

## Chapitre 6

# Expérimentations numériques. Perspectives.

### 6.1 Expérimentations numériques

---

L'algorithme a été mis en œuvre avec des ondelettes analytiques. La raison d'un tel choix et leur construction est détaillée en 4.2. Nous décrivons brièvement quelques détails d'implémentation et montrons comme l'algorithme se comporte sur différentes séquences classiques.

**Filtres** Les filtres que nous avons utilisés tout au long de nos expérimentations sont dans l'ensemble les mêmes: des filtres de DESLAURIERS–DUBUC d'ordres divers. Les filtres passe-bas de DESLAURIERS–DUBUC sont

$$\begin{bmatrix} -1 & 0 & 9 & 16 & 9 & 0 & -1 \end{bmatrix} / 16 \quad (6.1a)$$

ou

$$\begin{bmatrix} 3 & 0 & -25 & 0 & 150 & 256 & 150 & 0 & -25 & 0 & 3 \end{bmatrix} / 256 \quad (6.1b)$$

Ces filtres ont un certain nombre d'avantages. Ils ont une bonne résolution fréquentielle (et un bon nombre de moments nuls) par comparaison avec le nombre d'opérations qu'ils requièrent pour une convolution. Ils contournent notamment la limite classique selon laquelle le nombre de moments nuls est borné pour une taille de support fixée en ayant un certain nombre de coefficients intermédiaires *nuls* qui augmentent la taille du support sans coûter des opérations supplémentaires. Par ailleurs les coefficients de ces filtres sont des rationnels dyadiques, et on peut donc éviter des calculs en virgule flottante. Les transformées en ondelettes ont été sur-échantillonnées d'un facteur 2, si bien que la gamme de mesures estimée en (3.9) peut être multipliée par 2.

Les filtres passe-haut correspondants  $m_1$  sont les filtres passe-bas avec un déphasage de  $\pi$ . Le filtre analytique  $m_2$  est obtenu par déphasage du  $m_0$  d'une phase de  $\pi/2$ . Il faut

remarquer que la convolution avec un tel filtre a le même coût que la convolution par un filtre réel  $m_0$ , parce que les coefficients sont soit réels, soit imaginaires purs.

**Les effets de bords** Plusieurs méthodes ont été testées pour réduire l'effet dévastateur sur les mesures de la troncature des images aux bords, prolongement par 0, par une constante (par continuité) et par symétrie. Comme l'a remarqué SIMONCELLI [Sim98], les prolongements continus sont ceux qui font perdre le moins de précision. Pour limiter le nombre des calculs, les prolongements sont effectués à la demande à chaque convolution quand le support d'un filtre sort des limites de l'image.

### 6.1.1 Coût de calcul

Nous avons annoncé que l'algorithme avait un coût de calcul proportionnel au nombre de pixels de l'image. Il reste maintenant à déterminer la constante qui apparaît. Le coût de calcul a été estimé pour deux situations : avec et sans mesures de changements d'illumination. Le coût en flops inclut le préfiltrage, les filtrages en cascade, la résolution des systèmes, le décentrement de gammes de mesures et l'interpolation. Nous obtenons ainsi une carte à une résolution de blocs  $2 \times 2$ . Nous incluons également dans le coût de calcul le coût de raffinement de la carte de flot à la résolution du pixel par interpolation. Des complexités de calculs similaires ont été publiées par MAGAREY et KINGSBURY [MK98], et sont rappelées pour mémoire dans le tableau 6.1. La seule différence est éventuellement que MAGAREY et KINGSBURY ont estimé la complexité pour 5 niveaux de filtrage, alors que notre estimation vaut pour un nombre quelconque de niveaux.

Méthode	complexité
MAGAREY & KINGSBURY	1618
La nôtre (avec illumination)	863
La nôtre (sans illumination)	780

TAB. 6.1 – Complexité en opérations de virgule flottante (FLOPs) par pixel

### 6.1.2 Séquences réelles

Nous avons téléchargé différentes séquences rassemblées par BARRON *et coll.* sur leur site FTP à l'adresse *csd.uwo.ca*. L'algorithme a été testé sur la séquence « rubic » (un cube placé sur un plateau tournant), la séquence « taxi » (où trois véhicules se déplacent vers l'est, l'ouest et le nord-ouest). Une image de la séquence et la carte de flot mesurée sont représentées en Fig. 6.1.2 et 6.1.2.

### 6.1.3 Séquences synthétiques

L'algorithme que nous avons décrit a été essayé sur des séquences synthétiques classiques (dont la séquence « Yosemite »), et les erreurs d'estimation ont été comparées à celles des autres méthodes. L'erreur de mesure est faite comme l'usage le veut en termes d'erreur

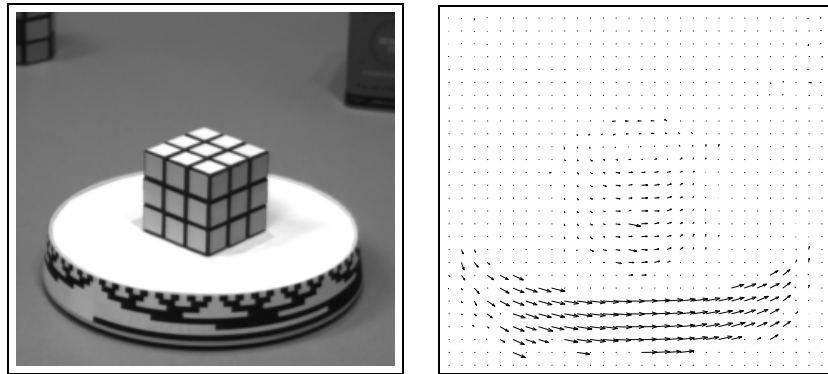


FIG. 6.1 – Séquence « rubic » et flot mesuré

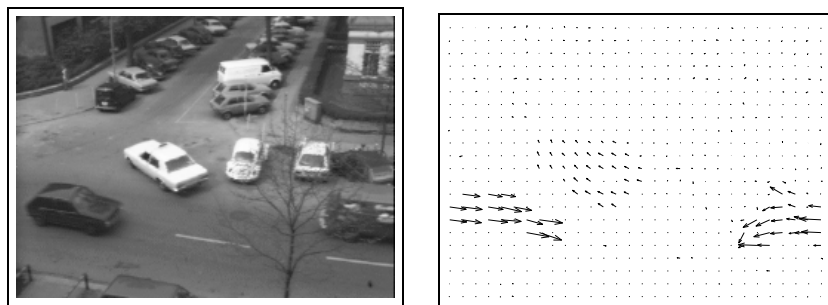


FIG. 6.2 – Séquence « taxi » et flot mesuré

angulaire, ainsi que cela a été proposé par FLEET et JEPSON dans [FJ90]. La « distance » entre une vitesse mesurée  $\tilde{\mathbf{v}} = (\tilde{v}_1, \tilde{v}_2)$  et une vitesse réelle  $\mathbf{v} = (v_1, v_2)$  est l'angle en degrés entre les vecteurs

$$\begin{bmatrix} \tilde{v}_1 \\ \tilde{v}_2 \\ 1 \end{bmatrix} \quad \text{et} \quad \begin{bmatrix} v_1 \\ v_2 \\ 1 \end{bmatrix}$$

soit  $\arccos(1 + \tilde{\mathbf{v}} \cdot \mathbf{v})$ .

Pour la séquences « translating trees », l'erreur angulaire moyenne est  $0.78^\circ$  avec une densité de mesure de 99,3%, qui peut être comparée aux  $1,32^\circ$  de MAGAREY et KINGSBURY. Le tableau 6.2 rappelle les résultats obtenus par d'autres méthodes.

Images	Méthode	Erreur moyenne	Densité
21	FLEET & JEPSON	$0,32^\circ$	74,5%
10	WEBER & MALIK	$0,49^\circ$	96,8%
2	MAGAREY & KINGSBURY	$1,32^\circ$	100,0%
2	Notre méthode (avec illumination)	$0,78^\circ$	99,3%

TAB. 6.2 – *Erreurs angulaires comparées pour la séquence « translating trees ». Dans ce tableau, la première colonne indique combien d'images successives sont nécessaires pour mesurer le flot optique. Il faut remarquer que de ce point de vue, les méthodes les moins coûteuses sont celle de MAGAREY et KINGSBURY et la nôtre.*

Pour la séquence « Yosemite », l'erreur angulaire moyenne sur toute l'image (ce qui inclut le sol et le ciel, moins une bande de 16 pixels par bord) est de  $6,52^\circ$ . La densité de mesure est 96,5%, parce que quelques mesures ont été rejetée à cause de l'*aliasing* temporel. Ce résultat se compare à celui de MAGAREY et KINGSBURY (de  $6,20^\circ$ ). Pour cette séquence, la carte de flot et la carte d'erreur angulaire sont représentées avec la carte de variations d'éclairements mesurés. L'erreur angulaire moyenne est comparées à celle d'autres méthodes en figure 6.3.

D'autres méthodes [WKCL98, MP98] permettent d'atteindre des erreurs d'estimation plus faibles, au prix d'une complexité de calcul sensiblement plus élevée. MÉMIN et PÉREZ [MP98] obtiennent une erreur moyenne de  $5.38^\circ$  pour la séquence *Yosemite*, et WU *et coll.* annoncent des erreurs également inférieures, mais ces méthodes consistent à minimiser des fonctionnelles non convexes par relaxation.

Images	Méthode	Erreur moyenne	Densité
21	FLEET & JEPSON	$4,63^\circ$	34,1%
10	WEBER & MALIK	$4,31^\circ$	64,2%
2	MAGAREY & KINGSBURY	$6,20^\circ$	100,0%
2	Notre méthode (avec illumination)	$6,50^\circ$	96,5%

TAB. 6.3 – *Erreurs comparées pour les différentes méthodes de mesure pour la séquence « Yosemite ». Voir les remarques faites dans la légende du tableau 6.2.*

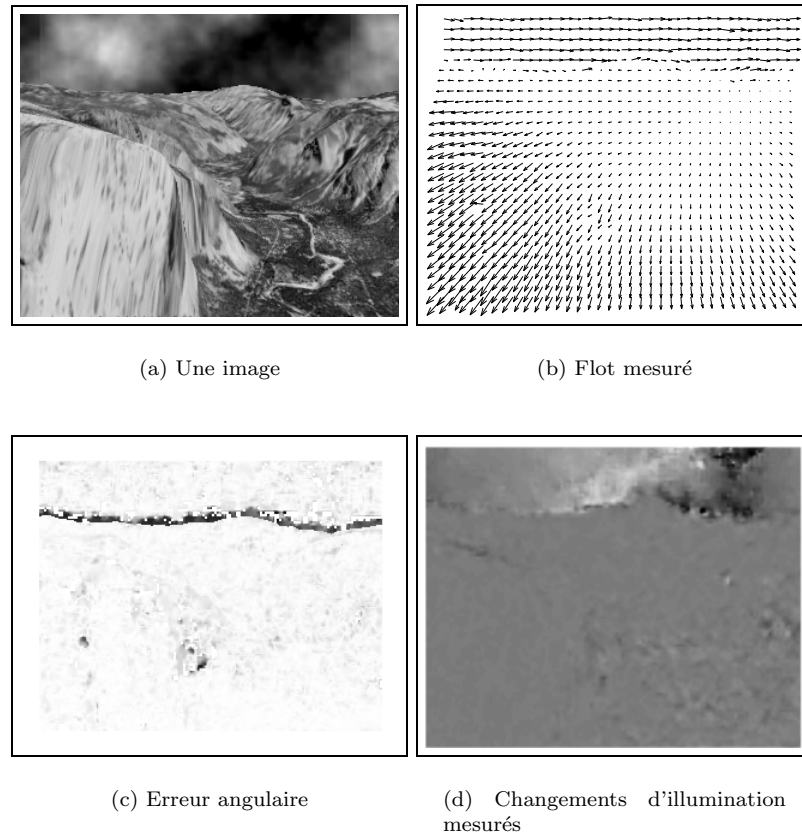


FIG. 6.3 – Séquence « Yosemite ». Nous voyons notamment que l'erreur angulaire est importante autour de l'horizon, où se situe une discontinuité du flot optique. La carte des changements d'illumination indique les zones où l'illumination est constante (en gris), où elle augmente (en blanc) et où elle diminue (en noir). L'algorithme détecte que le côté droit du nuage de gauche s'éclaircit, tandis que le côté gauche du nuage de droite s'assombrit, ce qui correspond l'impression que laisse la séquence quand on la regarde. Nous allons valider notre mesure d'illumination sur une autre séquence synthétisée.

### 6.1.4 Changement d'illumination

Nous utilisons l'équation du flot optique définie en 4. L'estimation du flot optique et des changements d'illumination est faite en utilisant une cinquième ondelette d'intégrale non nulle :

$$\Psi^0(x_1, x_2) = \phi(x_1)\phi(x_2) \quad (6.2)$$

Jusque là, toutes les ondelettes  $\Psi^1, \dots, \Psi^4$  décrites en (4.25a–d) que nous avons utilisées pour mesurer le flot optique ont une intégrale nulle. Ainsi, les mesures de déplacements ne sont en théorie pas trop perturbées par les changements d'illumination. Comme nous voulons maintenant mesurer ces changements d'illumination, il nous faut au moins une ondelette d'intégrale non nulle qui puisse « voir » les changements globaux de niveaux de gris. Tous les systèmes linéaires ont également une troisième inconnue  $L'/L$  et une contrainte linéaire réelle supplémentaire obtenue par produit scalaire de l'équation du flot optique avec  $\Psi^0$ .

Nous allons voir que cette mesure augmente l'insensibilité de notre approche aux changements d'illumination. Notre algorithme est insensible aux changements d'illumination additifs par construction (de même que ceux de MAGAREY et KINGSBURY et d'autres auteurs). Nous avons en revanche mesuré la sensibilité de l'algorithme à des changements multiplicatifs d'illumination. Pour cela, nous avons mesuré le déplacement entre les images 20 et 21 de la séquence « translating trees » où la deuxième image est multipliée par un facteur variable dans l'intervalle  $[0,5; 1,5]$ . Les résultats sont indiqués dans le tableau 6.4. Ils sont très bons comparés aux méthodes classiques (dont des résultats sont rapportés par MAGAREY et KINGSBURY) où des facteurs d'illuminations de 0,95 ou 1,05 suffisent à multiplier l'erreur par 10. Ils sont également très bons comparés à ceux de la méthode de MAGAREY et KINGSBURY qui n'estiment la robustesse de leur méthode que dans l'intervalle  $[0,95; 1,05]$ .

Nous avons comparé le changement moyen d'illumination mesuré avec la valeur réelle  $\Delta L/L$ . Pour des variations importantes, la vraie valeur est en fait

$$2 \frac{L(t+1) - L(t)}{L(t) + L(t+1)}$$

où  $t$  et  $t+1$  sont les temps des deux images successives sur lesquelles la mesure est faite. Ceci provient du fait que notre algorithme prend comme référence l'image qui se situe au temps  $t = 1/2$  et qui est obtenue par interpolation temporelle

$$I(t+1/2) \simeq \frac{I(t) + I(t+1)}{2}$$

comme cela est expliqué en détail dans la section 3.2 sur l'*aliasing* temporel.

À titre d'exemple, pour une multiplication de  $I(t+1)$  par un facteur de 0,5, nous nous attendons à avoir un facteur de changement d'illumination de  $2 \times (0,5 - 1)/(1 + 0,5) \simeq -0,67$ . Dans la table 6.4, nous voyons que ce terme est donc bien estimé sans biais.

Un deuxième essai a été effectué sur la même séquence « translating tree ». L'image 20 est inchangée, tandis que l'image 21 est multipliée par un profil gaussien

$$g(x, y) = 1 + e^{-\frac{(x-x_0)^2 + (y-y_0)^2}{2\sigma^2}} \quad (6.3)$$

Facteur d'éclairement $s$	0,5	0,7	0,9	1,0	1,1	1,3	1,5
Erreur angulaire	1,33°	0,84°	0,76°	0,78°	0,79°	0,84°	0,93°
$\Delta L/L$ réel $2(s-1)/(1+s)$	-0,67	-0,35	-0,105	0	0,095	0,26	0,4
$\Delta L/L$ mesuré	-0,67	-0,35	-0,105	0,00	0,095	0,26	0,40

TAB. 6.4 – *Erreur de mesure angulaire, changements d'illumination réels et mesurés (en moyenne sur l'image entière). Il faut remarquer que l'algorithme supporte des variations d'éclairement très violentes.*

dont le centre  $(x_0, y_0)$  est  $(75, 75)$  et  $\sigma = 75/2$ . Rappelons que les dimensions de l'image des « translating trees » sont  $150 \times 150$  pixels. Le profil de changement d'illumination est donc censé être :

$$\frac{L'}{L} \simeq 2 \frac{g(x, y) - 1}{1 + g(x, y)}$$

dont le maximum est  $2/3$ . Les deux images successives obtenues sont représentées en Fig. 6.4.



FIG. 6.4 – *Deux images successives de la séquence « translating trees » modifiée pour avoir un changement d'illumination artificiel. La deuxième image est multipliée par un profil gaussien dont la formule est donnée en (6.3).*

Le flot optique et les changements d'illumination ont été mesurés entre ces deux images. L'erreur angulaire moyenne pour l'estimation du flot optique est de  $2,45^\circ$ . L'erreur moyenne d'estimation de changements d'illumination a été estimée de la manière suivante : si on note  $\lambda$  l'illumination mesurée qui est censée estimer  $L'/L$ , on mesure l'erreur moyenne en norme  $L_1$

$$E = \frac{1}{NM} \sum_{i=1}^N \sum_{j=1}^M \left| \lambda(i, j) - \frac{L'(i, j)}{L(i, j)} \right|$$

Cette erreur de mesure était donc de 0,0255. Si on la ramène à un maximum de 0,67, cela

fait une erreur relative de 3,8%. La carte d'illumination théorique et la carte mesurée sont présentées sur la figure 6.5.

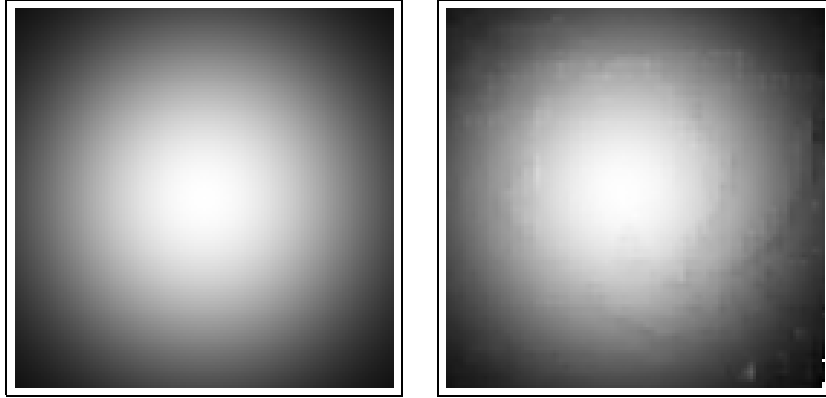


FIG. 6.5 – Carte de changement d'illumination pour la séquence « *translating trees* » modifiée. À gauche, la carte synthétique. À droite, la carte mesurée.

Nous pensons qu'en plus d'augmenter la robustesse de la mesure du flot optique vis-à-vis de changements d'illuminations, la mesure de ce paramètre supplémentaire pour être utilisée en codage de séquences vidéo. Pour de raisons diverses, l'éclairement d'une scène peut changer au cours de la prise de vue. Ceci peut être causé par des mouvements de sources, ou simplement dû au mouvement de la caméra. Si la caméra se déplace d'une zone sombre à une zone claire, le système d'équilibrage des lumières va changer la réponse du capteur optique, et ainsi changer la luminosité apparente d'un objet fixé de la scène.

## 6.2 Compression vidéo

La mesure du flot optique est un moyen de capturer la redondance temporelle entre des images successives d'une même séquence vidéo, et de réduire la taille du code nécessaire pour la représenter. Dans cet esprit, on peut proposer une première approche simple de la compression vidéo par compensation de mouvement.

On note la séquence vidéo  $I(t; \mathbf{x})$  où  $t \in \mathbb{N}$ . Un codage simple consiste à coder les images une par une par un schéma de compression d'images statiques (JPEG, SPIHT, etc.). Une manière élémentaire de tirer parti de la redondance temporelle de la séquence d'images est de coder une première image  $I(0)$ , puis de coder ensuite seulement les différences entre images successives  $I(1) - I(0)$ ,  $I(2) - I(1)$ , etc. Cette stratégie est intéressante quand une grande partie de l'image est immobile (des personnages qui se déplacent sur un fond fixe), mais est mise en défaut dès qu'une trop grande partie de la scène est mobile (par exemple à cause d'un mouvement de la caméra, comme un *travelling* ou un *zoom*).

La deuxième approche consiste à effectuer une compensation de mouvement. On mesure le déplacement  $\mathbf{v}(t + 1/2)$  entre les images  $I(t)$  et  $I(t + 1)$ . Si la mesure est précise et si le



flot suffit à expliquer les modifications de l'image, on peut espérer retrouver l'image  $I(t+1)$  par déplacement (en angl. *warping*) de l'image  $I(t)$  le long du champ de vecteurs  $\mathbf{v}(t+1/2)$ .

Si on note

$$\mathcal{T}_{\mathbf{v}}(f)(\mathbf{x}) = f(\mathbf{x} - \mathbf{v}(\mathbf{x}))$$

on pourra espérer que l'image au temps  $t$  déplacée le long du flot  $\mathbf{v}(t+1/2)$  sera proche de  $I(t+1)$ , c'est à dire

$$I(t+1) \simeq \mathcal{T}_{\mathbf{v}(t+1/2)}(I(t))$$

En général, il reste une erreur de prédiction  $E(t)$  qui est définie par

$$E(t) = I(t+1) - \mathcal{T}_{\mathbf{v}(t+1/2)}(I(t))$$

Le principe du codage de l'image par compensation de mouvement est donc le suivant :

- On transmet la première image  $I(0)$ .
- On transmet pour chaque image qui suit  $\mathbf{v}(t+1/2)$  et  $E(t)$ , qui permettent au décodeur de reconstituer l'image  $I(t+1)$  selon la formule

$$I(t+1) = \mathcal{T}_{\mathbf{v}(t+1/2)}(I(t)) + E(t)$$

Ce principe parie sur le fait qu'il est plus efficace de coder  $\mathbf{v}(t+1/2)$  et  $E(t)$  plutôt que de coder simplement  $I(t+1) - I(t)$ . Cela se conçoit, car idéalement  $E(t) = 0$ , et le champ  $\mathbf{v}$  est plus régulier que la dérivée  $I(t+1) - I(t)$  de la texture des objets qui se déplacent. Les erreurs de prédiction sont représentées pour deux séquences d'images dans la figure 6.6.

Ce principe a été mis en œuvre par deux binômes d'élèves sur la base de l'algorithme de mesure du déplacement, Paul FROMENT et Rodolphe GINTZ [FG99], puis Hervé DELAY et Adrien VEILLARD [DV99]. Dans les deux cas, ils ont mis au point un algorithme complet de compression et de décompression de séquences d'images vidéo qui soutiennent bien la comparaison avec des codeurs téléchargeables librement sur le réseau *internet*, ce qui est très honorable pour de premières implémentations.

## 6.3 Introduction de l'éclairement comme variable explicative supplémentaire

---

Quand l'éclairement d'une scène change, les modifications de l'image ne peuvent pas être décrites en termes de déplacement d'objets. La variation d'illumination doit donc intégralement prise en compte par dans l'image d'erreur  $E(t)$ . Si le profil d'éclairement  $g(t, \mathbf{x})$  est régulier (comme en (6.3)), l'image d'erreur sera de la forme

$$I(t) \times g(t) \tag{6.4}$$

c'est à dire qu'elle héritera de la complexité de l'image d'origine. Si nous ajoutons la variable explicative « éclairement » à notre modèle, nous aurons à coder le profil  $g(\mathbf{x})$  que nous aurons mesuré, et l'erreur de prédiction ne contiendra plus de terme d'erreur (6.4) à coder.

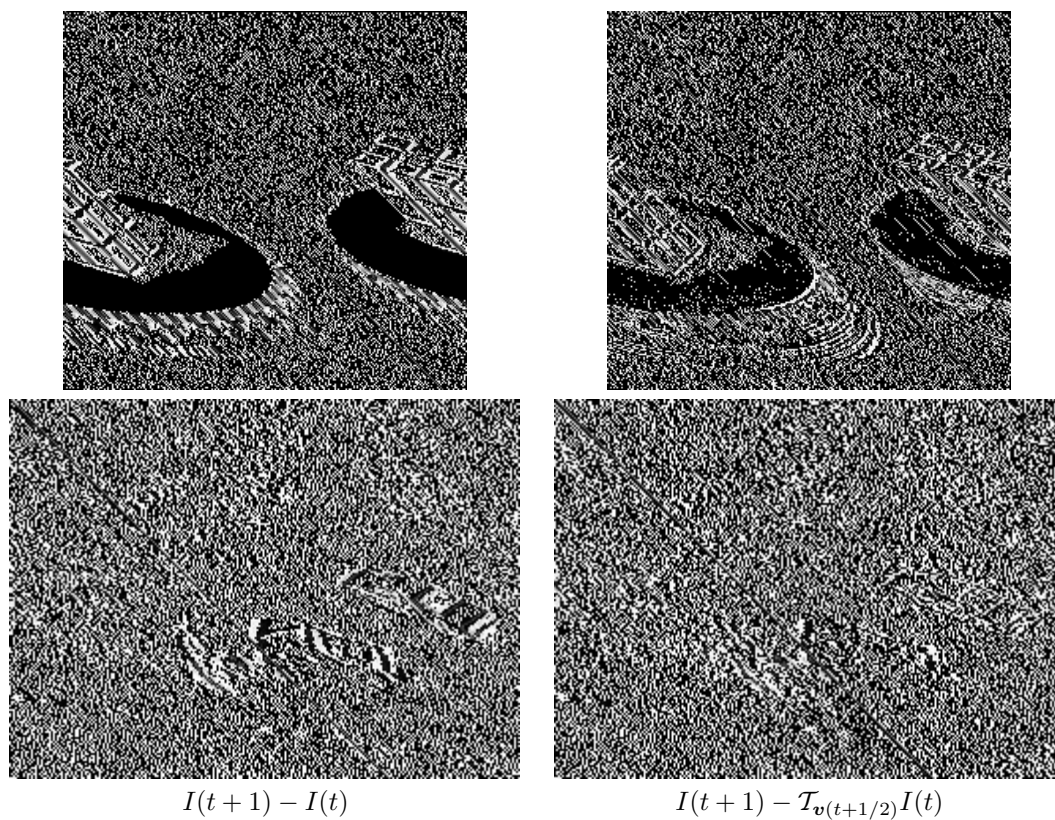


FIG. 6.6 – Pour les séquences « rubic » (en haut) et « taxi » (en bas), nous voyons les erreurs de prédiction sans compensation de mouvement (à gauche) et avec compensation de mouvement (à droite). Le contraste a été augmenté (avec une saturation forte des images de gauche) pour que les images soient comparables.

Ainsi, en ajoutant la variable d'éclairement dans le schéma de compensation, nous remplaçons le codage de  $I(t) \times g(t)$  par le seul codage de  $g(t)$  dont nous espérons qu'il soit moins coûteux.

Le nouveau modèle de prédiction de mouvement et d'illumination est donc le suivant. On mesure un déplacement  $\mathbf{v}(\mathbf{x}, t + 1/2)$  et un champ d'éclairement  $\lambda(\mathbf{x}, t + 1/2)$ . On prédit alors l'image  $I(t + 1)$  en fonction de l'image  $I(t)$  selon la formule

$$I(t + 1) = (1 + \lambda(t)) \times \mathcal{T}_{\mathbf{v}(t+1/2)} I(t)$$

d'où le principe de codage par compensation de mouvement *et de changements d'illuminations* :

- On transmet la première image  $I(0)$
- On transmet pour chaque image qui suit  $\mathbf{v}(t + 1/2)$ ,  $\lambda(t + 1/2)$  et  $E(t)$  au décodeur. Ce dernier peut reconstituer l'image  $I(t + 1)$  en fonction de  $I(t)$  selon la formule :

$$I(t + 1) = (1 + \lambda(t)) \times \mathcal{T}_{\mathbf{v}(t+1/2)}(I(t)) + E(t)$$

#### Remarque

*En principe, la compensation de changement d'éclairements devrait se faire non pas selon la formule*

$$I \leftarrow I \times (1 + \lambda)$$

*mais selon la formule plus précise*

$$I \leftarrow I \times \left( \frac{1 + \lambda/2}{1 - \lambda/2} \right)$$

*parce que le changement d'éclairement est supposé être mesuré en  $t + 1/2$  (cf. Sec. 6.1.4).*

## 6.4 Modèles non constants du flot optique

Une extension possible de l'algorithme présenté dans ces pages consiste à utiliser un modèle moins simple du flot qu'un modèle de flot localement constant. Deux modèles peuvent à la fois être motivés par des considérations géométriques et des modèles tridimensionnels sous-jacents [Kan89], tout en étant solubles dans le cadre présenté ci-dessus.

### 6.4.1 Modèle de projection stéréographique d'éléments plans

Considérons un modèle de caméra à projection stéréographique, c'est à dire qu'un point de l'espace  $(x, y, z)$  se projette sur le plan du film au point de coordonnées  $(X, Y)$  selon la formule

$$\begin{aligned} X &= \frac{x}{z} \\ Y &= \frac{y}{z} \end{aligned}$$

Considérons un élément plan dans l'espace. Il peut être caractérisé par une équation cartésienne. Nous ferons l'hypothèse que ce plan n'est pas « vu par la tranche », autrement dit qu'il ne passe pas par l'origine. Son équation cartésienne peut alors être écrite :

$$ax + by + cz = 1 \quad (\text{II})$$

La projection entre le plan image et ce plan peut être inversée. On obtient aisément  $x$ ,  $y$  et  $z$  en fonction de  $X$  et  $Y$  quand on suppose que le point vu en  $(X, Y)$  est l'image d'un point du plan (II).

$$\begin{aligned} x &= \frac{X}{aX + bY + c} \\ y &= \frac{Y}{aX + bY + c} \\ z &= \frac{1}{aX + bY + c} \end{aligned}$$

Si on considère maintenant que la facette considérée incluse dans le plan (II) a un mouvement solide, le vecteur vitesse en un point de coordonnées  $(x, y, z)$  s'écrit  $\mathbf{v} = \mathbf{v}_0 + \boldsymbol{\omega} \wedge \mathbf{OM}$ , soit

$$\begin{aligned} \dot{x} &= \dot{x}_0 + \omega_y z - \omega_z y \\ \dot{y} &= \dot{y}_0 + \omega_z x - \omega_x z \\ \dot{z} &= \dot{z}_0 + \omega_x y - \omega_y x \end{aligned}$$

On peut donc en déduire le flot optique en fonction du mouvement réel

$$\begin{aligned} \dot{X} &= \frac{\dot{x}}{z} - \dot{z} \frac{x}{z^2} \\ \dot{Y} &= \frac{\dot{y}}{z} - \dot{z} \frac{y}{z^2} \end{aligned}$$

et tenant compte du mouvement solide, on tire

$$\begin{aligned} \dot{X} &= \frac{\dot{x}_0 + \omega_y z - \omega_z y}{z} - \dot{z} \frac{x_0 + \omega_x y - \omega_y x}{z^2} \\ &= \frac{x_0}{z} + \omega_y - \omega_z Y - \dot{z}_0 \frac{X}{z} - \omega_x XY + \omega_y X^2 \end{aligned}$$

et d'après l'hypothèse de planéité :

$$\begin{aligned} \dot{X} &= x_0(aX + bY + c) + \omega_y - \omega_z Y - \dot{z}_0 X(aX + bY + c) - \omega_x XY + \omega_y X^2 \\ &= X^2(\omega_y - a\dot{z}_0) + XY(-\omega_x - b\dot{z}_0) \\ &\quad + X(a\dot{x}_0 - c\dot{z}_0) + Y(b\dot{x}_0 - \omega_z) + c\dot{x}_0 + \omega_y \end{aligned}$$

Par un calcul identique, on obtient :

$$\begin{aligned} \dot{Y} &= XY(-a\dot{z}_0 + \omega_y) + Y^2(-b\dot{z}_0 - \omega_x) \\ &\quad + X(a\dot{y}_0 + \omega_y) + Y(-b\dot{y}_0 - c\dot{z}_0) + c\dot{y}_0 - \omega_x \end{aligned}$$

### 6.4.2 Cas d'un modèle de caméra à projection orthogonale

Dans le cas d'une projection orthogonale (ou  $X$  et  $Y$  sont petits), on obtient les mêmes formules sans les termes de degré 2, soit

$$\begin{aligned}\dot{X} &= X(a\dot{x}_0 - c\dot{z}_0) + Y(b\dot{x}_0 - \omega_z) + c\dot{x}_0 + \omega_y \\ \dot{Y} &= X(a\dot{y}_0 + \omega_y) + Y(-b\dot{y}_0 - c\dot{z}_0) + c\dot{y}_0 - \omega_x\end{aligned}$$

### 6.4.3 Estimation d'un flot non localement constant avec des ondelettes

Nous considérons simplement le modèle de flot linéaire, qui correspond à une projection orthogonale d'un objet plan de mouvement solide sur le plan de la caméra, bien que les calculs ci-dessous puissent s'étendre dans le même esprit au cas plus général de la projection stéréographique.

Le modèle de flot obtenu est donc de la forme

$$\begin{aligned}v_x &= ax + by + c \\ v_y &= dx + ey + f\end{aligned}$$

où l'identification doit porter sur les coefficients  $a, b, c, d, e$  et  $f$ . L'hypothèse que le flot optique est constant sur le support des ondelettes est ici remplacée par une hypothèse selon laquelle ce modèle de dépendance linéaire du flot en fonction de la position est vrai sur le support des ondelettes  $\psi^n = \psi_{j\mathbf{k}}^n$  utilisées. La projection de l'équation du flot optique s'écrit donc

$$\left\langle \frac{\partial I}{\partial x} v_x, \psi^n \right\rangle + \left\langle \frac{\partial I}{\partial y} v_y, \psi^n \right\rangle + \left\langle \frac{\partial I}{\partial t}, \psi^n \right\rangle = 0$$

soit par application du modèle

$$\left\langle \frac{\partial I}{\partial x} (ax + by + c), \psi^n \right\rangle + \left\langle \frac{\partial I}{\partial y} (dx + ey + f), \psi^n \right\rangle + \left\langle \frac{\partial I}{\partial t}, \psi^n \right\rangle = 0$$

et après des intégrations par parties

$$\begin{aligned}a \left( \left\langle xI, \frac{\partial \psi^n}{\partial x} \right\rangle + \langle I, \psi^n \rangle \right) + b \left\langle yI, \frac{\partial \psi^n}{\partial y} \right\rangle + c \left\langle I, \frac{\partial \psi^n}{\partial x} \right\rangle \\ + d \left\langle xI, \frac{\partial \psi^n}{\partial x} \right\rangle + e \left( \left\langle yI, \frac{\partial \psi^n}{\partial y} \right\rangle + \langle I, \psi^n \rangle \right) + f \left\langle I, \frac{\partial \psi^n}{\partial y} \right\rangle = \left\langle \frac{\partial I}{\partial t}, \psi^n \right\rangle\end{aligned}$$

Ainsi, les coefficients de  $a$  à  $f$  sont solutions de  $N$  contraintes linéaires dont les coefficients sont des coefficients d'ondelettes de  $I, xI$  et  $yI$ . On peut ainsi effectuer des calculs rapides de ces coefficients (en  $\mathcal{O}(N)$  opérations) de la même manière que le flot optique était calculé rapidement ci-dessus.

#### Remarque

*Le lecteur peut s'imaginer que dans le cas d'un modèle de flot localement quadratique, nous pouvons obtenir un résultat similaire, à savoir que les coefficients des systèmes linéaires à résoudre seront cette fois des coefficients d'ondelettes des images  $I, xI, yI, x^2I, y^2I$  et  $xyI$ .*

Ces modèles plus riches peuvent avoir plusieurs usages, et ont certains avantages sur le modèle simple de flot localement constant. Comme cela a été mentionné en 4.1.3, un tel modèle peut être utilisé pour faire de la segmentation de flot optique, ainsi que l'ont démontré ODOBEZ et BOUTHEMY[OB98]. Un modèle trop simple détecte des singularités du flot optique également là où le flot varie fortement, comme à l'intérieur d'une surface plane qui a un mouvement de zoom par rapport à la caméra, et n'est donc pas approprié.

Un tel modèle peut aussi augmenter la fiabilité d'une mesure: MENDELSON et coll. [MSB97], avec un modèle local plan d'objet (en projection stéréographique) augmentent la fiabilité de leurs mesures, avec un gain appréciable pour les déplacements importants. Une autre application est la reconstruction de scènes tridimensionnelles en imposant une certaine régularité locale aux objets qui sont reconstitués.

## Bibliographie

- [BF93] P. BOUTHEMY et E. FRANÇOIS. Motion segmentation and qualitative dynamic scene analysis from an image sequence. *Int. Journal of Computer Vision*, 10(2):157–182, 1993.
- [BFB94] J. BARRON, D. FLEET et S. BEAUCHEMIN. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77, 1994.
- [DV99] H. DELAY et A. VEILLARD. Compression de séquences d’images par une méthode de flot. Rapport technique, École Polytechnique, France, École Polytechnique Fédérale de Lausanne, mars 1999.
- [FG99] P. FROMENT et R. GINTZ. Mesure du flot optique et compression vidéo. Rapport technique, Études d’Approfondissement, École Polytechnique, France, janvier 1999.
- [FJ90] D. FLEET et A. JEPSON. Computation of component image velocity from local phase information. *International Journal of Computer Vision*, 5:77–104, 1990.
- [Kan89] K. KANATANI. *Group-theoretical methods in image understanding*. Springer Series in Information Sciences. Springer, 1989.
- [MK98] J. MAGAREY et N. KINGSBURY. Motion estimation using a complex-valued wavelet transform. *IEEE Trans. on Signal Processing*, 46(4):1069–1084, avril 1998.
- [MP98] E. MÉMIN et P. PÉREZ. Dense estimation and object-based segmentation of the optical flow with robust techniques. *IEEE Trans. Image Processing*, 7(5):703–719, mai 1998.
- [MSB97] J. MENDELSON, E. SIMONCELLI et R. BAJCSY. Discrete-time rigidity-constrained optical flow. Dans *7th Int’l Conf Computer Analysis of Images and Patterns, Kiel, Germany*. septembre 1997.
- [OB98] J.-M. ODOBEZ et P. BOUTHEMY. Direct incremental model-based image motion segmentation for video analysis. *Signal Processing*, 6(2):143–155, 1998.
- [Sim98] E. P. SIMONCELLI. Bayesian multi-scale differential optical flow. Dans H. JÄHNE et GEISSLER, rédacteurs, *Handbook of computer vision and applications*. Academic Press, 1998.
- [WKCL98] Y.-T. WU, T. KANADE, J. COHN et C.-C. LI. Optical flow estimation using wavelet motion model. Dans *Sixth International Conference on Computer Vision*, pages 992–998. Narosa Publishing House, 1998.

- [WM95] J. WEBER et J. MALIK. Robust computation of optical flow in a multi-scale differential framework. *International Journal of Computer Vision*, 14(1):5–19, 1995.