Extraction de trajectoires basée sur la cinématique dans les séquences d'images

G. MOSTAFAOUI[1], C. ACHARD[1], M. MILGRAM[1], L. LACASSAGNE[2]

[1]L'aboratoire des Instruments et Systèmes d'Iles de France, 4 place Jussieu, 75252 Paris cedex 05

[2]Institut d'Electronique Fondamental, Bât. 220, Centre d'Orsay, 91 405 Orsay cedex

ghiles.mostafaoui@lis.jussieu.fr, achard@ccr.jussieu.fr, maum@ccr.jussieu.fr, lionel.lacassagne@ief.u-psud.fr

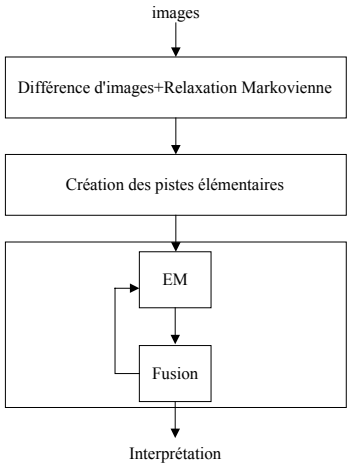
Résumé – Le suivi de personnes en mouvements sans aucune connaissance à priori du nombre de personnes présentes dans la scène et en prenant en compte les différents problèmes d'occlusions, s'avère difficile. Une détection de mouvement nous permet d'avoir des régions avec un nombre important de sous-segmentations et de sur-segmentations dues aux mauvaises conditions d'acquisition. L'étape de suivi qui doit prendre en compte tous ces problèmes, est réalisée avec l'algorithme EM (Expectation Maximization). Ce dernier utilise un model cinématique : on suppose que le mouvement apparent est rectiligne et uniforme, cette hypothèse reste localement valide pour beaucoup d'applications. Grâce à cette approche, des résultats satisfaisants ont été obtenus sur plusieurs séquences et ce sans aucune initialisation préalable.

Abstract *– The problem of moving person tracking, without knowledge about the number of persons in the scene, and by taking account of occlusion, is challenging. A first movement detection gives us regions with several under-segmentation and over-segmentation problem due to bad acquisition conditions. The tracking step, which has to manage all these problems, is realised with the EM algorithm (Expectation Maximization). This one uses a cinematic model: we suppose a rectilinear and uniform apparent motion, this hypothesis remains locally accurate in most of the applications. Goods results are obtained with this approach on several sequences, with any initialisation.*

1. Introduction

Il existe dans la littérature un nombre important d'algorithmes de suivi, ils se découpent en trois grandes approches : le suivi de régions, le suivi de points d'intérêts et le suivi de contours (Snakes, Bsplines). Notre méthode s'inscrit dans la première catégorie qui ne requiert aucune initialisation préalable. Différentes stratégies peuvent être employées pour l'étape de suivi. Parmi les plus courantes, on peut citer des méthodes heuristiques simples à mettre en œuvre comme [4] ; le filtrage particulaire [6] ne gérant qu'une seule piste mais pouvant aussi être étendu à plusieurs [5] ;des méthodes itératives utilisant la prédiction de la position des objets (filtres de Kalman étendu à plusieurs pistes) comme le JPDAF (Joint Probabilistic Data Association Filter) [1] ou encore, le MHT (Multiple Hypothésis Tracker) qui formule de manière récursive toutes les possibilités d'association des régions aux pistes [8]. Comme le suggère [3], le problème de suivi peut-être décomposé en deux : un problème d'association des régions aux pistes et un problème d'estimation des paramètres des pistes, ces deux problèmes interférant grandement l'un avec l'autre. Cette difficulté sera résolue dans notre cas grâce à l'algorithme EM (Expectation Maximization).

Nous décrivons dans cet article une méthode de suivi d'objets en mouvement basée sur la cinématique qui permet de résoudre les occlusions simples et de gérer les problèmes de sur-segmentation. Les différentes étapes sont résumées dans le synoptique ci-dessous.



2. Détection de mouvement

A chaque image de la séquence est soustrait une image dite de référence (image du fond). Ces images de différence sont très bruitées, c'est pourquoi il est utile de réaliser une étape de relaxation markovienne. Nous utilisons pour cela des cliques spatio-temporelles d'ordre 2 et l'algorithme ICM (Iterated Conditional Modes) très rapide en temps de calcul. D'autres détails sont fournis dans [2], [7].

3. Création des pistes élémentaires

Soient R_i^t et R_j^{t+1} deux régions en mouvement qui appartiennent respectivement aux images I_t et I_{t+1}. Ces deux régions sont voisines si l'aire de leur intersection (lorsqu'elles sont projetées dans la même image) n'est pas nulle.

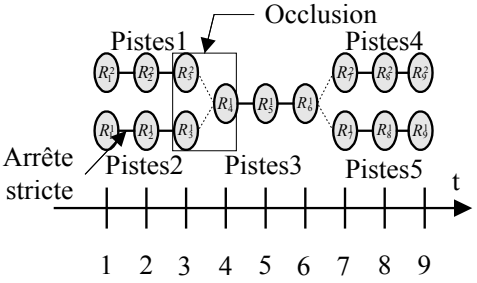


FIG. 1

On dit que R_i^t et R_j^{t+1} sont liées par une **arrête stricte** si R_i^t a pour seul voisin (au temps $t+1$) R_j^{t+1} et R_j^{t+1} a pour seul voisin (au temps t) R_i^t. On définit le graphe G des voisinages stricts (figure 1). Les nœuds de G sont les régions R_i^t. (R_i^t, R_j^{t+1}) est une arrête de G si les deux régions sont liées par une arrête stricte.

Une ***piste élémentaire*** est par définition un chemin dans le graphe G. Notons qu'une piste élémentaire ne peut comporter deux régions dans la même image I_t. Les régions présentes dans le graphe ci-contre vont ainsi amener à la construction de 5 pistes élémentaires.

Cette notion de ***pistes élémentaires*** est très importante : elle constitue l'élément de base de la construction des vraies trajectoires. On peut ainsi générer sur une séquence un nombre important de pistes élémentaires. Ne voulant prendre aucune décision irrévocable au bas niveau, nous avons jugé préférable d'attendre les informations du haut niveau pour réduire le nombre de pistes.

Nous introduisons maintenant des ***modèles cinématiques*** pour, plus tard, regrouper les pistes élémentaires. A chaque piste élémentaire, nous associons une droite d'espace temps ayant pour équation $x=x_0+ t*dx$ et $y=y_0+t*d$. Les paramètres (dx,dy) représente le vecteur vitesse du modèle en pixels/image. L'intérêt de cette modélisation est qu'un objet bien segmenté aux instants $[t_0\ t_1]$ et $[t_2\ t_3]$ mais invisible (occlusion) sur $]t_1\ t_2[$ (avec $t_0<t_1<t_2<t_3$), donnera une piste élémentaire pour chaque intervalle de temps mais les deux pistes auront approximativement les mêmes modèles cinématiques. Ce modèle n'est rigoureux que pour des mouvements apparents rectilignes uniformes, il reste encore localement valide dans la plupart des applications.

Soit P la piste élémentaire constituée de la suite de régions $\{R_1^t ... R_k^{t+k-1} ... R_k^{t+n-1}\}$ d'aire $\{S_1 ... S_k ... S_n\}$ et de centres de gravités $([x_1,y_1,t_1], \ldots, [x_k,y_k,t_k], \ldots, [x_n,y_n,t_n])$. Nous voulons estimer les paramètres (dx, dy, x_0, y_0) de cette piste. Pour cela, on estime d'abord la vitesse de chaque région (basée uniquement sur les voisinages stricts) : $(VX_k,VY_k)=(x_{k+1}-x_k, y_{k+1}-y_k)$. Malheureusement, cette vitesse est très sensible aux problèmes de segmentation.

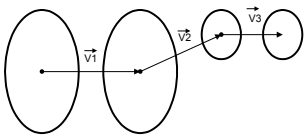


FIG. 2

Ainsi dans le cas de la figure 2, *V2* est erroné à cause d'une mauvaise segmentation. Pour des pistes de courte durée, cela engendre une erreur importante sur la vitesse moyenne de la piste élémentaire (dx, dy). Nous pondérons chaque vitesse par une mesure de pertinence m_k. m_k est ici une fonction de la différence d'aire entre les surfaces car les problèmes apparaissent quand deux régions consécutives ont des aires différentes. Les autres paramètres (x_0 et y_0) se déduisent alors aisément avec les relations suivantes :

$$\begin{cases} dx = \sum_{k=1}^{k=n-1} m_k * VX_k \\ dy = \sum_{k=1}^{k=n-1} m_k * VY_k \end{cases}$$

avec

$$m_k = \frac{\exp\left\{\frac{-|(S_{k+1}-S_k)|}{\alpha}\right\}}{\sum_{j=1}^{n-1}\exp\left\{\frac{-|(S_{j+1}-S_j)|}{\alpha}\right\}}$$

et

$$\begin{cases} x_0 = x_1 - t_1 * dx \\ y_0 = y_1 - t_1 * dy \end{cases}$$

α est un paramètre utilisé dans le '*softmax*' qui sert à diminuer la décroissance de l'exponentielle(il a été fixé à 50). La vitesse moyenne (dx,dy) ainsi calculée, est alors affectée définitivement à chacune des régions composant la piste (ce qui réalise un lissage des vitesses).

4. fusion des pistes

A cette étape, la séquence est représentée par N pistes élémentaires, chacune étant caractérisée par ses paramètres (dx, dy, x_0, y_0). Notre objectif est ici de résoudre les occlusions et les sur-segmentations à partir des paramètres cinématiques des pistes élémentaires. En d'autres termes fusionner les pistes élémentaires qui devraient appartenir au même objet en mouvement mais qui ont été séparées par des occlusions ou des sur-segmentations. Nous utilisons pour cela l'algorithme EM qui est itératif, chaque itération étant composée de deux étapes. La première (étape E) résout de manière probabiliste le problème d'association des régions aux pistes, la deuxième (étape M) ré-estime les paramètres des modèles cinématiques des pistes.

4.1 Algorithme EM

L'étape E permet de calculer les probabilités d'appartenance de chaque région aux modèles de pistes

élémentaires. Soit R_k^t une région de centre de gravité (x_R, y_R, t_R) et de direction θ_R (orientation du vecteur vitesse calculé précédemment) et P une piste élémentaire de paramètre $(dx_0^p, dy_0^p, x_0^p, y_0^p)$ et de direction cinématique θ_p. L'erreur (distance modulée par la direction) entre la région R_k^t et la piste P est définie par :

$$E_k^P = F(|\theta_R - \theta_P|) * \sqrt{\left(x_R - (x_0^P + t_R * dx_0^P)\right)^2 + \left(y_R - (y_0^P + t_R * dy_0^P)\right)^2}$$

$$avec \begin{cases} F(\theta) = 1 \;\; si \;\; \theta \le \varepsilon \\ F(\theta) = \exp\left(\dfrac{|\theta - \varepsilon|}{A}\right) \;\; si \;\; \theta > \varepsilon \end{cases} \qquad heuristiquement \quad A = 18$$

La fonction $F(\theta)$ a pour rôle d'éloigner les régions dont l'orientation est différente de celle de la piste. On calcule maintenant la probabilité d'appartenance de la région R_k^t à la piste P :

$$w_k^P = \frac{\exp\left(-\dfrac{E_k^P}{\alpha}\right)}{\sum_{i=1}^{NR} \exp\left(-\dfrac{E_i^P}{\alpha}\right)}$$

où NR est le nombre total de régions détectées à travers toute la séquence. On remarque de plus que $\sum_{P=1}^{NP} w_k^P = 1$, NP étant le nombre de pistes

L'étape M consiste à ré-estimer les paramètres des modèles de pistes par la méthode des moindres carrés. On détermine les paramètres du modèle p qui minimise la somme des distances entre ce modèle (droite spatio-temporelle) et les centres de gravité des régions pondérés par les probabilités calculées à l'étape E.

4.2 Fusion

Soient P^1 et P^2, deux pistes éventuellement fusionnables. On calcule une erreur globale associée à chaque piste représentant la dispersion des régions autour de celle-ci. Pour la piste 1, $E^1 = \sum_{k=1}^{NR} w_k^1 E_k^1$. On calcule E^2 de la même façon. L'hypothèse de fusion se traduit par la prise en compte d'une nouvelle piste P^{12} dont on estime les paramètres avec les moindres carrés : minimisation de la somme des distances entre cette piste et les centres de gravité des régions pondérés par les probabilités $w_i^{P12} = w_i^{P1} + w_i^{P2} \quad \forall i$. On est maintenant en mesure de déterminer l'erreur globale E^{12}de la piste P^{12}. La différence entre E^{12} et la moyenne de E^1 et E^2 nous renseigne sur la distance entre les pistes P^1 et P^2. En seuillant (S_f) cette différence, on détermine si on fusionne les deux pistes ou non. Si elles sont fusionnées (P^1 et P^2 sont supprimées et remplacées par P^{12}), on recommence à l'étape EM. Sinon, on teste deux autres pistes jusqu'à ce que plus aucune ne soit fusionnables.

Plus précisément, à chaque fois que l'algorithme EM est appliqué, chaque piste P^i est fusionnée avec la piste P^j la plus proche (si P^j est à une distance inférieure à S_f). Le processus s'arrête si plus aucune piste n'est fusionnable.

5. Résultats et conclusion

Notre algorithme de suivi a été testé sur une dizaine de séquences réelles. Nous présentons ci-dessous les résultats obtenus sur deux séquences. La première représente deux personnes qui se croisent. La seconde une seule personne qui progresse vers les x négatifs et qui rebrousse chemin pour se déplacer vers les x positifs.

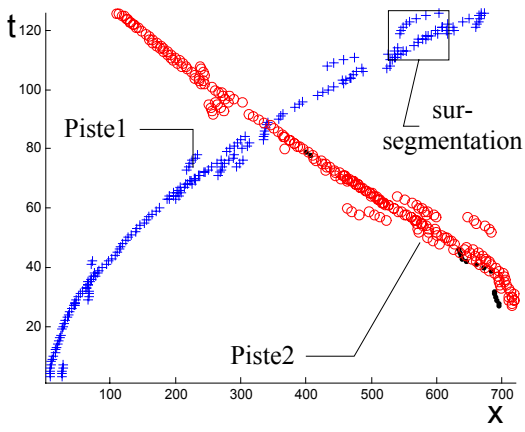


FIG. 3

Nous trouvons sept trajectoires pour la première séquence (figure 3, Tableau 1) :

- Les deux vraies trajectoires qui correspondent aux deux personnes en mouvement.
- Cinq autres trajectoires de durée très courte (22 régions pour les cinq trajectoires soit une durée moyenne de 4.4 images/trajectoire.

Remarquons qu'un simple seuillage sur le nombre de régions de chaque piste permettrait de récupérer simplement les deux pistes principales (ce qui est également vrai pour la deuxième séquence). Cette séquence présente beaucoup de sur-segmentations (voir figure 3). Celles-ci ont amené la création d'un nombre important de pistes élémentaires (154). Les différentes régions ont néanmoins été associées à la même piste grâce à l'algorithme EM (sans connaissance *a priori* du nombre d'objets en mouvement).

Pour la seconde séquence (figure 4, Tableau 1), nous obtenons deux pistes principales correspondant aux deux mouvements aller et retour (les six autres totalisent 23 régions concentrées au moment du demi-tour soit une durée moyenne de 3.83 images/trajectoire). Ce résultat est logique vu que nous utilisons un modèle de mouvement apparent uniforme (droites spatio-temporelles).

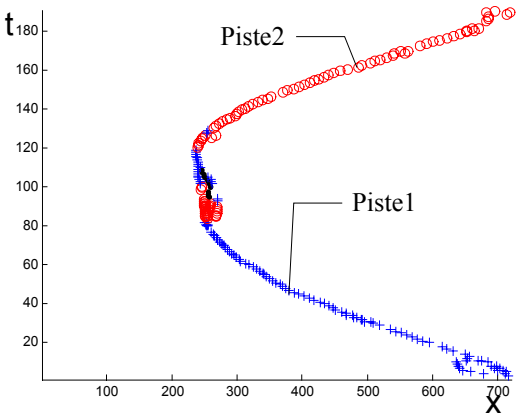


FIG. 4

TAB. 1 : Résultats

	Séquence 1	Séquence 2
nbre de personnes	2	1
nbre total de régions	524	287
nbre de pistes élémentaires	154	91
nbre de trajectoires finales	7	8

Nous avons montré l'efficacité de notre algorithme de suivi de mouvement quant à la gestion des occlusions simples et de la sur-segmentation sur des séquences réelles. Certains types de mouvements non uniformes ne sont toujours pas gérés par ce système, ce qui se traduit par la création de plusieurs trajectoires pour un même objet (séquence 2). Une étape de fusion supplémentaire, utilisant d'autres informations que la cinématique (couleur, forme, …) devrait résoudre ces problèmes.

Références

[1] Y. Bar-Shalom, XR Li, *Multitarget-Mulisensor tracking*, Publisher: Yaakov Bar-Shalom, 1995.

[2] A. Caplier, F. Luthon, C. Dumontier , *Algorithme markovien de détection de mouvement, mise en œuvre 'temps réel'*, GRETSI 1995.

[3] H. Gauvrit, Extraction multi-pistes : approche probabiliste et approche combinatoire, thèse de 3ème cycle, nov. 1997

[4] I. Haritaoglu, D. Harwood, L.S. Davis, *W4S : a real time system for detecting and tracking people in 2,5D*, European Conference Computer Vision, 1998, Maryland.

[5] C. Hue, J.P. Le cadre, P. Perez, *Tracking multiple objects with particle filtering*, RR INRIA n° 4033, 2000

[6] M. Isard, A. Blake, *Condensation | conditional density propagation for visual tracking*, Int. J. Computer Vision, 29, 1, 5--28, 1998.

[7] L. Lacassagne, F. Lohier, M. Milgram, P. Garda, *Implémentation temps réel 'algorithmes de détection de mouvement par champs de Markov sur RISC et DSP C6x,*GRETSI**,** Septembre '99, Vannes.

[8] D.B. Reid, *An algorithm for Tracking Multiple Targets*, IEEE Trans. on Automatic Control, Vol. AC-24, N° 6, pp 843-854, 1979.