DIRO
IFT 2425

DEVOIR N° 2

Méthode de Jacobi pour l'estimation du Flot Optique

Max Mignotte

DIRO, Département d'Informatique et de Recherche Opérationelle.
http : //www.iro.umontreal.ca/∼mignotte/IFT2425/
E-mail : mignotte@iro.umontreal.ca

Estimation du Flot Optique

En vision par ordinateur, l'estimation du flot optique consiste à estimer, à partir d'une séquence d'images[1], le mouvement apparent des objets composant une scène tridimensionnelle (ou plus formellement, à recouvrer la projection des différents mouvements tridimensionnels de la scène sur le plan image). Concrètement, pour une image $I(t)$ extraite d'une séquence d'image $\{I(t=0), I(t=1), \ldots, I(t), I(t+1), \ldots\}$. Le calcul du mouvement apparent consiste à associer à chaque pixel (x, y) de cette image, un vecteur (v_x, v_y) représentant la vitesse apparente du pixel estimé à cet instant t.

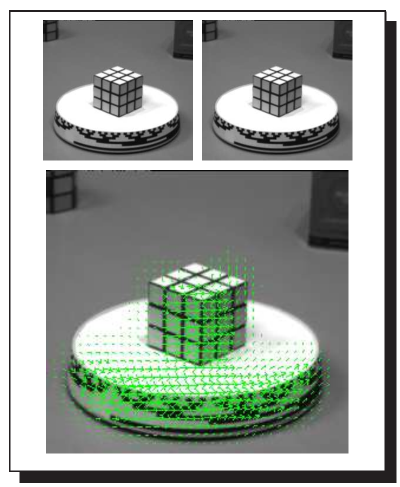
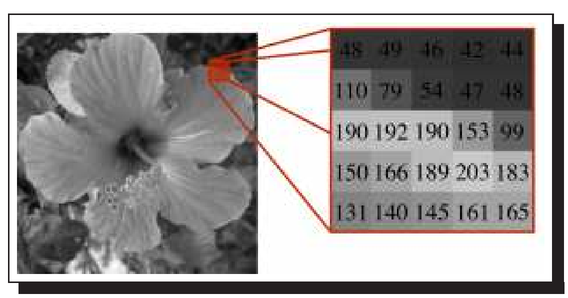


Fig. 1 –
Exemple de calcul du flot optique

Ce mouvement apparent est estimé uniquement à partir des changements de la distribution spatiale des intensités lumineuses (ou niveaux de gris des pixels) existant entre deux images consécutives de la séquence. Ce mouvement est dit appelé "apparent" car il ne correspond pas nécessairement à la réalité. Dans le cas, où il existe un mouvement mais aussi aucune variation d'intensité existant dans les deux images (cas d'une sphère de couleur homogène, non texturée, tournant sur elle -même), les vitesses apparentes seront estimées comme étant nulle ! C'est ce qui distingue le calcul du flot optique (mouvement apparent reflété par une variation d'intensité lumineuse) et le mouvement apparent réel. Dans le cas, où les objets de la scène possèdent des bords et quelques textures, le mouvement apparent réel peut raisonnablement être estimé par le flot optique, *i.e.*, sur la base des variations de la fonction de luminance.

[1] Une image numérique (ou discrétisée) est simplement définie par une mosaique de petits carrés appelés *pixel* dont le niveaux de gris (pour une image en noir et blanc) est codé linéairement entre 0 et 255 avec la convention suivante ; 0 désigne le noir et 255 le blanc. Une image numérique de taille 5 lignes et 5 colonnes définit donc une image avec 25 pixels dont leur niveau de gris est résumé dans une structure de données qui est simplement un tableau deux dimensions de niveaux de gris (cf. Fig. à droite). Le squelette de programme que je vous donne sur ma page web chargera les deux images de la Fig. 1 dans deux tableaux 2Ds à utiliser dans la méthode de résolution par Jacobi qui vous est demandé dans la suite du TP.



Méthode de Horn & Shunck

La méthode de Horn & Shunck repose sur l'hypothèse principale que le flot optique peut être déduit si l'on suppose que l'intensité de chaque point de l'objet reste constante le long de sa trajectoire, ce qui revient à écrire :

$$I(x + \delta x, y + \delta y, t + \delta t) = I(x, y, t) \tag{1}$$

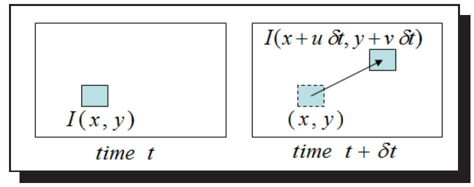


Fig. 2 – Hypothèse d'invariance de l'intensité lumineuse (cf. Eq. (1))

En utilisant l'hypothèse de petits déplacements entre les deux images (hypothèse 2) et en appliquant le développement en série de Taylor du premier ordre au membre de gauche de l'équation (1), on obtient :

$$\begin{aligned}
I(x,y,t) + \delta x \cdot \frac{\partial I}{\partial x} + \delta y \cdot \frac{\partial I}{\partial y} + \delta t \cdot \frac{\partial I}{\partial t} &= I(x,y,t) \\
\frac{\delta x}{\delta t}\frac{\partial I}{\partial x} + \frac{\delta y}{\delta t}\frac{\partial I}{\partial y} + \frac{\partial I}{\partial t} &= 0 \\
v_x\, I_x + v_y\, I_y + I_t &= 0 \qquad (2)
\end{aligned}$$

On obtient donc finalement, pour chaque pixel, l'équation de la *contrainte de l'équation du flot optique*, avec $v_x = \delta x/\delta t, v_y = \delta y/\delta t$; les dérivées spatiales de l'intensité, représentant la vitesse apparente du pixel estimé à l'instant t (ou composante du flot optique selon x et y) et $(\partial I/\partial x, \partial I/\partial y, \partial I/\partial t)$ désignent les dérivées spatiales I_x, I_y et temporelles I_t de l'intensité I de l'image.

L'équation (2) montre que, pour chaque pixel, il n'existe qu'une seule équation à deux inconnus v_x, v_y. Ce problème (connu sous le nom de *problème d'ouverture*) est donc mal posé et il existe, à ce stade, une infinité de solutions au problème de l'estimation du flot optique. Une contrainte (ou hypothèse supplémentaire) est donc requise pour solutionner ce problème ; ce sera l'hypothèse d'homogénéité du flot optique. On suppose que des pixels voisins ont un champ de déplacement (ou flot optique) similaire. Une façon d'exprimer cette contrainte est de dire, que pour chaque pixel de coordonnée (x, y), on cherchera à minimiser l'amplitude au carré du gradient du flot optique *via* la fonction suivante :

$$\left(\frac{\partial v_x}{\partial x}\right)^2 + \left(\frac{\partial v_x}{\partial y}\right)^2 + \left(\frac{\partial v_y}{\partial x}\right)^2 + \left(\frac{\partial v_y}{\partial y}\right)^2 = \|\nabla v_x\|^2 + \|\nabla v_y\|^2 \tag{3}$$

Avec cette dernière contrainte, le but est d'estimer $v = (v_x, v_y)$ qui minimise, pour chaque pixel, l'équation (2) & (3) :

$$\begin{cases}
\mathcal{E}_b = v_x\, I_x + v_y\, I_y + I_t \\
\mathcal{E}_c^2 = \alpha^2 \Big(\|\nabla v_x\|^2 + \|\nabla v_y\|^2\Big)
\end{cases}$$

Le facteur α^2 dans $\mathcal{E}_c$ permet de mettre plus ou moins d'importance dans la troisième hypothèse, à savoir le souhait de trouver un flot optique plus ou moins homogène. Pour l'ensemble des pixels de coordonnée (x, y) de l'image, l'erreur totale à minimiser est donc :

$$\mathcal{E}^2 = \iint \left(\mathcal{E}_c^2 + \mathcal{E}_b^2\right) dx\, dy$$

En utilisant le calcul des variations[2], on obtient le système :

[2] ou équations d'Euler-Lagrange sur : $\mathcal{E}^2 = \iint \overbrace{\left[(v_x\, I_x + v_y\, I_y + I_t)^2 + \alpha^2(\|\nabla v_x\|^2 + \|\nabla v_y\|^2)\right]}^{L} dx\, dy$ avec $v_{y,x} = \partial v_y/\partial x$,

le minimum de cette intégrale minimise aussi le système [2] : $\begin{cases} \frac{\partial L}{\partial v_x} - \frac{\partial}{\partial x}\frac{\partial L}{\partial v_{x,x}} - \frac{\partial}{\partial y}\frac{\partial L}{\partial v_{x,y}} = 0 \\ \frac{\partial L}{\partial v_y} - \frac{\partial}{\partial x}\frac{\partial L}{\partial v_{y,x}} - \frac{\partial}{\partial y}\frac{\partial L}{\partial v_{y,y}} = 0 \end{cases}$

$$\begin{cases} I_x^2 v_x + I_x I_y v_y = \alpha^2 \nabla^2 v_x - I_x I_t \\ \\ I_x I_y v_x + I_y^2 v_y = \alpha^2 \nabla^2 v_y - I_y I_t \end{cases}$$

En utilisant l'approximation du Laplacien $\nabla^2 v = \overline{v} - v$ avec $\overline{v}$ le vecteur v moyen calculé autour d'un voisinage local de pixel (cf. Eq. (5)), on obtient finalement :

$$\begin{cases} (\alpha^2 + I_x^2) v_x + I_x I_y v_y = (\alpha^2\, \overline{v}_x - I_x I_t) \\ \\ I_x I_y v_x + (\alpha^2 + I_y^2) v_y = (\alpha^2\, \overline{v}_y - I_y I_t) \end{cases}$$

En ré-arrangeant le système précédent, on obtient :

$$\begin{cases} (\alpha^2 + I_x^2 + I_y^2) v_x = (\alpha^2 + I_y^2)\overline{v}_x - I_x I_y \overline{v}_y - I_x I_t \\ \\ (\alpha^2 + I_x^2 + I_y^2) v_y = -I_x I_y \overline{v}_x + (\alpha^2 + I_x^2)\overline{v}_y + (\alpha^2 + I_x^2) - I_y I_t \end{cases}$$

ou encore :

$$\begin{cases} (\alpha^2 + I_x^2 + I_y^2)(v_x - \overline{v}_x) = -I_x \left[+I_x \overline{v}_x + I_y \overline{v}_y + I_t \right] \\ \\ (\alpha^2 + I_x^2 + I_y^2)(v_y - \overline{v}_y) = -I_y \left[+I_x \overline{v}_x + I_y \overline{v}_y + I_t \right] \end{cases}$$

Résolution par Jacobi

Pour chaque pixel ou point de l'image, nous avons donc une paire d'équation que l'on peut résoudre itérativement par la méthode itérative de Jacobi en partant d'un champ de déplacement initialement nulle ($v_x^{[0]} = 0, v_y^{[0]} = 0$) :

$$\begin{cases} v_x^{[n+1]} = \overline{v}_x^{[n]} - I_x \big(I_x \overline{v}_x^{[n]} + I_y \overline{v}_y^{[n]} + I_t \big) / (\alpha^2 + I_x^2 + I_y^2) \\ \\ v_y^{[n+1]} = \overline{v}_y^{[n]} - I_y \big(I_x \overline{v}_x^{[n]} + I_y \overline{v}_y^{[n]} + I_t \big) / (\alpha^2 + I_x^2 + I_y^2) \end{cases} \qquad (4)$$

où n représente l'indice de l'échantillonnage temporel. De plus, on rappelle que $\overline{v}_x$ et $\overline{v}_y$ désignent les versions moyennées localement (dans un voisinage du pixel de coordonnée (x, y) de l'image) de v_x et v_y. Si i et j désignent respectivement la ligne et la colonne de l'image et $v_x(i, j)$, le vecteur v_x pour le pixel de coordonné (i, j) :

$$\begin{aligned} \overline{v}_x(i,j) \;\; = \;\; & \big(v_x(i-1,j) + v_x(i+1,j) + v_x(i,j+1) + v_x(i,j-1) \big)/6.0 \\ & + \big(v_x(i-1,j-1) + v_x(i-1,j+1) + v_x(i+1,j+1) + v_x(i+1,j-1) \big)/12.0 \end{aligned} \qquad (5)$$

De plus, les dérivées spatiales et temporelles de l'intensité I de l'image, *i.e.*, $(\partial I/\partial x, \partial I/\partial y, \partial I/\partial t)$ se calcule (une fois pour toute avant l'algorithme itératif de Jacobi) de la façon suivante :

$$\begin{aligned} I_x \;\; = \;\; & \big(I_{i,j+1,t} - I_{i,j,t} + I_{i+1,j+1,t} - I_{i+1,j,t} \\ & + I_{i,j+1,t+1} - I_{i,j,t+1} + I_{i+1,j+1,t+1} - I_{i+1,j,t+1} \big)/4.0 \\ I_y \;\; = \;\; & \big(I_{i+1,j,t} - I_{i,j,t} + I_{i+1,j+1,t} - I_{i,j+1,t} \\ & + I_{i+1,j,t+1} - I_{i,j,t+1} + I_{i+1,j+1,t+1} - I_{i,j+1,t+1} \big)/4.0 \\ I_t \;\; = \;\; & \big(I_{i,j,t+1} - I_{i,j,t} + I_{i+1,j,t+1} - I_{i+1,j,t} \\ & + I_{i,j+1,t+1} - I_{i,j+1,t} + I_{i+1,j+1,t+1} - I_{i+1,j+1,t} \big)/4.0 \end{aligned} \qquad (6)$$

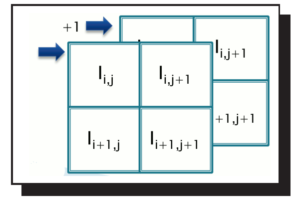


FIG. 3 – Disposition des pixels dans l'image I_t et I_{t+1}

1. Dans ce TP, je vous demande de résoudre le système 4 par la méthode itérative de Jacobi et de tracer la solution du flot optique pour l'image RUBIKSEQ (paramètre $\alpha = 500$) et pour l'image SALESMAN (paramètre $\alpha = 200$) (cette solution du flot optique pourra être ensuite visualisée avec le logiciel de traitement d'images gratuit "XnView" ou "Gimp" ou visualisé sur l'écran de l'ordinateur[1] (cf. Fig. 4).

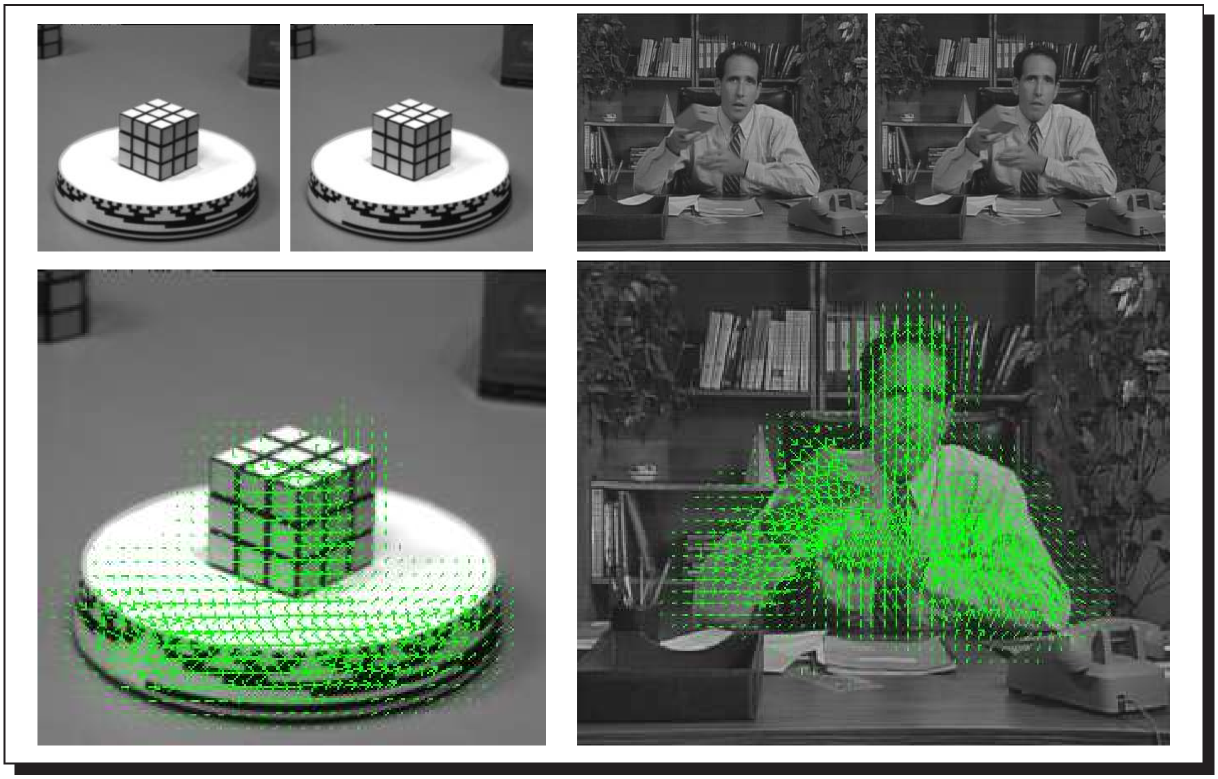


FIG. 4 – Estimation du flot optique obtenu sur l'image RUBIKSEQ ($\alpha = 500$) et l'image SALESMAN ($\alpha = 200$) après 1000 itérations de la méthode de Jacobi.

Réferences

[1] B.K.P. Horn and B.G Shunck. "Determining optical flow." *Artificial Intelligence, vol 17, pp. 185-203*, 1981.

[2] Wikipedia

[3]

Conseils Pratiques

Utiliser le programme que je vous donne sur ma page web. Ce programme initialise déja tout et la suite du programme affichera à l'écran, sous forme d'une séquence d'images, l'évolution temporelle du flot optique que vous aurez estimé par la méthode de Jacobi que vous allez implémenter.

[4]

Remise & Rapport

Vous devez rendre électroniquement le(s) programme(s) fait en C/C++ avant la date de remise spécifiée dans le fichier *barême* situé sur la page web du cours. Pour la remise électronique, utilisez le programme *remise* (*man remise* pour plus de détails) pour remettre votre code dans le répertoire TP<Numéro du Tp>. N'oubliez pas d'inscrire vos noms, courrier électronique en commentaire en haut du fichier .c remis. Les noms des programmes à remettre devront avoir le format suivant *Tp<Numero du Tp>-IFT2425-<Numéro de la question>.c*. Les programmes devront se compiler et s'executer sur Linux tel qu'indiqué dans le barême.