

Approche de Suivi d'Objet par Courbes de Niveau

A. GANOUN^{1,2}, R. CANALS¹, S. TREUILLET³

¹Laboratoire d'Electronique, Signaux, Images - Universite d'Orleans - 12 rue de Blois, 45067 Orléans Cedex 2

²Faculté d'Ingénierie, Université de Garyounis, Benghazi, LIBYE

³Laboratoire de Vision et Robotique - Université d'Orléans - 12 rue de Blois, 45067 Orléans Cedex 2

Ali.Ganoun@univ-orleans.fr, Raphael.Canals@univ-orleans.fr, Sylvie.Treuillet@univ-orleans.fr

Résumé – *Beaucoup d'approches ont été développées pour résoudre le problème de suivi de cible, dont l'approche statistique basée sur la méthode des courbes de niveau. Deux des principaux problèmes de suivi de cible en temps réel sont le coût de calcul, et la robustesse de l'algorithme pour le suivi des objets déformables. Dans cet article nous présentons deux techniques pour résoudre ces deux problèmes. La première est une méthode rapide pour diminuer le coût de calcul de l'algorithme standard, basée sur l'utilisation du signe de la fonction de vitesse au lieu d'employer sa valeur, et l'utilisation d'une zone spécifique près de la courbe de niveau zéro. La deuxième technique c'est quand nous calculons la force statistique en chaque point, nous considérons l'effet moyen des points voisins au lieu de l'effet du seul point traité.*

Abstract – *Many approaches were developed to solve the target tracking problem, from which the statistical framework based on the Level Set Method (LSM). Two of the problems related to the real time target tracking problem are the computational cost, and the robustness of the algorithm to track deformable targets. In this paper we present two techniques to solve these two problems. The first is a fast method to decrease the computational cost of the standard algorithm, based on the use of the sign of the speed function that moves the propagating interface instead of using its value, and the use of a specific zone near the zero level set. The second technique is to use the average effect of the neighbouring points for calculating the statistical force acting at each point instead of the effect of only one point.*

1. Introduction

La méthode des courbes de niveau, présentée par Osher et Sethian [1], est basée sur l'équation différentielle partielle de courbe de niveau à valeur initiale. Cette méthode réalise le suivi de l'évolution d'une fonction implicite dépendante du temps dont le niveau zéro correspond toujours à la position de l'interface de propagation définissant le contour de l'objet à suivre. Les principales raisons de l'utilisation du LSM sont diverses et variées [2], [3] : sa flexibilité, sa stabilité, sa gestion automatique d'éventuels changements de topologie, tels que la division ou la fusion.

Un problème avec l'algorithme standard est qu'il est trop lent pour les applications temps réel. En effet, cet algorithme nécessite la mise à jour et le calcul de la fonction pour tous les points de l'image et pas uniquement pour la courbe de niveau zéro. Par conséquent, le développement d'un algorithme rapide et efficace est de grande importance. Parmi les algorithmes rapides relatifs à l'approche LSM, on trouve la « Fast Marching Method » (FMM) [2], mais qui n'est pas adaptée à notre besoin, la « Narrow Band Method » (NBM) [4], [5], et la « Signed Function Method » (SFM) [6].

Un autre problème lié à l'approche LSM est celui de l'initialisation. Généralement, le suivi débute par la détermination du contour de l'objet à suivre ; nous avons constaté que si le contour initial n'est pas précis, l'algorithme de suivi cible risque fortement d'échouer. C'est la raison pour laquelle nous avons développé un algorithme qui peut résoudre ce problème en employant l'effet moyen des points voisins au lieu de celui du seul point traité.

Cet article est organisé comme suit : la section 2 est consacrée à la méthode des courbes de niveau et explique comment le coût calculatoire peut être réduit afin d'atteindre des objectifs de fonctionnement temps réel. Dans la section 3, nous présentons la manière dont nous utilisons également le voisinage du point traité pour augmenter la robustesse de notre approche. Les résultats expérimentaux sur images réelles sont présentés en section 4, suivis d'une conclusion en section 5.

2. La Méthode des Courbes de Niveau

Dans la méthode des courbes de niveau, nous déformons une image donnée avec une équation différentielle partielle (EDP) que nous résolvons afin d'obtenir la solution à notre problème. L'idée centrale est de suivre l'évolution d'une fonction $\Phi(\gamma, t)$ dont le niveau zéro ($\Phi = 0$) correspond toujours à la position de l'interface de propagation γ [1], l'EDP étant donnée par :

$$\Phi_t + F |\nabla \Phi| = 0 \quad (1)$$

où F est la fonction de vitesse dans la direction normale à l'interface de propagation (Fig. 1). Dans cette dernière équation, la surface fermée γ est représentée implicitement par la courbe de niveau zéro de la fonction $\Phi(\gamma, t)$.

Une approche pour obtenir la nouvelle courbe de niveau zéro est d'employer la technique d'itération formulée dans [2]:

$$\Phi^{n+1} = \Phi^n + \Delta t [F |\nabla \Phi^n|] \quad (2)$$

où Δt est le pas de discrétisation temporel et n le numéro de l'itération. La phase d'itération commence après la phase d'initialisation de l'algorithme qui consiste à trouver la courbe initiale de niveau zéro Φ^0 , et à calculer la fonction de vitesse qui déplace l'interface vers sa nouvelle position. Pour chaque point (x,y) , nous calculons cette fonction F qui reste alors identique durant toute la phase d'itération [9], [10]:

$$\bar{F} = \log\left(\frac{P_{in}}{P_{out}}\right)\bar{n} \quad (3)$$

P_{in} et P_{out} sont les probabilités pour que ce point (x,y) dans I^{n+1} soit respectivement à l'intérieur ou à l'extérieur de la région à suivre, et \bar{n} est la normale en ce point. Si l'intensité en ce point correspond mieux à la distribution de la région R_1 que celle de la région complémentaire R_1^c , alors ce point se déplacera suivant \bar{n} .

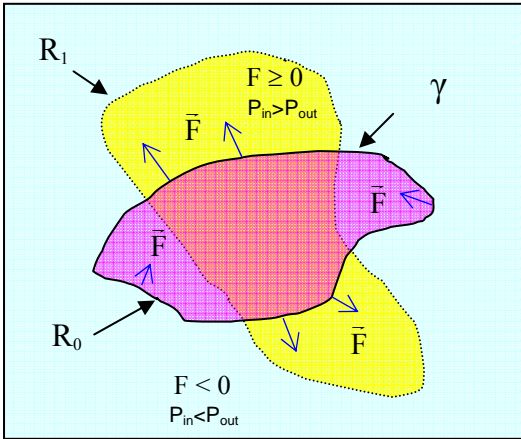


Fig.1 : La surface fermée et la fonction F

Comme nous l'avons expliqué auparavant, l'algorithme standard est trop lent pour les applications temps réel; c'est la raison pour laquelle nous considérons l'algorithme rapide SFM décrit dans [6]. Le SFM est basé sur deux approches: celle de la fonction signée et celle du cadre spécifique, inspirée de la NBM.

La première approche du SFM vise à employer le signe de la fonction de vitesse F au lieu de calculer sa valeur, comme donné dans les équations (1) et (2), [6], [10], [11], [12]. Comme le coût calculatoire de l'algorithme dépend en grande partie du temps nécessaire à la phase d'itération, le SFM diminue le temps de cette phase. Ainsi, pour tous les points, si le signe de F est positif, au lieu de calculer la valeur de Φ^{n+1} qui croît progressivement vers $+\infty$ au fur et à mesure de chaque itération, nous lui attribuons simplement une valeur positive arbitraire (+1); et si le signe de F est négatif, au lieu de calculer la valeur de Φ^{n+1} qui décroît progressivement vers

$-\infty$ au fur et à mesure de chaque itération, nous lui attribuons directement une valeur négative arbitraire (-1). Nous pouvons ainsi observer l'interface de propagation se déplacer rapidement au fil des itérations, jusqu'à atteindre, en quelques itérations, sa position finale correspondant à R_1 , lorsque plus aucun changement de Φ^n n'a lieu.

La deuxième approche du SFM, le cadre spécifique, est semblable à la NBM [2], [4] dans laquelle nous calculons l'évolution des courbes de niveau uniquement dans une zone spécifique de quelques pixels à l'intérieur et à l'extérieur de la courbe de niveau zéro, au lieu de calculer l'évolution des courbes de niveau en tous les points de l'image. En fait, cette approche est basée sur l'hypothèse que nous connaissons l'étendue maximale de mouvement d'une image à la suivante dans la séquence. L'évaluation des courbes de niveau est faite uniquement pour les points à l'intérieur de la zone spécifique qui est habituellement très petite par rapport à l'espace global. Néanmoins, le coût calculatoire de la gestion de cette zone étant important, il semble alors plus judicieux d'opter pour une zone rectangulaire avec une marge δ d'une dizaine de pixels autour de la courbe de niveau zéro. Le choix de δ réalisé de cette manière pour cette approche du cadre spécifique n'est pas optimal, car d'autres techniques plus sophistiquées peuvent être utilisées, mais il est suffisant si sa valeur n'est pas trop faible.

3. L'approche de voisinage

Pour expliquer l'approche de voisinage, nous commençons par présenter la technique employée pour calculer P_{in} et P_{out} telle qu'elle est donnée dans [10], avec la référence à la figure 1 et aux équations (1), (2) et (3). A partir de ces équations, on peut présenter l'équation de suivi sous la forme:

$$\begin{aligned} \frac{\partial \Phi}{\partial t} &= \left[\inf_{\{z: \|z\| \leq \delta, (x+z, y+z) \in R_0\}} (I^{n+1}(x,y) - I^n(x+z, y+z))^2 \right] \vec{\nabla} \Phi \\ &- \left[\inf_{\{z: \|z\| \leq \delta, (x+z, y+z) \in R_0^c\}} (I^{n+1}(x,y) - I^n(x+z, y+z))^2 \right] \vec{\nabla} \Phi + [\lambda_L \kappa_\Phi(x,y)] \vec{\nabla} \Phi \\ &= F_1 - F_2 + \alpha \end{aligned} \quad (4)$$

où λ_L est une constante et κ_Φ la fonction de courbure de la courbe fermée γ . α est une force liée à la courbure de γ et est assez petite. Généralement, α est négligée et la force F_1 au point (x,y) est trouvée en déterminant la valeur minimale de la différence entre les points $I^{n+1}(x,y)$ et $I^n(x+z, y+z)$ dans la zone $z: \|z\| \leq \delta, (x+z, y+z) \in R_0$. De même, la force F_2 au point (x,y) correspond à la valeur minimale de la différence entre les points $I^{n+1}(x,y)$ et $I^n(x+z, y+z)$ dans la zone $z: \|z\| \leq \delta, (x+z, y+z) \in R_0^c$. Ainsi la force finale sera correcte si la définition de R_0 est précise, ce qui permettra à l'algorithme de trouver la région R_1 correcte; dans le cas contraire, l'algorithme fournira des résultats erronés.

Dans le cas de notre approche, la force F_1 est maintenant définie comme la différence entre le point $I^{n+1}(x,y)$ et la moyenne des points $I^n(x+z, y+z)$ dans la zone définie par $z: \|z\| \leq \delta, (x+z, y+z) \in R_0$, puis la force F_2 comme la différence entre le point $I^{n+1}(x,y)$ et la moyenne des points $I^n(x+z, y+z)$ avec $z: \|z\| \leq \delta, (x+z, y+z) \in R_0^c$. L'effet introduit par des points erronés devient alors négligeable

puisque la moyenne des points dans la zone de recherche est proche de la valeur correcte, car nous supposons que le nombre de ces points mal placés - quelques points de R_0 situés dans R_1^c , ou quelques points de R_1^c dans R_0 - est plus petit que le nombre de points correctement classés dans la région de recherche.

4. Résultats expérimentaux

L'algorithme des courbes de niveau standard et l'algorithme SFM ont été implantés sur un PC 2 GHz équipé de Matlab. De nombreuses séquences d'images 8 bits ont été utilisées pour tester les algorithmes. Les temps d'exécution du SFM sont de l'ordre de 19 à 23 fois plus faibles que ceux de l'algorithme standard, suivant le type d'image à traiter, avec un nombre d'itérations environ 7 fois moins important.

La figure 2 illustre parfaitement la robustesse de l'algorithme SFM lié à l'approche de voisinage par rapport à l'algorithme SFM. A partir des deux images de synthèse, il semble très difficile de déterminer précisément les contours de la région R_0 dans l'image 1. A cause de ce problème d'initialisation, l'algorithme SFM seul est incapable de fournir un résultat cohérent avec la réalité (image gauche en bas). Par contre, notre approche permet de passer outre ces erreurs de détermination de R_0 et d'aboutir à un résultat tout à fait correct de R_1 (image droite en bas). A noter le faible nombre d'itération de l'algorithme SFM par rapport à l'algorithme standard.

Nous présentons ensuite les résultats obtenus sur deux séquences d'images réelles (Fig. 3 et 4). Avant de lancer le suivi d'objet, il faut passer par une phase d'initialisation du contour de l'objet. Celle-ci est réalisée grâce à une approche itérative de Fast Marching (FM) combinant l'information de gradient et celle de statistique de région présente dans l'image. Pour chaque séquence, seules trois images acquises à des instants différents sont présentées pour illustrer l'efficacité de notre algorithme. Bien que le résultat de l'initialisation ne soit pas tout à fait correct, l'algorithme arrive à suivre les objets de manière efficace dans les séquences avec un nombre restreint d'itérations.

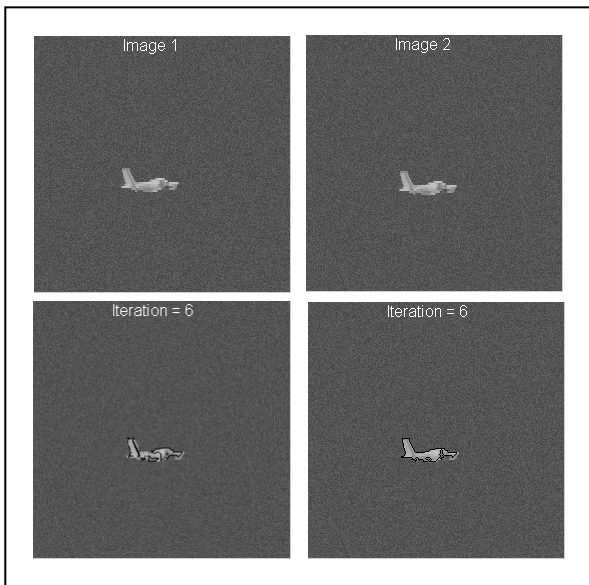


Fig.2 : Le résultat de l'algorithme SFM

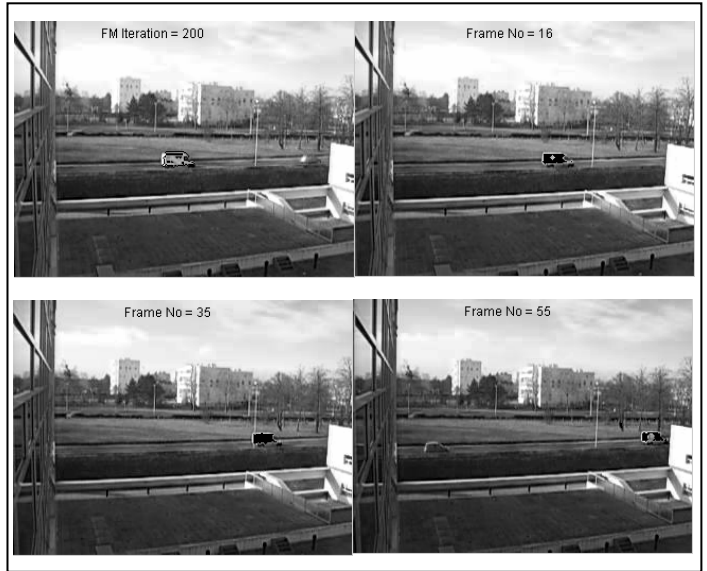


Fig.3 : Le résultat de suivi d'un objet dans une séquence d'images

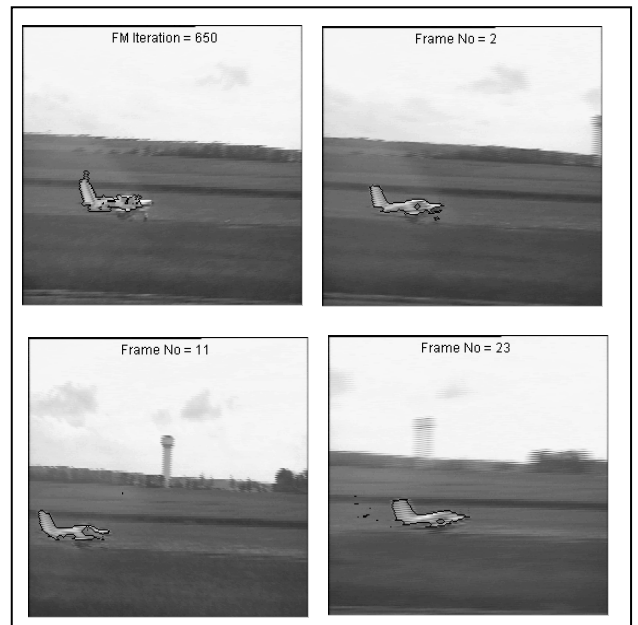


Fig.4 : Le résultat de suivi d'un objet dans une séquence d'images

5. Conclusion

Nous avons présenté ici une technique basée sur la méthode des Courbes de Niveau pour suivre un objet dans une séquence d'images. L'approche SFM utilisée dans l'algorithme a permis de réduire considérablement le nombre d'itérations nécessaire pour aboutir à notre résultat. L'association d'une approche de voisinage a amélioré la robustesse de l'algorithme de suivi.

Dans cette technique, la détermination initiale de l'objet à suivre n'est plus une étape primordiale pour obtenir des résultats cohérents. Les résultats expérimentaux valident la nouvelle approche et montrent son efficacité.

La prochaine étape est l'implantation de cette approche dans un système embarqué en vue d'un fonctionnement temps réel pour une application de suivi de cible.

Références

- [1] S. Osher, J. Sethian, "Fronts Propagating With Curvature Dependent Speed : Algorithms Based On Hamilton-Jacobi Formulations", *Jour. Computing Phys.* 79, pp.12-49, 1988.
- [2] J. Sethian, "*Level Set Methods*", Cambridge University Press, 1996.
- [3] R. Fedkiw, G. Sapiro, C. Shu, Shock Capturing, Level Sets, And Based Methods In Computer Vision And Image Processing: A Review Of Osher's Contributions", *Jour. of Computational Physics*, 185, pp. 309-341, 2003.
- [4] D. Adalsteinsson, J. Sethian, "A Fast Level Set Method For Propagating Interfaces", *Jour. Computing Phys.*, Vol. 118, pp. 269-277, 1995.
- [5] D. Peng, B. Merriman, S. Osher, H. Zhao, M. Kang, "A PDE-based fast local level set method", *Jour. Computing Phys.*, Vol. 155, November 1999.
- [6] A. Ganoun, R. Canals, "A new fast Level Set Method", *6th NORDIC Signal Processing Symposium*, 9 - 11/6/2004, ESPOO, FINLAND.
- [7] A. Mansouri, T. Chomaud, J. Konrad, "A Comparative Evaluation Of Algorithms For Fast Computation Of Level Set PDEs with Applications To Motion Segmentation", *In Proc. International Conf. on Image Processing*, vol III, pp. 636-639, 2001.
- [8] B. Nilsson, A. Heyden, "A Fast Algorithm For Level Set-Like Active Contours", *Pattern Recognition Letters* 24 (2003) 1331-1337.
- [9] S. Zhu, A. Yuille, "Region Competition : Unifying Snakes, Region Growing, and Bayes/MDL for Multi-band Image Segmentation", *IEEE Trans. Pattern Analysis And Machine Intelligence* Vol. 18, No 9, pp.884-900, Sept 1996.
- [10] A. Mansouri, "Region Tracking Via Level Set Pdes Without Motion Computation", *IEEE Trans. Pattern Analysis And Machine Intelligence* Vol. 24, No 7, July 2002.
- [11] C. Rasmussen, G. Hager, "Probabilistic data Association Methods for Tracking Complex Visual Objects", *IEEE Trans. Pattern Analysis And Machine Intelligence* Vol. 23, No 6, pp. 560-576, June 2001.
- [12] E. Sifakis, C. Garcia, and G. Tziritas, "Bayesian Level Sets for Image Segmentation", *Journal of Visual Communication and Image Representations*. No 13, 44-64, 2002.