

Contours actifs pour le suivi d'objet en temps-réel: multi-topologies et multi-résolutions

Sébastien LEFÈVRE¹, Nicole VINCENT²

¹ Laboratoire des Sciences de l'Image, de l'Informatique et de la Télédétection (LSIIT)
Université Louis Pasteur – Strasbourg I
Parc d'Innovation, Bd Brant, BP 10413, 67412 Illkirch Cedex

² Centre de Recherche en Informatique de Paris V (CRIP5)
Université René Descartes – Paris V
45 rue des Saints Pères, 75270 Paris Cedex 06

lefevre@lsiit.u-strasbg.fr, nicole.vincent@math-info.univ-paris5.fr

Résumé – Dans cet article notre objectif est de présenter des solutions au suivi d'objets multiples dans une séquence d'images avec une contrainte de temps-réel et une caméra mobile. Nous proposons une solution basée sur les modèles de contours actifs, en s'affranchissant de leurs limites bien connues liées à l'initialisation, au réglage optimal des paramètres, au coût de calcul, et à l'incapacité à gérer les changements de topologie. Pour cela, l'algorithme est basé sur des énergies originales, un processus de suivi par boîte englobante, des étapes de scission et fusion, et une analyse multirésolution des images.

Abstract – In this paper our purpose is to present some solutions to multiple object tracking in an image sequence with a real time constraint and a mobile camera. We propose a solution which is built upon active contours models, without their well-known limitations related to initialisation, optimal parameter setting, computational cost, and inability to deal with topology changes. To do so, the algorithm is based on original energies, a box-based tracking process, splitting and merging steps, and multiresolution image analysis.

1 Introduction

Parmi les problèmes qu'il est nécessaire de résoudre lors de l'analyse de séquences vidéo, le suivi d'objet occupe une place prépondérante. Lorsque les objets à suivre sont de nature non-rigide (comme des corps humains), l'utilisation de modèles déformables comme les contours actifs est préconisée. La puissance d'un outil comme les *level sets* [7] n'est plus également à démontrer, mais le temps de calcul nécessaire est généralement prohibitif dans un contexte d'exécution en temps-réel. C'est pourquoi nous considérerons plutôt ici les contours actifs, de complexité plus réduite. Les contours actifs [4] ont été largement utilisés pour résoudre des problèmes tels que la segmentation d'image et le suivi des objets.

Dans cet article, nous montrons qu'il est possible d'utiliser les contours actifs pour suivre des objets en intégrant au modèle des améliorations qui permettent de supprimer leurs principales limites relatives au réglage des paramètres, à la sensibilité à l'initialisation, à l'incapacité de changer de topologie, et au temps de calcul prohibitif.

2 Modèles de contours actifs

Dans la définition initiale de Kass *et al* [4], un contour actif est représenté par une courbe v , fermée ou non, évoluant au cours du temps. Cette évolution s'effectue selon la minimisa-

tion d'une fonctionnelle d'énergie :

$$E(v) = \int_0^1 [\alpha_{\text{int}} E_{\text{int}}(v(s)) + \alpha_{\text{ext}} E_{\text{ext}}(v(s))] ds \quad (1)$$

où E_{int} et E_{ext} représentent les énergies internes et externes. Les coefficients α_{int} et α_{ext} sont utilisés pour donner plus ou moins d'influence à chaque énergie. L'énergie interne représente les propriétés physiques intrinsèques du contour (contour globalement homogène et de forme simple) tandis que l'énergie externe est le terme d'attache aux données qui permettra au contour actif de se fixer sur les limites des objets suivis.

A partir de ce modèle initial, différentes implémentations ont été proposées et nous avons adopté l'algorithme glouton (ou *greedy*) [10] connu pour être le plus efficace (cf. l'étude de Denzler et Niemann [3]). Il consiste en une implémentation discrète et locale du contour. La définition de la fonction d'énergie est alors la suivante :

$$E(V) = \sum_{i=1}^m E(V^\gamma(i)) \quad \text{et} \quad V = \arg \min_{V(i), i \in [1, m]} E \quad (2)$$

où V représente le contour actif discret et $V(i)$ son $i^{\text{ème}}$ point. Ce contour actif est constitué par un nombre m de points. La déformation est effectuée itérativement et localement en chaque point du contour, V^γ représentant le contour à l'itération γ .

3 Définition des énergies utilisées et description de la méthode de suivi

La méthode de suivi proposée dans cet article est basée sur différentes énergies internes et externes, et se différencie de l'existant par une initialisation par boîte englobante et non par estimation de mouvement.

L'énergie interne est constituée des énergies classiques de continuité et de courbure. Nous y intégrons également une autre énergie non liée au contenu de l'image, l'énergie ballon inversée. Cette dernière permet de réduire le contour actif plutôt que de l'étendre, comme nous le verrons par la suite.

L'énergie externe est quant à elle composée de deux énergies liées respectivement à des informations relatives au gradient et à l'arrière-plan de l'image. Pour la première, nous nous basons sur l'estimateur de Sobel, appliqué de façon marginale sur les trois composantes RGB, et dont la valeur est seuillée pour limiter la sensibilité au bruit. Cette énergie a pour but de fixer le contour actif sur les frontières de l'objet suivi. La seconde énergie permet au contour actif de rester sur les bords de l'objet, évitant ainsi une convergence à l'intérieur de la forme de l'objet suivi. Si l'arrière-plan est homogène, elle utilise des informations a priori sur ses caractéristiques couleur, et est définie en chaque point comme la différence (seuillée également pour limiter la sensibilité au bruit) entre la couleur du point et la couleur moyenne de l'arrière-plan. Dans le cas d'un fond non homogène, cette énergie est obtenue par comparaison non plus avec la couleur moyenne mais avec une image de référence, ce qui nécessite une caméra statique. Nous ne considérons pas le cas simultané d'un fond complexe et d'une caméra en mouvement pour laquelle nous n'avons aucun paramètre.

Puisque le but ici est de suivre en temps réel des objets non-rigides couleur par une caméra en mouvement, nous avons décidé afin de limiter le temps de calcul de ne pas effectuer de pré-traitement (tel un filtrage) ni d'estimation du mouvement des objets ou de compensation du mouvement de la caméra. Ces choix différencient clairement notre contribution des travaux précédents de la littérature. Pour effectuer le suivi, nous considérons sur chaque image deux étapes successives : l'initialisation puis la déformation du contour actif.

L'initialisation est obtenue par la création d'un rectangle englobant le résultat fourni à l'image précédente. Ce rectangle est supposé englober également le résultat attendu sur l'image courante. La dilatation du rectangle englobant permet de prendre en compte le mouvement de la caméra et de l'objet.

La déformation amène ensuite le contour actif à se réduire pour se fixer sur les limites de l'objet. Ces deux étapes sont illustrées dans la figure 1 qui nous permet de plus de comparer notre approche originale avec le schéma traditionnel. Une initialisation spécifique est nécessaire pour la première trame d'un plan, où aucune information sur le contour actif n'est disponible *a priori*. Dans ce cas, l'initialisation est effectuée de manière automatique en utilisant le résultat d'une étape de séparation du fond et des objets [5]. La technique de suivi présentée ici ne permettant pas de gérer des objets se croisant dans la scène, nous proposons dans la section suivante un processus pour considérer des topologies multiples.

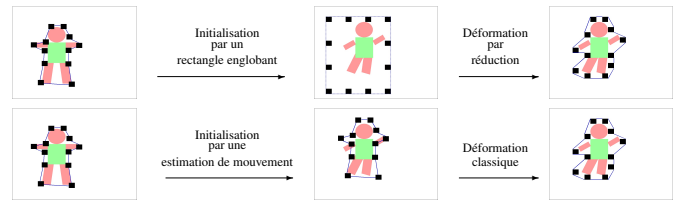


Figure 1: Suivi avec des contours actifs : le processus de suivi débute toujours par le résultat précédent (image de gauche) pour obtenir le résultat courant (image de droite). Cependant, notre contribution ne nécessite pas d'estimation de mouvement (ligne du haut) contrairement aux autres approches par contours actifs (ligne du bas).

4 Multi-topologies

Lorsque plusieurs objets d'intérêt sont présents dans la scène, il est nécessaire d'intégrer au modèle de contour actif une capacité à changer de topologie. D'après la taxonomie introduite par Cheung *et al* [1], notre approche n'est donc pas basée sur une représentation indépendante d'une topologie particulière comme les ensembles de niveaux, mais requiert plutôt l'utilisation explicite d'une technique de *division-et-fusion* [2, 6]. Nous considérons un mécanisme de scission et fusion pour respectivement séparer un contour en deux lorsque deux objets proches sont suivis par un unique contour et s'éloignent l'un de l'autre et pour regrouper deux contours en un seul lorsque deux objets éloignés se rapprochent. Pour plus d'efficacité, cette étape de scission/fusion ne sera effectuée sur chaque image qu'après convergence du ou des contours actifs.

L'objectif de l'étape de scission est de diviser le contour actif en plusieurs composants. D'après l'équation (2), l'énergie obtenue est un minimum. Puisque nous utilisons une implémentation locale et discrète, chacun des points $V(i)$ est un optimum local. Comme nous le verrons plus loin, nous souhaitons donner autant d'importance aux énergies interne et externe. Cependant, les énergies externes ne fournissent pas toujours un unique minimum (ou optimum). En effet, ces termes ont été seuillés pour accroître la robustesse au bruit. Localement, les valeurs de ces énergies peuvent être uniformément égales à 0. Une fois la convergence du contour actif obtenue, certains points du contours peuvent être piégés dans ces zones d'énergie externe constante et ne pas se fixer correctement sur les bords des objets suivis, comme l'illustre la figure 2.

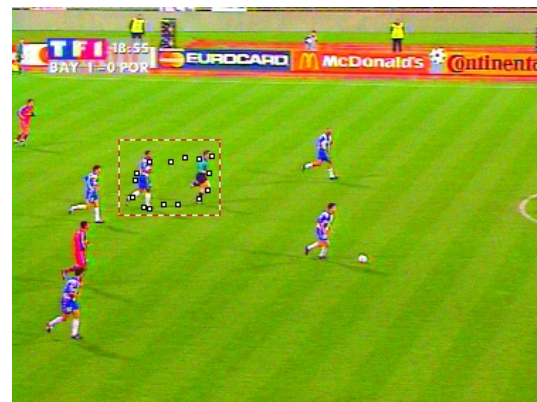


Figure 2: Répartition précise des points du *snake* avant d'effectuer l'étape de scission.

Nous proposons donc de supprimer ces points et de scinder les contours actifs aux positions de ces points incorrects. A partir de la liste des points restantes, chaque séquence de points successifs est utilisée pour créer un nouveau contour fermé. La figure 3 illustre ce procédé. La qualité de ces nouveaux contours n'est pas optimale, mais est suffisante pour le suivi dans les images suivantes de la séquence.

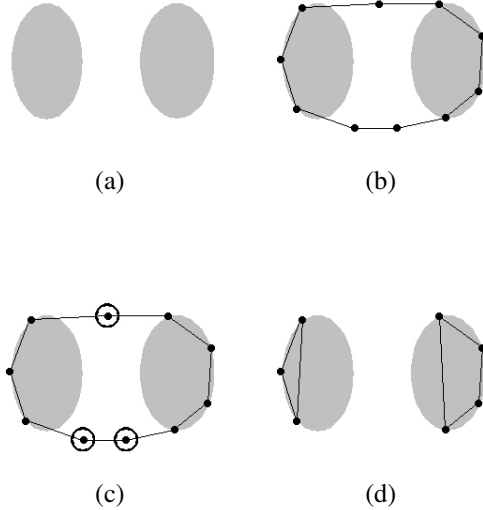


Figure 3: Processus de scission de deux objets (a), associés à un unique contour (b) ; ce contour contient des points incorrects (c), supprimés pour créer les contours finaux (d).

Il peut arriver que les contours créés par l'étape de scission ne soient pas corrects. En effet, certains de ces contours peuvent être associés à des pixels représentant le bruit ou l'arrière-plan présent dans l'image. Une étape de sélection est donc ensuite nécessaire pour valider les contours obtenus et supprimer ceux qui semblent non valides. La validation d'un composant est basée sur la comparaison de l'aire délimitée par le contour avec un intervalle prédéfini, correspondant à la taille des objets suivis.

L'étape de scission décrite précédemment nécessite la définition d'une étape de fusion correspondante. La fusion de deux contours actifs a lieu si les objets suivis par ceux-ci se rapprochent, avant un phénomène d'occlusion. L'heuristique utilisée dans ce cadre consiste à comparer les centres de gravité des contours actifs et à évaluer la proportion de chevauchement.

5 Multi-résolutions

Notre approche intègre également un aspect multirésolution afin d'accélérer le traitement nécessaire au suivi des objets. Afin d'effectuer le suivi dans un cadre multirésolution, nous traitons les images selon une approche pyramidale *coarse-to-fine*. Le processus multirésolution n'est donc pas associé au modèle de contour actif mais plutôt aux images analysées, contrairement aux approches similaires de la littérature [9]. De plus, notre processus multirésolution est stoppé dès qu'une solution correcte a été obtenue, au sens du critère de décision défini dans la section précédente. Le processus peut donc ne

pas être réalisé jusqu'à la résolution originale, fournissant ainsi un résultat imprécis en contrepartie d'un temps de calcul plus faible. Cet aspect est particulièrement intéressant lorsqu'un contour obtenu à faible résolution sur une trame t est suffisant pour initialiser l'algorithme sur une trame $t + 1$.

L'intégration d'un cadre multirésolution au modèle de contour actif implique naturellement la modification des paramètres de l'algorithme selon la résolution, et nous avons donc été amenés à définir des fonctions d'évolution des paramètres selon la résolution étudiée. Ainsi, la taille du rectangle englobant, le nombre de points composant le contour actif, le nombre d'itérations du processus de déformation, et les seuils utilisés peuvent varier selon la résolution considérée. Les résolutions pour lesquelles l'algorithme peut donner des résultats pertinents sont illustrées par la figure 4.



Figure 4: Représentation d'une image à différentes résolutions, depuis $r = 0$ (en haut à gauche) jusqu'à $r = 5$ (en bas à droite). Les images ont été rééchantillonnées spatialement pour observer les différences de détail.

De plus, nous pouvons remarquer que cette approche permet de bénéficier des codages compressés. En effet, dans le cas de séquences vidéo compressées à l'aide d'une Transformée en Cosinus Discrète, les coefficients DC peuvent être utilisés pour fournir une image faible résolution, de taille 8×8 fois inférieure à la taille originale.

6 Résultats et Conclusion

Nous avons effectué des tests sur des séquences vidéo couleur de matchs de football obtenues à l'aide d'une caméra en mouvement. Les images, de taille 384×284 pixels, ont été acquises à une fréquence de 15 Hz. A la résolution originale, nous avons considéré un voisinage de taille 5×5 pixels, un nombre d'itérations maximal fixé à 30, et nous décidés de supprimer les contours d'aire inférieure à 10 pixels. Le processus multirésolution fonctionne à une résolution de taille $16 \times 16 = 256$ fois inférieure à l'image originale ($r = 4$), comme l'illustre la figure 5. On peut observer le manque de précision de la forme du contour actif, qui est due à la faible résolution à laquelle l'image est analysée.

Le contour actif est également capable de changer de topologie pour s'adapter aux multiples joueurs présents dans la scène. La figure 6 illustre l'intérêt de cette capacité.

Les temps de calcul nécessaires, généralement de 35ms par trame pour un PC Pentium IV cadencé à 1,7 GHz, sont faibles par rapport à d'autres approches récentes [8]. Nous avons également testé notre approche sur des séquences de vidéosurveillance obtenues par une webcam statique avec un fond hétéro-

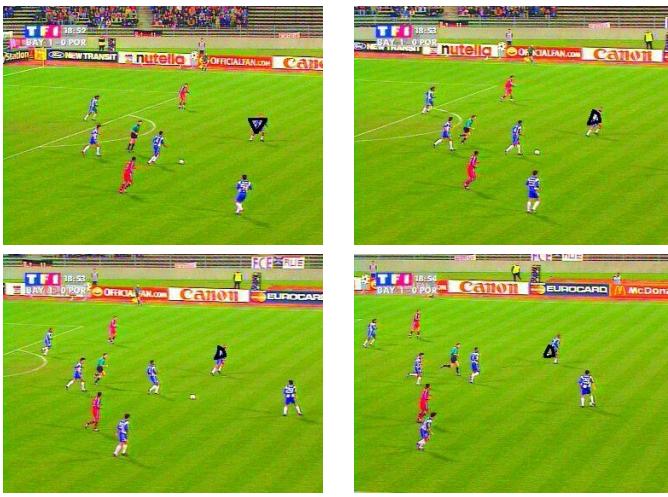


Figure 5: Suivi d'objet non-rigide à une résolution 256 fois inférieure à la résolution originale (images similaire à la 5^{ème} image de la figure 4).

gène, ceci afin de vérifier la généralité de notre approche et de valider la définition de l'énergie d'arrière-plan basée sur une image de référence.

Dans cet article, nous proposons une solution au problème du suivi d'objet multiples dans une séquence vidéo couleur acquise par une caméra mobile. Notre approche s'affranchit des limites habituelles des contours actifs et consiste en deux étapes que sont l'initialisation par un rectangle englobant et la déformation par réduction. Elle se différencie notamment par le fait qu'elle ne nécessite ni prétraitement, ni calcul et utilisation de l'information de mouvement. De plus, pour gérer des objets multiples aux trajectoires croisées, nous avons intégré au modèle une capacité à changer de topologie par un mécanisme de scission et fusion. Nous avons également introduit un cadre multirésolution dans lequel l'analyse des images est effectuée.

Nous souhaitons maintenant intégrer à notre modèle des énergies plus robustes, puis le comparer aux techniques émergentes de *level sets* en considérant simultanément des critères de qualité et d'efficacité. Nous devons également intégrer au modèle des caractéristiques calculées sur les régions (telles que la couleur ou la texture) pour assurer une correspondance des objets après un phénomène d'occlusion.

Références

- [1] K.W. Cheung, D.Y. Yeung, and R.T. Chin. On deformable models for visual pattern recognition. *Pattern Recognition*, 35(7):1507–1526, Juillet 2002.
- [2] H. Delingette and J. Montagnat. Shape and topology constraints on parametric active contours. *Computer Vision and Image Understanding*, 83(2):140–171, Août 2001.
- [3] J. Denzler and H. Niemann. Evaluating the performance of active contours models for real-time object tracking. In *Asian Conference on Computer Vision*, volume 2, pages 341–345, Singapour, 1995.
- [4] M. Kass, A. Witkins, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, Janvier 1988.

- [5] S. Lefèvre, L. Mercier, V. Tiberghien, and N. Vincent. Multiresolution color image segmentation applied to background extraction in outdoor images. In *IS&T European Conference on Color in Graphics, Image and Vision*, pages 363–367, Poitiers, France, Avril 2002.
- [6] T. McInerney and D. Terzopoulos. T-snakes: Topology adaptive snakes. *Medical Image Analysis*, 4(2):73–91, Juin 2000.
- [7] N. Paragios and R. Deriche. Geodesic active contours and level sets for the detection and tracking of moving objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(3):266–280, Mars 2000.
- [8] F. Precioso. *Contours actifs paramétriques pour la segmentation d'images et vidéos*. Thèse de doctorat, Université de Nice - Sophia Antipolis, Septembre 2004.
- [9] N. Ray, B. Chanda, and J. Das. A fast and flexible multiresolution snake with a definite termination criterion. *Pattern Recognition*, 34(7):1483–1490, Juillet 2001.
- [10] D.J. Williams and M. Shah. A fast algorithm for active contours and curvature estimation. *Computer Vision, Graphics and Image Processing: Image Understanding*, 55(1):14–26, Janvier 1992.

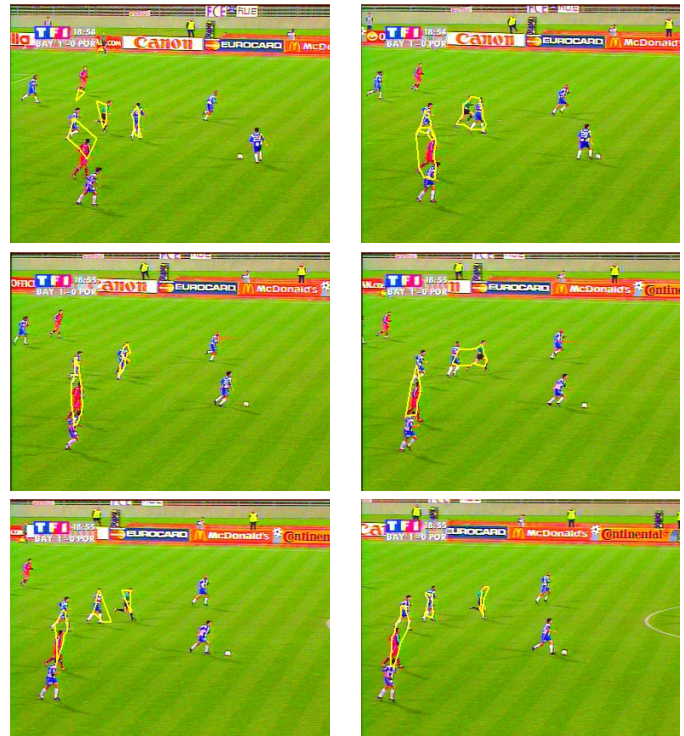


Figure 6: Intérêt des étapes de scission et fusion dans le cas d'objets proches et d'occlusions temporaires.