

# Suivi automatique de personnes en mouvement par analyse d'images couleurs successives.

## Application au suivi de joueurs de football.

Nicolas Vandembroucke <sup>(1)</sup>, Ludovic Macaire <sup>(2)</sup> et Christophe Vieren <sup>(2)</sup>

<sup>(1)</sup>Ecole des Mines de Douai  
Centre de Recherche de Dorignies, Productique  
59500 Douai Dorignies, France

<sup>(2)</sup>Centre d'Automatique de Lille  
USTL, Cité Scientifique, Bâtiment P2  
59655 Villeneuve d'Ascq cedex, France

### RÉSUMÉ

Dans cet article, nous présentons une méthode de suivi de joueurs de football où les joueurs sont modélisés par des contours actifs évalués après classification couleur de chaque pixel. La position de chaque joueur extrait des images d'un match peut ainsi être déterminée sauf si ce joueur est caché par un adversaire. Dans ce cas, la reconnaissance de l'équipe des joueurs par classification des pixels permet le suivi automatique de chacun d'eux. Cet algorithme utilise un apprentissage supervisé basé sur une analyse colorimétrique des tenues des joueurs.

### ABSTRACT

In this paper, we present a soccer player tracking method where the players are modeled by active contours evaluated after color classification of each pixel. The position of each player extracted from images of a match can be so determined except if this player is hidden by another one. In this case, the recognition of the players team by pixels classification allows the automatic tracking of each of them. This algorithm uses a supervised learning scheme based on the colorimetric analysis of the players soccer suits.

## 1 Introduction

Un match de football comporte de tels enjeux économiques que les entraîneurs veulent disposer d'un maximum d'informations sur son déroulement afin d'affiner leur stratégie. Quelques auteurs se sont ainsi proposés de concevoir des systèmes d'analyse d'images d'un match de football retransmises par une chaîne de télévision et enregistrées sur une bande vidéo [3, 9].

Nous présentons un système qui suit automatiquement chaque joueur à partir d'images couleurs successives acquises par une caméra fixe visant le terrain de football. Ainsi, il sera possible d'évaluer des statistiques sur le comportement de chaque joueur.

Un joueur, extrait dans l'image par un algorithme décrit ultérieurement, est modélisé par un modèle de contour actif ou snake. Chaque contour actif est mis à jour au fil des images dans le but de déterminer le mouvement des joueurs [8]. Quand, lors d'une séquence d'images, un joueur est caché par un autre dans le champ de la caméra, leur modélisation par contours actifs n'est plus possible et nécessite alors une réinitialisation interactive. Dans la majorité des cas, deux joueurs en contact n'appartiennent pas à la même équipe. C'est pourquoi, afin d'affecter automatiquement chaque contour actif à un joueur, nous proposons qu'ils soient évalués après classifi-

cation couleur des pixels selon deux classes correspondant à chacune des deux équipes [7].

En vue de la détection des défauts de surface de bois, Adel [1] et Brunner [2] recherche le système de représentation de la couleur le plus discriminant. Nous proposons de déterminer automatiquement, parmi un ensemble de systèmes de représentation, les trois composantes qui discriminent le mieux les deux équipes. Ces trois composantes forment un « espace couleur hybride ». Dans une première partie, nous décrivons l'apprentissage supervisé grâce auquel nous déterminons l'espace couleur hybride pour un match donné. Puis, nous détaillons, dans une deuxième partie, l'algorithme de classification des pixels dans cet espace. Les résultats sont exploités par une méthode de suivi des joueurs à partir de contours actifs décrite dans la troisième partie. Enfin, nous présentons des résultats dans la dernière partie afin d'illustrer notre démarche.

## 2 L'apprentissage supervisé

Dans un premier temps, nous dissociions les pixels représentant la pelouse de ceux représentant les joueurs : les « pixels joueurs ». Afin de modéliser la pelouse, nous sélectionnons de façon interactive dans une image d'apprentissage une fenêtre de pelouse notée  $W$ . Puis, différentes fenêtres conte-

nant uniquement les pixels joueurs d'un seul joueur sont sélectionnées interactivement sur l'image d'apprentissage segmentée. Ces « fenêtres joueurs » contiennent donc des pixels qui sont des prototypes de chacune des deux classes, à savoir les deux équipes. L'analyse colorimétrique de leurs tenues permet de déterminer l'espace couleur hybride formé des trois composantes chromatiques apportant la meilleure séparation des deux classes en présence. Enfin, nous spécifions les attributs colorimétriques des deux classes.

## 2.1 Extraction des pixels joueurs

Dans la fenêtre  $W$ , les histogrammes des composantes chromatiques rouge et verte normalisées, notées  $r$  et  $g$ , sont unimodaux. De cette façon, nous pouvons déterminer quatre seuils correspondant aux limites droites et gauches des modes des deux histogrammes.

Pour tout pixel  $P(x, y) \in W$ , nous calculons :

le seuil bas de la composante  $r$  :  $S_l^r = \min [r(x, y)]$ .

le seuil haut de la composante  $r$  :  $S_h^r = \max [r(x, y)]$ .

le seuil bas de la composante  $g$  :  $S_l^g = \min [g(x, y)]$ .

le seuil haut de la composante  $g$  :  $S_h^g = \max [g(x, y)]$ .

Pour chaque image de la séquence, la segmentation est ensuite réalisée dans le plan chromatique  $(r, g)$  selon la règle de décision suivante :

Soit un pixel  $P(x, y)$  : si  $r(x, y) \notin [S_l^r, S_h^r]$  et  $g(x, y) \notin [S_l^g, S_h^g]$ , alors le pixel  $P(x, y)$  est un pixel joueur.

Si la pelouse est uniforme, la sélection d'une seule fenêtre est suffisante à la segmentation des images couleurs. Par contre, si elle présente des dégradations, des ombres, des variations de couleurs, etc, alors nous sélectionnons autant de fenêtres pelouse que celle-ci présente d'aspects différents.

## 2.2 Détermination de l'espace couleur hybride

Une image couleur numérisée issue du système  $(R, G, B)$  peut être convertie vers d'autres systèmes afin d'en dégager de nouvelles caractéristiques. Pour notre étude, nous retenons un ensemble de systèmes les plus couramment utilisés [5] : le système de couleurs normalisées  $(r, g, b)$ , le système de perception de la couleur humain  $(I, H, S)$ , les systèmes de coordonnées couleurs de la CIE  $(X, Y, Z)$  et  $(x, y, z)$ , les systèmes couleurs uniformes de la CIE  $(L^*, a^*, b^*)$ ,  $(L^*, u^*, v^*)$  et leurs systèmes dérivés  $(L_{uv}^*, C_{uv}^*, h_{uv}^\circ)$ ,  $(L_{ab}^*, C_{ab}^*, h_{ab}^\circ)$  et  $(L_{uv}^*, S_{uv}^*, h_{uv}^\circ)$ , le système d'Otha  $(I1, I2, I3)$ , celui de Faugeras  $(A, C1, C2)$  et les systèmes de coordonnées couleurs pour les signaux de télévision  $(Y, I, Q)$  et  $(Y, U, V)$ . Dans cet ensemble de systèmes de représentation de la couleur, nous voulons déterminer les trois composantes les plus discriminantes vis-à-vis des deux classes en présence selon le match considéré. Pour cela, nous établissons les notations suivantes : Soient  $C_1$  et  $C_2$ , les deux classes de pixels joueurs qui correspondent respectivement aux équipes 1 et 2.

Soit  $\omega_{i,j}$ , une fenêtre joueur sélectionnée interactivement sur l'image d'apprentissage segmentée où  $i$  indique le numéro de la fenêtre et  $j$  le numéro de l'équipe.  $\Omega$ , l'ensemble de ces fenêtres est l'union de  $\Omega_1$  et  $\Omega_2$ , les ensembles respec-

tifs des fenêtres joueurs  $\omega_{i,1}$  et  $\omega_{i,2}$ . Nous sélectionnons le même nombre  $N_\Omega/2$  de fenêtres joueurs pour l'équipe 1 et pour l'équipe 2.

Soit  $\Pi = (R, G, B, r, g, b, I, H, S, X, Y, Z, x, y, z, L^*, a^*, b^*, u^*, v^*, I2, I3, A, C1, C2, Y, I, Q, U, V, C_{uv}^*, h_{uv}^\circ, S_{uv}^*, C_{ab}^*, h_{ab}^\circ)$ , l'espace multidimensionnel défini par l'ensemble des composantes formant les systèmes couleurs présentés précédemment. Un pixel joueur  $P(x, y)$  est représenté par un point dans l'espace  $\Pi$  dont la  $k^{ième}$  coordonnée est  $\pi_k(x, y)$ . Les points dans  $\Pi$  associés aux pixels joueurs extraits d'une fenêtre joueur  $\omega_{i,j} \in \Omega_j$  sont considérés comme un sous-nuage, noté  $SC_{i,j}$ . Le nuage de points de la classe  $C_j$  est alors défini en groupant les  $N_\Omega/2$  sous-nuages  $SC_{i,j}$  correspondant aux  $N_\Omega/2$  fenêtres joueurs de  $\Omega_j$ .

Soit  $P_{\omega_{i,j}}$  l'ensemble des pixels joueurs d'une fenêtre joueur  $\omega_{i,j} \in \Omega_j$ , nous calculons les moyennes des pixels joueurs selon chaque composante  $\pi_k(x, y)$  :

$$m_{\omega_{i,j}}^{\pi_k} = \sum_{P(x,y) \in P_{\omega_{i,j}}} \left( \frac{\pi_k(x, y)}{\text{card}(P_{\omega_{i,j}})} \right) \quad (1)$$

$m_{\omega_{i,j}}^{\pi_k}$  représente la  $k^{ième}$  coordonnée du centre de gravité du nuage  $SC_{i,j}$  évalué avec une pondération selon la fonction de densité de chaque point dans la fenêtre  $\omega_{i,j}$ .

Dans le but d'évaluer le pouvoir discriminant de chacune des composantes couleurs de l'espace  $\Pi$ , nous évaluons  $D_{min}^{\pi_k}$ , la différence minimum entre les  $k^{ième}$  coordonnées des centres d'un couple de sous-nuages de classes différentes :

$$D_{min}^{\pi_k} = \min \left( \left| m_{\omega_{i,1}}^{\pi_k} - m_{\omega_{i',2}}^{\pi_k} \right| \right) \quad (2)$$

où  $i$  et  $i'$  varient de 1 à  $N_\Omega/2$ . De même, nous calculons  $D_{max}^{\pi_k}$ , pour un couple de sous-nuages d'une même classe.

$$D_{max}^{\pi_k} = \max \left( \left| m_{\omega_{i,j}}^{\pi_k} - m_{\omega_{i',j}}^{\pi_k} \right| \right) \quad (3)$$

Finalement, le pouvoir discriminant de chacune des composantes couleur est issu du critère  $D^{\pi_k} = D_{min}^{\pi_k} - D_{max}^{\pi_k}$ . Sa valeur maximale indique la composante chromatique la plus discriminante. Un nouvel espace couleur hybride appelé  $(\pi_1, \pi_2, \pi_3)$  réunit les trois composantes les plus discriminantes.

## 2.3 Les attributs des classes

Pour chaque classe  $C_j$ , nous déterminons trois attributs noté  $M_j^{\pi_1}, M_j^{\pi_2}, M_j^{\pi_3}$ . Ce sont les trois moyennes, évaluée selon chaque composante de l'espace couleur hybride  $(\pi_1, \pi_2, \pi_3)$ , de tous les pixels joueurs appartenant à toutes les fenêtres  $\omega_{i,j}$ .

$$M_j^{\pi_k} = \frac{2}{N_\Omega} \times \sum_{i=1}^{i=N_\Omega/2} m_{\omega_{i,j}}^{\pi_k} \quad (4)$$

## 3 Classification des pixels joueurs

La méthode proposée classe les pixels joueurs d'une image couleur préalablement segmentée selon la règle de décision vue au paragraphe 2.1. La classification est basée sur une distance spécifique évaluée dans l'espace couleur hybride entre un pixel joueur et les attributs d'une classe.

### 3.1 Distance spécifique entre un pixel joueur et les attributs d'une classe

De l'image d'apprentissage, nous retenons, pour chacune des deux classes  $C_j$ , une fenêtre joueur  $\omega_{0,j} \in \Omega_j$ . Dans chacune d'elles, nous définissons une région  $RE_j$ , constituée par les pixels joueurs de  $\omega_{0,j}$  et un centre de gravité  $G_j$ , le centre de  $\omega_{0,j}$ . La région  $RE_j$  peut être interprétée comme une forme générique des joueurs de l'équipe  $j$  dans les images couleurs. Pour tout pixel joueur  $P(x, y)$  d'une image à analyser, nous évaluons la région d'intersection  $RE_I$  entre  $RE_j$  et  $RE_P$ , région de centre  $P(x, y)$  constituée des pixels joueurs de l'image. La région  $RE_j$  est alors translatée de sorte que son centre de gravité coïncide avec le pixel joueur  $P(x, y)$  (cf. figure 1).

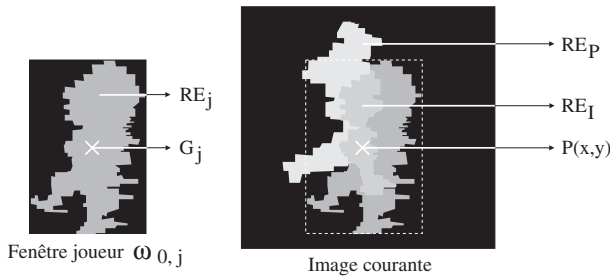


FIG. 1 — Région d'intersection

Nous définissons alors, dans la région  $RE_I$  de surface  $S(RE_I)$ , la distance  $D_j(x, y)$  entre les valeurs  $\pi_k(x, y)$  des composantes de l'espace couleur hybride du pixel joueur  $P(x, y)$  et les attributs de la classe  $C_j$  :

$$D_j(x, y) = \frac{1}{3} \times \sum_{k=1}^{k=3} \left( \left| M_j^{\pi_k} - \sum_{P(x,y) \in RE_I} \frac{\pi_k(x, y)}{S(RE_I)} \right| \right) \quad (5)$$

### 3.2 Règle de décision

Pour chaque pixel joueur  $P(x, y)$ , la règle de décision est basée sur la comparaison entre les deux distances correspondant aux deux classes. Selon le résultat de la comparaison, le pixel joueur est assigné à l'une ou à l'autre des deux classes. Deux images  $I_1$  et  $I_2$ , correspondant aux équipes 1 et 2 sont alors créées de la façon suivante :

Si  $D_1(x, y) < D_2(x, y)$  alors  $P(x, y) \in C_1$ ,  $I_1(x, y) = 1$  et  $I_2(x, y) = 0$ .

Si  $D_1(x, y) > D_2(x, y)$  alors  $P(x, y) \in C_2$ ,  $I_1(x, y) = 0$  et  $I_2(x, y) = 1$ .

Si  $D_1(x, y) = D_2(x, y)$  alors  $P(x, y) \in \emptyset$ ,  $I_1(x, y) = 0$  et  $I_2(x, y) = 0$ .

Les images  $I_1$  et  $I_2$  de la figure 2 correspondent au résultat de la classification des pixels de l'image  $I$ .

## 4 Le suivi des joueurs

Classiquement, le suivi consiste en l'appariement des primitives extraites des images successives. Le problème devient rapidement complexe lorsqu'un objet peut être représenté par

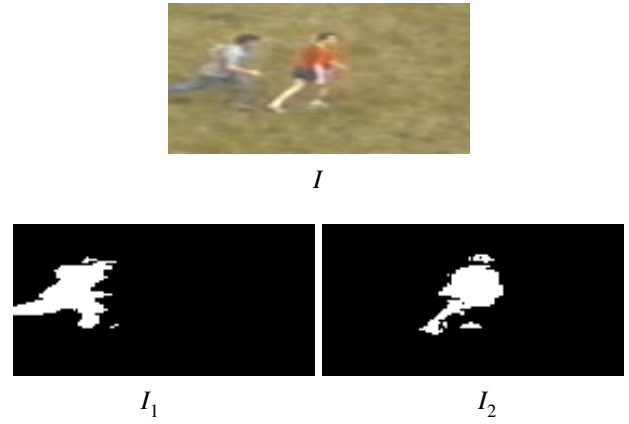


FIG. 2 — Résultats de la classification

plusieurs primitives et réciproquement lorsqu'une primitive peut représenter plusieurs objets. Dans notre cas, l'aspect déformable des joueurs, lié essentiellement au mouvement de leurs membres, accroît la difficulté de l'appariement. C'est pourquoi, nous employons une approche globale consistant à associer à chacun des joueurs un contour actif. Ceci permet de réaliser simultanément les phases de modélisation et d'appariement.

### 4.1 Le modèle de contour actif

Le modèle de contour actif retenu est celui proposé par Kass et Terzopoulos [4]. Ce modèle est défini comme une courbe fermée représentée paramétriquement par  $V(s) = \{x(s), y(s)\}^T$  avec  $s \in [0, 1]$ , et  $V(0) = V(1)$ . Une fonctionnelle d'énergie est définie en chaque point  $V(s)$  de cette courbe comme la somme de deux termes : d'une part, l'énergie interne  $E_{int}$  composée elle-même de deux termes contrôlant respectivement l'« élasticité » et la « flexion » de la courbe et, d'autre part, l'énergie externe  $E_{ext}$  destinée à obtenir la convergence du contour actif vers les objets à modéliser. On considère que le contour actif est une bonne description des objets lorsque la fonctionnelle d'énergie  $E_{snake} = E_{int} + E_{ext}$  est minimale.

L'influence relative de l'élasticité et de la flexion durant le processus de minimisation de l'énergie  $E_{snake}$  est pondérée par les paramètres  $\alpha$  et  $\beta$  de sorte que :

$$E_{int} = \alpha \times \int_0^1 \|V_s(s)\|^2 ds + \beta \times \int_0^1 \|V_{ss}(s)\|^2 ds \quad (6)$$

où  $V_s(s)$  et  $V_{ss}(s)$  représentent respectivement les dérivées curvilignes première et seconde de  $V(s)$ . Afin que le contour actif soit attiré par le contour des objets à modéliser, l'énergie externe doit présenter un minimum sur ces contours. C'est pourquoi, elle est calculée en utilisant l'amplitude du gradient de l'image. Dans notre cas, cette image est soit  $I_1$ , soit  $I_2$  selon l'équipe à laquelle appartient les joueurs suivis. Ainsi, l'évaluation de l'énergie externe d'un contour actif associé à un joueur de l'équipe 1 n'est pas sensible à la proximité d'un joueur de l'équipe 2.

## 4.2 Initialisation

L'inconvénient bien connu de l'utilisation des modèles de contours actifs est le processus d'initialisation. Dans notre cas, les joueurs entrant séparés sur le terrain, ce processus devient aisé : chaque contour actif est initialisé grâce à un chaînage des points de contour des régions contenues dans les images  $I_1$  et  $I_2$ .

## 4.3 Suivi

Le suivi des joueurs s'obtient par l'estimation du contour actif associé à chacun d'eux dans chaque image de la séquence. A partir du contour actif estimé dans l'image précédente, on réalise une prédiction de sa position dans l'image courante en supposant le joueur animé de mouvements uniformément accélérés entre deux images successives. Cependant, afin de prendre en compte les possibles variations d'accélération, cette prédiction du second ordre est complétée par un 'gonflement' du contour actif. Ce gonflement est obtenu facilement en remplaçant durant cette phase l'énergie du contour actif par un terme qui permet de contrôler l'aire de ce dernier [6].

Après cette phase de prédiction, la phase d'estimation est réalisée en recherchant l'énergie minimale du contour actif. Parmi les différentes méthodes de recherche du minimum proposées dans la littérature, nous avons utilisé la méthode de la descente du gradient, réputée peu stable mais rapide. Néanmoins, les images dont nous disposons pour évaluer l'énergie externe étant binaires, cette méthode demeure satisfaisante.

## 5 Résultats expérimentaux

En ce qui concerne la procédure de classification des pixels, on se référera à [7] qui atteste de sa robustesse.

Pour illustrer les résultats obtenus dans le cadre du suivi de footballeurs, nous présentons un extrait de séquence de scène réelle. Les images a, b, c et d de la figure 3 sont référencées de façon chronologique. La courbe entourant les deux joueurs matérialise leur contour actif associé. Nous constatons l'efficacité de la méthode dans le cas d'un masquage quasi total.

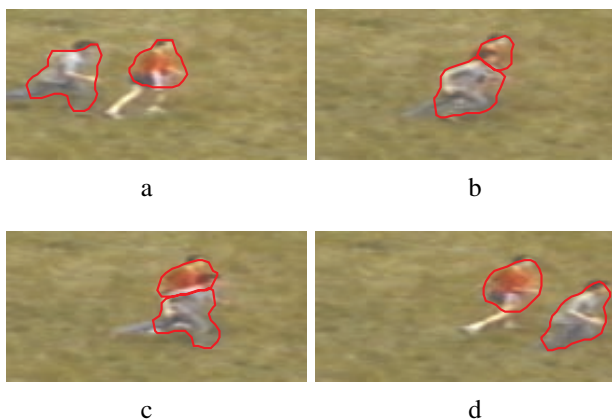


FIG. 3 — Résultats

## 6 Conclusion

L'approche proposée, couplant un algorithme de classification colorimétrique à une méthode de suivi par contours actifs, permet pour notre application de résoudre une large majorité des cas d'occlusions. Les cas d'occlusions restants pourraient être résolus par une fusion d'informations issues d'une analyse stéréoscopique ainsi que d'une analyse de mouvement. Nous travaillons actuellement sur la généralisation de la méthode à plus de deux classes afin d'envisager d'autres applications.

## Références

- [1] M. Adel, D. Wolf, R. Husson, and R. Vogrig. Evaluation of color spaces in computer vision : an application of wood defects detection. In *International conference on systems man and cybernetics*, pages 499–504, Le Touquet, 1993.
- [2] C. C. Brunner, A. G. Maristany, D. A. Butler, D. Vanleuween, and J. W. Funck. An evaluation of color spaces for detecting defects in douglas-fir veneer. *Industrial metrology*, 2 :169–184, 1992.
- [3] Y. H. Gong, H. C. Chua, and T. S. Lim. An automatic video parser for TV soccer games. In *Second asian conference on computer vision*, pages 509–513, Singapore, 1995.
- [4] M. Kass, A. Witkin, and D. Terzopoulos. Snakes : Active contour models. *International journal of computer vision*, 3 :321–331, 1988.
- [5] Q. T. Luong. La couleur en vision par ordinateur : une revue. *Traitement du signal*, 8 :548–555, 1991.
- [6] N. Rougon. Kinematics of interface evolution with application to active contour models. In *Curves and surfaces in computer vision and graphics II*, volume 610, pages 336–348, Boston, 1992.
- [7] N. Vandenbroucke, L. Macaire, and J. G. Postaire. Soccer player recognition by pixels classification in an hybrid color space. In *Multispectral and hyperspectral imagery III*, Orlando, 1997.
- [8] C. Vieren, F. Cabestaing, and J. G. Postaire. Catching moving objects with snakes for motion tracking. *Pattern recognition letters*, 16 :679–685, 1995.
- [9] D. Yow, B. L. Yeo, M. Yeung, and B. Liu. Analysis and presentation of soccer highlights from digital video. In *Second asian conference on computer vision*, pages 499–503, Singapore, 1995.