésolution, ce qui produit une estimation moyennée, même lorsqu'un modèle constant est utilisé. En revanche, l'algorithme RMR élimine les erreurs importantes dès les premières itérations dans l'estimation multirésolution.

Considérons maintenant l'aspect calculatoire. L'algorithme PSM nécessite plusieurs descentes à travers la pyramide. Sa complexité est approximativement celle de l'algorithme MCMR multipliée par le nombre de passes. Dans l'algorithme RMR, il n'y a qu'une descente du plus grossier au plus fin, mais le calcul de chaque incrément  $\Delta\Theta_k$  implique une minimisation par la méthode des MCPI.

**Tableau 3.** – Temps cpu moyen (en secondes) de l'estimation. Les dix images de taille  $224 \times 256$  de la séquence ROND-POINT sont considérées ; le nombre d'incrément à un niveau est toujours limité à 8 ; le nombre d'itérations des MCPI est limité à 6, et le nombre de descentes de pyramide est également limité à 6 pour l'algorithme PSM modifié.

Algorithme	MRLS	RMRmod	PSMmod
Sparc 2	11.66	11.85	38.20
Sparc 10	4.05	4.05	12.60

En fait, le tableau 3 indique (pour notre implémentation sans optimisation particulière) que la complexité des algorithmes MCMR et RMR est équivalente. Ceci s'explique de la façon suivante. D'une part les MCPI convergent très rapidement, et donc, cette minimisation ne consomme pas trop de temps cpu, et d'autre part, le nombre d'incrément à calculer à un niveau pour atteindre la stabilité est moins important avec l'algorithme RMR qu'avec les algorithmes MCMR ou PSM grâce aux MCPI (c.a.d., les incréments sont mieux estimés). Ainsi, comme chaque calcul d'incrément implique l'interpolation de l'image des intensités en chaque point (nous effectuons une interpolation bilinéaire), l'algorithme RMR épargne de ce fait du temps cpu comparativement aux deux autres algorithmes.

## CONCLUSION

Nous avons présenté dans cet article une méthode d'estimation robuste multiéchelle de modèles paramétrés de mouvement. Deux variantes de cette approche (aux performances voisines) ont été décrites et ont été comparées favorablement à une technique de moindres-carrés multirésolution, à travers des évaluations relativement conséquentes sur des exemples synthétiques et des données réelles correspondant à des scènes complexes de types variés. Ce module a en fait de nombreuses implications. Pouvoir évaluer correctement le mouvement global dans l'image ou dans une zone de l'image, sans que la présence de mouvements secondaires éventuellement significatifs, ou de zones où la mesure est mal conditionnée, ne vienne perturber cette estimation, et ce sans passer par une phase explicite de segmentation, est en soi d'un intérêt évident et d'une utilisation immédiate dans de nombreux contextes. Citons par exemple, la robotique mobile pour des tâches de fixation d'attention, la télésurveillance pour la détection d'éléments mobiles par une caméra en panoramique permanent, l'échographie cardiaque où on peut s'intéresser à l'interprétation du mouvement dans des zones prédéfinies comme dans [21]. Ce module peut aussi représenter une étape préalable efficace d'un schéma complet de segmentation en régions, ou de mesure du champ des vitesses apparentes.

*Cette étude a été soutenue par le GDR-PRC "Communication Homme-Machine" (Pôle Vision), et par la Région Bretagne (via le cofinancement d'une bourse de thèse).*

## BIBLIOGRAPHIE

- [1] J.K. Aggarwal and N. Nandhakumar. On the computation of motion from sequences of images- a review. *Proc. of the IEEE*, Vol.76, No.8 :917-935, August 1988.
- [2] B. Basclé and N. Deriche. Stereo matching, reconstruction and refinement of 3D curves using deformable contours. In *Proc. 4th Int. Conf. Computer Vision, Berlin*, pages 421-430, May 1993.
- [3] R. Battiti, E. Amaldi, and C. Koch. Computing optical flow across multiple scales : an adaptive coarse-to-fine strategy. *Intern. J. Comput. Vis.*, 6 :2 :133-145, 1991.
- [4] J.R. Bergen, P. Anandan, K. Hanna, and R. Hingorani. Hierarchical model-based motion estimation. In *Proc. of 2<sup>nd</sup> European Conf. Computer Vision*, pages 237-252, Springer-Verlag, S.Margherita Ligure, Italie, 1992.
- [5] J.R. Bergen, P.J. Burt, R. Hingorani, and S. Peleg. Computing two motions from three frames. In *Proc. 3rd Int. Conf. on Computer Vision*, pages 27-32, Osaka, Déc. 1990.
- [6] M. J. Black. *Robust incremental optical flow*. PhD thesis, N° 923, Yale University, Computer Science Dept, Septembre 1992.
- [7] P. Boutheymy. Modèles et méthodes pour l'analyse du mouvement dans une séquence d'images. *Technique et Science Informatiques*, Vol.7, No.6 :527-546, 1988.
- [8] P. Boutheymy and E. François. Motion segmentation and qualitative dynamic scene analysis from an image sequence. *Int. Journal of Computer Vision*, Vol.10, No 2 :157-182, Avril 1993.

- [9] P.J. Burt. The pyramid as a structure for efficient computation. In A. Rosenfeld, editor, *Multiresolution Image Processing and Analysis*, pages 6–35, Springer-Verlag, 1984.
- [10] P.J. Burt, R. Hingorani, and R.J. Kolczynski. Mechanisms for isolating component patterns in the sequential analysis of multiple motion. In *IEEE Workshop on Visual Motion*, pages 187–193, Princeton, Octobre 1991.
- [11] T. Darrell and A. Pentland. Robust estimation of a multi-layered motion representation. In *Proc. IEEE Workshop on Visual Motion*, pages 173–178, Princeton, Oct. 1991.
- [12] W. Enkelmann. Investigations of multigrid algorithms for the estimation of optical flow fields in image sequences. *Computer Vision, Graphics and Image Processing*, Vol.43 :150–177, 1988.
- [13] F.R. Hampel, E.M. Ronchetti, P.J. Rousseeuw, and W.A. Stahel. *Robust Statistics : The Approach Based on Influence Functions*. John Wiley and Sons, New York, 1986.
- [14] M. Hoetter. Differential estimation of the global motion parameters zoom and pan. *Signal Processing*, Vol.16 :249–265, 1989.
- [15] P.W. Holland and R.E. Welsch. Robust regression using iteratively reweighted least squares. *Commun. Stat.- Theor. Meth.*, A6 :813–828, 1977.
- [16] B.K.P. Horn and B.G. Schunck. Determining optical flow. *Artificial Intelligence*, Vol.17 :185–203, 1981.
- [17] P.J. Hubert. *Robust statistics*. Wiley, 1981.
- [18] M. Irani, B. Rousso, and S. Peleg. Detecting and tracking multiple moving objects using temporal integration. In *Proc. of 2nd ECCV-92*, S.Margherita Ligure, Italie, pages 282–287, Springer-Verlag, Mai 1992.
- [19] J.M. Jolion, P. Meer, and S. Batauche. Robust clustering with application in computer vision. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13 :791–802, Août 1991.
- [20] J. Konrad and E. Dubois. Multigrid Bayesian estimation of image motion fields using stochastic relaxation. In *Proc. 2nd Int. Conf. Computer Vision*, pages 354–362, Tarpon Springs, Floride, Déc. 1988.
- [21] G.E. Mailloux, F. Langlois, P.L. Simard, and M. Bertrand. Restoration of the velocity field of the heart from two-dimensional echocardiograms. *IEEE Trans. on Medical Imaging*, 11(2) :143–153, Juin 1989.
- [22] P. Meer, D. Mintz, and A. Rosenfeld. Robust regression methods for computer vision : a review. *International Journal of Computer Vision*, 6(1) :59–70, 1991.
- [23] F. Meyer and P. Bouthemy. Estimation of time-to-collision maps from first order motion models and normal flows. In *Proc. 11th Intern. Conf. on Pattern Recognition*, La Haye, pages 78–82, 1992.
- [24] F.G. Meyer and P. Bouthemy. Region-based tracking using affine motion models in long image sequences. *CVGIP : Image Understanding*, Vol. 60(2) :119–140, Septembre 1994.
- [25] H.H. Nagel. From image sequences towards conceptual descriptions. *Image and Vision Computing Jal*, Vol.6, No.2 :pp 59–74, Mai 1988.
- [26] S. Negahdaripour and S. Lee. Motion recovery from image sequences using first-order optical flow information. In *Proc. of the IEEE Workshop on Visual Motion*, Princeton, pages 132–139, Oct. 1991.
- [27] H. Nicolas and C. Labit. Global motion identification for image sequence analysis and coding. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, pages 2825–2828, Toronto, Mai 1991.
- [28] P.J. Rousseeuw. Least median of squares regression. *Jal of the American Statistical Association*, Vol.79, No 388 :871–880, Déc. 1984.
- [29] J. Schmetz and M.S. Mhita. Diurnal and interdiurnal variability of IR and WV brightness temperatures from Meteosat. *ESA Journal*, 13 :329–341, 1989.
- [30] T. Viéville and O. Faugeras. Robust and fast computation of unbiased intensity derivatives in images. In *Proc. of 2nd ECCV-92*, S.Margherita Ligure, Italie, pages 203–212, Mai 1992.
- [31] S.F. Wu and J. Kittler. A gradient-based method for general motion estimation and segmentation. *Jal of Visual Communication and Image Representation*, 4(1) :25–38, Mars 1993.
- [32] W.-Z. Zhao, F.-H. Qi, and T.Y. Young. Dynamic estimation of optical flow field using objective functions. *Image and Vision Computing*, Vol.7, No.4 :259–267, Nov. 1989.

## LES AUTEURS

### Jean-Marc ODOBEZ



Jean-Marc Odohez est ingénieur ENST-Bretagne (1990), Docteur de l'Université de Rennes 1 (1994), mention Traitement du signal et télécommunications. Son travail de thèse a été effectué au sein de l'équipe Temis de l'Irisa/Inria-Rennes. Il a porté sur l'estimation, la détection et la segmentation du mouvement dans des séquences d'images. Ses domaines d'intérêt incluent le traitement d'image, la vision par ordinateur, et plus particulièrement l'exploitation du mouvement dans des séquences d'images. Il est actuellement en séjour post-doctoral à Philadelphie, USA, dans le laboratoire GRASP de l'Université de Pennsylvanie.

### Patrick BOUTHEMY



Patrick Bouthemy est ingénieur Télécom-Paris, Docteur-Ingénieur (1982) et Habilité à Diriger des Recherches (1989) de l'Université de Rennes 1. Il est actuellement Directeur de Recherche Inria à l'Irisa. Ses axes principaux de recherche se situent dans les domaines Image et Perception d'une part, Image et Communication d'autre part, et sont les suivants : modèles statistiques pour l'analyse d'image en particulier modèles markoviens, analyse du mouvement et des déformations, analyse multi-échelle, vision qualitative et vision active. Il est responsable du pôle Vision du GDR-PRC "Communications Homme-Machine".

Manuscrit reçu le 14 septembre 1994.