Découpage automatique de vidéos de sport amateur par détection de personnes et analyse de contenu colorimétrique

Axel Baldanza, Jean-François Aujol, Yann Traonmilin, François Alary

Découpage automatique de vidéos de sport amateur par détection de personnes et analyse de contenu colorimétrique

Axel Baldanza[1,2,*], Jean-François Aujol[2], Yann Traonmilin[2], and François Alary[1]

[1]Rematch
[2]Univ. Bordeaux, Bordeaux INP, CNRS, IMB, UMR 5251,F-33400 Talence, France.
[*]Contacter l'auteur : a.baldanza@rematch.fr

Résumé

Cet article présente une méthode de recherche de bornes de découpe pour des vidéos de sport visant à supprimer les sous-séquences indésirables de début et de fin. Une séquence est définie comme indésirable lorsque la caméra n'est pas dirigée vers le terrain. L'objectif est de trouver les bornes de début et de fin de la nouvelle séquence à conserver à partir d'une vidéo de K images en entrée. Notre méthode optimise une fonction objectif construite à partir de différents descripteurs sur les images adaptés à la discrimination de sous-séquences inutiles.

Mots Clefs

Découpe automatique, vidéos sport

Abstract

This article presents a searching method for trimming bounds in sport videos which aims at removing unwanted subsequences at the start and at the end. A sequence is labelled as unwanted if the camera is not pointed at the sport field. The goal is to find two bounds respectively for the beginning and the end of the new sequence we want to keep from a K frame video. Our method optimizes a loss function built from different descriptors adapted from the literature for the discrimination of subsequence uselessness.

Keywords

Automatic trimming, sport videos

1 Introduction

L'analyse du contenu vidéo dans le sport est devenue depuis plusieurs années un sujet important en vision par ordinateur, pour pouvoir étudier et améliorer les performances des sportifs, mais aussi pour améliorer la qualité des images (vue à 360°, caméra isolée, suivi des joueurs ...). Les systèmes modernes d'analyse dans le sport professionnel sont équipés de caméras puissantes, ou de caméras multiples qui permettent de réaliser des détections rapides et performantes. [11, 15, 19] présentent des méthodes de détection de joueurs ou d'analyse de contenu qui utilisent des caméras haute définition avec des angles de vue qui facilitent cette détection. Pour le sport amateur, filmé avec des caméras de téléphones portables hétérogènes et depuis des angles de vue variables, il est nécessaire de développer une méthode robuste à ce type de changement.

Dans cet article, on propose un algorithme de recherche de bornes de découpe pour des vidéos de sport amateur, visant à supprimer les sous-séquences indésirables des vidéos. Une image est considérée comme indésirable si elle n'est pas composée du terrain et des joueurs (la figure 8 permet de distinguer les images à conserver des images indésirables). L'algorithme prend en entrée une vidéo de sport amateur et retourne les bornes de découpe qui correspondent au début et à la fin de la nouvelle vidéo. On considère ici les sports collectifs (le football, le rugby, le basketball, le handball, le volley ...). Une sous-séquence indésirable peut se situer n'importe où dans la vidéo. Pour éviter les faux raccords, la méthode ne peut supprimer une sous-séquence que si elle se situe à une extrémité de la vidéo. Une sous-séquence indésirable située entre deux sous-séquences à conserver ne doit pas être supprimée et le nombre maximum de bornes à déterminer est 2. Cette application a pour objectif d'obtenir un maximum de vidéos bien coupées tout en gardant un nombre de vidéos endommagées très faible. Une des principales contraintes imposées à ce travail concerne le temps d'exécution, qui doit être en moyenne inférieur à 15 secondes pour limiter la charge de l'algorithme sur les serveurs. Les descripteurs utilisés doivent donc être simples et efficaces.

Cet article présente comment l'algorithme discrimine, grâce à des descripteurs adaptés, les images que l'on souhaite supprimer et celles que l'on veut garder. La figure 8 illustre le contenu des vidéos situé au milieu des sous-séquences définies par chaque descripteur. Une des approches pour les différencier consiste à détecter les joueurs présents dans l'image [5, 18] : si aucun joueur n'est détecté, on est dans le cas d'une image indésirable. D'autres approches étudient les caractéristiques colorimétriques des images [16], qui varient selon que l'image doit être supprimée ou non, qui permettent de renforcer les performances de l'algorithme pour qu'il vérifie les objectifs sur le nombre de vidéos endommagées tout en respectant la contrainte de temps de traitement. Ce document décrit ensuite l'étape de recherche des bornes de découpe, grâce à l'analyse temporelle des descripteurs précédents, afin de trouver les endroits des plus fortes variations des descripteurs que l'on considère comme les probables bornes de transition entre deux sous-séquences.

1.1 Notations générales

Soit I une image de taille $N \times M$ pixels, $[\![1; K]\!]$ l'ensemble des entiers compris entre 1 et K. Considérons I comme une fonction définie par :

$$\begin{array}{rccc} I & : & [\![1;N]\!] \times [\![1;M]\!] & \to & \mathbf{R}^3 \\ & & (x,y) & \mapsto & I(x,y). \end{array} \quad (1)$$

Considérons une vidéo comme composée de K images. I_j est l'image de taille $N \times M$ définie comme ci-dessus qui représente la j-ème image de la vidéo, $j \in [\![1; K]\!]$. Les indices R, G et B : I^R, I^G et I^B permettent dans le cas où les calculs sont faits sur un seul canal couleur et non toute l'image, de préciser de quel canal il s'agit.

2 Analyse du contenu par descripteurs

On définit

$$\begin{array}{rccc} D & : & [\![1;K]\!] & \to & [0;1] \\ & & j & \mapsto & D(j) \end{array} \quad (2)$$

un opérateur qui associe à la j-ème image de la vidéo une valeur qui indique si son contenu est indésirable ou à conserver.

2.1 Détection de personnes

Les méthodes de détection de personnes classiques présentées dans [2, 6, 7] n'étant pas assez efficaces dans notre modèle (joueurs parfois trop loin, ou image de trop basse qualité), notre méthode combine la détection de personnes selon deux principes : les histogrammes de gradient orientés (HOG) [5] et les cascades de Haar [18]. Soit $l, l' \in \mathbb{N}$ respectivement le nombre de personnes détectées par ces méthodes, on définit le descripteur principal

$$D_1(j) = \mathbb{1}_{l_j + l'_j > 0}, \quad (3)$$

avec $\mathbb{1}$ la fonction indicatrice définie par :

$$\mathbb{1}_{l_j+l'_j>0} = \begin{cases} 1 & \text{si } l_j + l'_j > 0. \\ 0 & \text{sinon.} \end{cases} \quad (4)$$

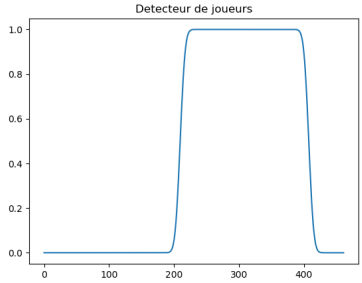


FIGURE 1 – Variations du détecteur de joueurs D_1 sur une vidéo qui doit être découpée.

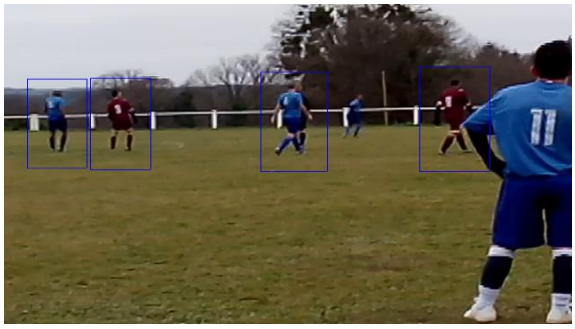
La figure 1 montre le résultat du détecteur de joueurs sur une vidéo composée de deux sous-séquences indésirables : il ne détecte des joueurs qu'entre les images 200 et 420.
La figure 2 montre des images dans lesquelles des joueurs ont été détectés par les deux méthodes ainsi que la position des joueurs détectés. Les figures 2c et 2d permettent de comprendre l'intérêt de la méthode : la mauvaise qualité de l'image rend la détection impossible pour la méthode de Haar, et seule la méthode basée sur les HoG a réussi à trouver des joueurs. Les figures 3a et 3b montrent des images dans lesquelles aucun joueur n'est détecté.
Pour les sports en salle, où les joueurs sont plus proches de la caméra, les performances de ce descripteur lui permettent d'être utilisé seul.

2.2 Descripteurs colorimétriques

Pour les sports en extérieur, on définit des descripteurs colorimétriques adaptés de [16] qui permettent de valider les bornes trouvées par D_1.

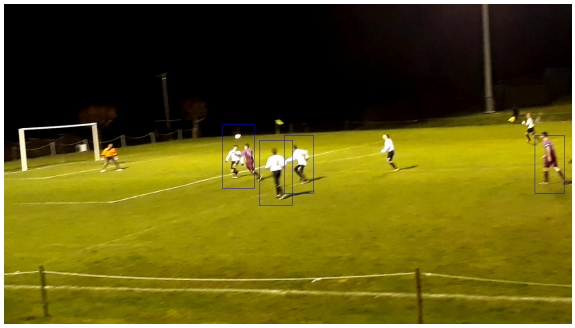
Couleur du ciel. Soit $N \in \mathbb{N}$ la largeur de l'image I_j. D_2 est défini comme la moyenne du canal couleur bleu de l'image dans la bande horizontale des $p \in \mathbb{N}$ pixels en haut de l'image :

$$D_2(j) = \frac{1}{p \times N} \sum_{x=1}^{N} \sum_{y=1}^{p} I_j^B(x,y). \quad (5)$$



(a)

(b)



(c)

(d)

FIGURE 2 – (a),(c) : Joueurs détectés par la méthode HoG [5]. (b),(d) : Joueurs détectés par la méthode cascades de Haar [18].

(a)

(b)

FIGURE 3 – Images dans lesquelles aucun joueur n'est détecté.

On constate dans l'image 4a que la valeur de D_2 est nettement supérieure à celle de la figure 4b. L'objectif de ce descripteur est de détecter le ciel dans l'image afin de différencier les images qui ne contiennent que le sol de celles où le ciel est présent.

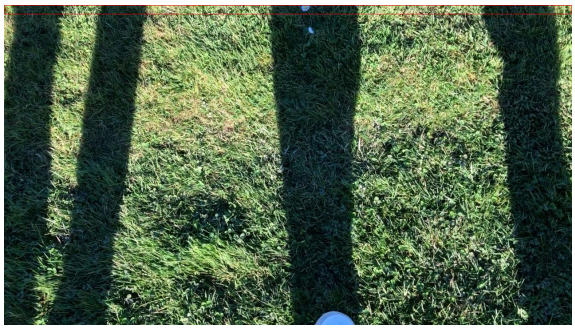
Différence entre le haut et le bas de l'image. L'objectif de ce descripteur est d'analyser la différence d'intensité entre le haut de l'image et le bas, qui est plus petite lorsque l'image contient uniquement du sol que lorsque l'image contient du sol et du ciel. Soit $M \in \mathbb{N}$ la hauteur de l'image, D_3 calcule la différence de moyennes entre le premier et le dernier quart horizontal de l'image I_j

$$D_3(j) = \frac{1}{3N\frac{M}{4}} \sum_{x=1}^{N} \sum_{y=1}^{M/4} \sum_{C=\{R,G,B\}} I_j^C(x,y) - I_j^C\left(x, y + \frac{3M}{4}\right). \quad (6)$$

Dans les images 5a et 5b, le rectangle rouge représente le premier quart horizontal de l'image, et le rectangle bleu le dernier.

Variation du contenu. Dans cette section, nous proposons un descripteur qui permet d'étudier les variations du contenu dans les séquences. Soit $H_c(j) \in \mathbb{R}^b$ l'histogramme cumulé calculé sur

(a) Sur cette image, $D_2(I) = 0.89$.



(b) Sur cette image, $D_2(I) = 0.35$.

FIGURE 4 – Calcul de la détection de ciel D_2 sur une image à conserver et une image à supprimer.

(a) Sur cette image, $D_3(I) = 0.72$.

(b) Sur cette image, $D_3(I) = 0.09$.

FIGURE 5 – Calcul de la différence entre premier et dernier quart horizontal D_3 sur une image à conserver et une image à supprimer.

l'image I_j, b le nombre d'intervalles utilisés pour construire l'histogramme. On définit $d_{H_c}(j) \in \mathbb{R}$ la différence entre l'histogramme cumulé d'une image et celui de l'image précédente

$$d_{H_c}(j) = ||H_c(j) - H_c(j-1)||_1. \tag{7}$$

Ce descripteur cherche à détecter un fort changement entre les images successives de la vidéo qui correspond à une transition entre une sous-séquence indésirable et une à conserver. Soit G un noyau gaussien défini par

$$G(x,s) = \frac{1}{2\pi s} e^{-x^2/2s}. \tag{8}$$

Soit

$$\begin{array}{rcccl} \Phi & : & [\![1;K]\!] & \rightarrow & \mathbb{R} \\ & & j & \mapsto & \Phi(j) = \sum_{t=1}^{j} (G * d_{H_c})(t) \end{array} \tag{9}$$

le résultat de l'intégration entre 1 et j du produit de convolution entre d_{H_c} et le noyau gaussien G donné par (8).

Pour augmenter la robustesse du descripteur, on définit D_4 de manière à normaliser ses variations en fonction de la valeur de la dérivée de $\Phi(j)$. On commence par définir D_4' par

$$D_4'(j) = \begin{cases} 1 & \text{si } |\Phi'(j)| > \mu. \\ 0 & \text{sinon.} \end{cases} \tag{10}$$

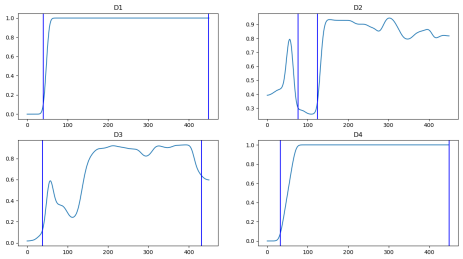
avec $\mu \in \mathbb{R}$ un seuil fixé. Le descripteur D_4 est l'intégration de D_4' normalisée entre 0 et 1, c'est-à-dire

$$D_4(j) = \frac{\sum_{i=1}^{j} D_4'(i)}{\sum_{i=1}^{K} D_4'(i)}. \tag{11}$$

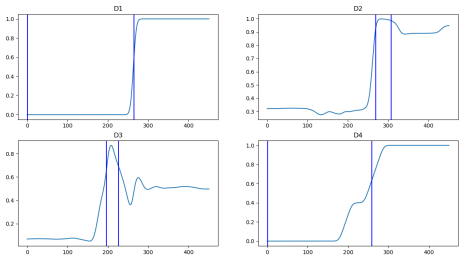
3 Recherche de bornes

Soit $\delta_k(j_1, j_2) = \frac{1}{j_2 - j_1 + 1} \sum_{j=j_1}^{j_2} D_k(j)$ la fonction qui renvoie la moyenne du descripteur D_k sur une sous-séquence comprise entre les images $j_1, j_2 \in [\![1;K]\!]^2$. On définit les bornes de transition $\hat{j}_{1_k}, \hat{j}_{2_k} \in [\![1;K]\!]^2$ du descripteur D_k comme les valeurs qui maximisent les différences de moyennes entre les sous-séquences du descripteur :

$$\begin{aligned} (\hat{j}_{1_k}, \hat{j}_{2_k}) = \underset{j_1, j_2}{\text{argmax}} \; & ||\delta_k(1, j_1) - \delta_k(j_1, j_2)||^2 \\ & + ||\delta_k(j_2, N) - \delta_k(j_1, j_2)||^2. \end{aligned} \tag{12}$$



(a)



(b)

FIGURE 6 – Variations de chaque descripteur sur deux vidéos (a) et (b).

On maximise l'équation (12) par recherche exhaustive. On conserve les sous-séquences définies par D_1 qui contiennent des joueurs. Pour les sports en salle, l'algorithme utilise la détection de personnes seule pour déterminer les bornes de découpe. On maximise (12) sur D_1 seulement. Pour les sports en extérieur, (12) est maximisée sur le détecteur de joueurs et les descripteurs colorimétriques. Une borne de découpe déterminée par D_1 est conservée si et seulement si l'équation (12) a trouvé une borne identique à $\Delta \in \mathbb{N}$ images près sur un ou plusieurs descripteurs colorimétriques. On définit $J_1, J_2 \in [\![1; K]\!]^2$ les bornes de découpe en extérieur par :

$$J_1 = \begin{cases} \hat{j}_{11} & \text{si } \min\limits_{\substack{i\in\{1,2\} \\ k\in\{2,\dots,4\}}} |\hat{j}_{11} - \hat{j}_{ik}| < \Delta. \\ 0 & \text{sinon.} \end{cases} \tag{13}$$

$$J_2 = \begin{cases} \hat{j}_{21} & \text{si } \min\limits_{\substack{i\in\{1,2\} \\ k\in\{2,\dots,4\}}} |\hat{j}_{21} - \hat{j}_{ik}| < \Delta. \\ K & \text{sinon.} \end{cases} \tag{14}$$

Dans les figures 6a et 6b, tous les descripteurs ont trouvé la bonne borne de transition à $\Delta = 35$ images près. Les bonnes bornes ont été labellisées manuellement. Les vidéos sont coupées selon les bornes trouvées par le détecteur de joueurs D_1.

4 Résultats

Pour mesurer l'efficacité de notre méthode, les performances de l'algorithme ont été évaluées pour un Δ fixé à 35 images sur une base de plus de 2900 vidéos prises sur la plateforme Rematch [1], composée de vidéos qui nécessitent des découpes ainsi que de vidéos qui n'en nécessitent pas sur différents sports en salle et en extérieur. La position des sous-séquences à supprimer varie entre le début de la vidéo, la fin, ou les deux. Toutes les vidéos de la base de données ont été labellisées manuellement avec les bonnes bornes de découpes. La valeur choisie pour Δ a été déterminée par apprentissage et les descripteurs utilisés ont été retenus après avoir été comparés à un ensemble de 10 descripteurs déjà adaptés à la discrimination d'images indésirables. Un apprentissage a été réalisé afin de connaître les combinaisons de descripteurs les plus performantes. On étudie ici le traitement de toutes les vidéos, mais plus précisément les résultats de l'algorithme sur les vidéos qui contiennent des sous-séquences indésirables.

Sur le graphique du haut de la figure 7, il s'agit des résultats de l'algorithme sur toutes les vidéos. Les vidéos non-coupées correspondent à celles qui ne nécessitaient pas d'être coupées et qui ont correctement été traitées par l'algorithme. Les vidéos bien coupées sont celles qui avaient besoin d'être coupées et où l'algorithme a renvoyé les bonnes bornes de découpe. Les vidéos mal coupées correspondent à celles qui nécessitaient d'être coupées et où l'algorithme n'a pas renvoyé les bonnes bornes de découpe mais n'a pas supprimé de séquence à conserver, et les vidéos endommagées sont des vidéos où l'algorithme a supprimé des séquences que l'on souhaitait conserver. Le graphique du bas reprend ces informations en analysant le ratio vidéos bien coupées et vidéos mal coupées seulement sur l'ensemble des vidéos qui nécessitaient une découpe.

1. https://www.rematch.tv

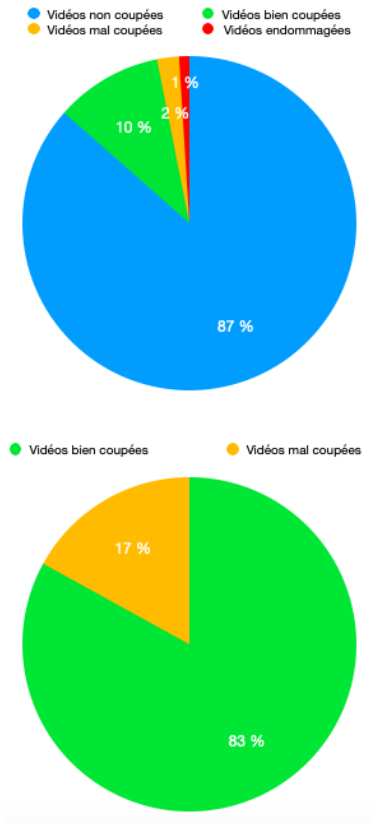


FIGURE 7 – Résultats de l'algorithme sur la base de données de 2919 vidéos : en haut, résultats sur la base de données complète ; en bas, résultats sur les vidéos qui nécessitaient une découpe. 83% des vidéos qui devaient être coupées ont été bien traitées. Moins de 1% des vidéos ont été endommagées (et la majorité des séquences endommagées étaient inférieures à 1 seconde).

La base de données de test regroupe 2919 vidéos, et est composée de 365 vidéos qui doivent être coupées. Le graphique du haut de la figure 7 montre que l'algorithme a traité correctement 2828 vidéos, soit $96,9\%$ (pas de découpe si la vidéo ne contient que des images à conserver, ou calcule les bonnes bornes si la vidéo nécessite d'être coupée). Le graphique du bas de la figure 7 montre que parmi les 365 vidéos qui avaient besoin d'être coupées, l'algorithme en a traité correctement 303, soit 83% de cette sous-base. 29 vidéos sur toute la base ont été endommagées ce qui représente 0.99% de la base de données complète, et les séquences endommagées sont en majorités inférieures à 1 seconde. Toutes les vidéos endommagées étaient des vidéos qui n'avaient pas besoin d'être coupées.

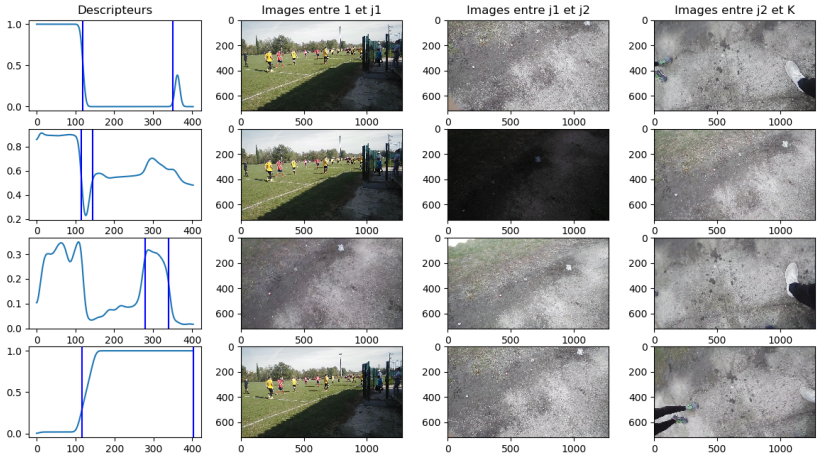
La figure 8 montre trois exemples de résultats des descripteurs sur des vidéos bien coupées par l'algorithme ainsi que des images présentes dans les sous-séquences définies par chaque descripteur afin de mieux comprendre comment ils fonctionnent.

5 Conclusion

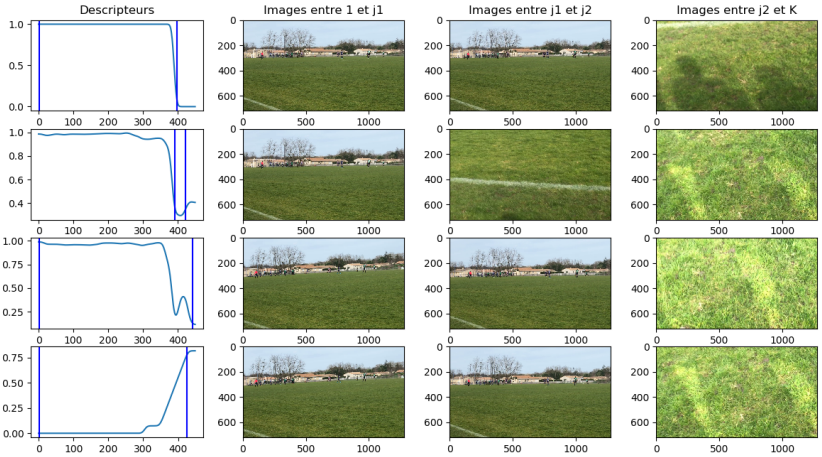
Nous avons présenté une méthode qui permet de trouver les bornes de découpe en fonction du contenu d'une vidéo permettant de supprimer les sous-séquences indésirables. Cette méthode utilise principalement la détection de personnes qui doit être validée pour les vidéos en extérieur par des descripteurs colorimétriques adaptés à notre problème. L'avantage de cette méthode est qu'elle permet d'éviter d'endommager des vidéos tout en conservant un pourcentage de vidéos bien coupées supérieur à 80%. Il existe d'autres possibilités pour améliorer les performances de cet algorithme. A présent nous disposons d'une grande base de données annotées : il est donc possible de définir un modèle d'apprentissage sur cette base afin d'entraîner un réseau de neurones à discriminer les sous-séquences à conserver de celles que l'on veut supprimer. Les résultats des méthodes basées sur les réseaux de neurones pour la classification d'images justifient l'intérêt d'une telle méthode pour cette application. Il est également possible de définir un modèle d'apprentissage sur nos données pour améliorer les performances du détecteur de joueurs, notamment pour augmenter sa robustesse face aux mouvements de la caméra et des joueurs ainsi qu'aux forts changements d'échelle.

6 Remerciements

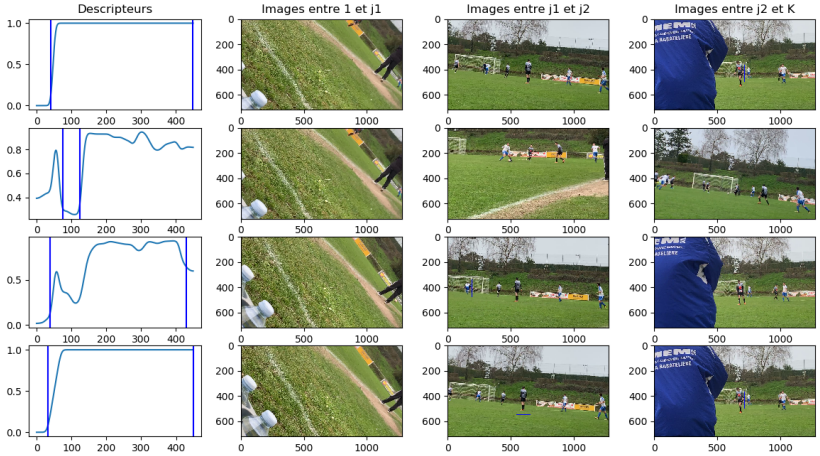
Les auteurs remercient l'entreprise Rematch, qui a financé une grande partie de ces travaux et fourni toutes les données nécessaires à son bon déroulement, plus particulièrement Franck Si-Hassen et Pierre Husson qui ont fait tout leur possible pour que ce projet soit mené à bien. Nous souhaitons remercier le Ministère en charge de l'Enseignement Supérieur, de la Recherche et de l'Innovation ainsi que l'ANRT qui financent également le dispositif CIFRE.



(a)



(b)



(c)

FIGURE 8 – A gauche, les variations de chaque descripteur sur les vidéos (a), (b) et (c), et à droite, des images prélevées au milieu de chaque sous-séquence définie par le descripteur en question (dans l'ordre de haut en bas : D_1, D_2, D_3, D_4).

Références

[1] P. Beaudet. Rotationally invariant image operators. International Journal of Current Pharmaceutical Research, pages 579–586, 1978.

[2] R. Benenson, M. Omran, J. Hosang, and B. Schiele. Ten Years of Pedestrian Detection, What Have We Learned ? Computer Vision - ECCV 2014 Workshops, 2015.

[3] P. Burt and E. Adelson. The Laplacian pyramid as a compact image code. Readings in Computer Vision, Morgan Kaufmann, pages 671-679, 1987.

[4] J. Crowley and R. Stern. Fast computation of the difference of low pass transform. IEEE Transactions on Pattern Analysis and Machine Intelligence, 6(2) : 212-222, 1984.

[5] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), pp. 886-893 vol. 1, 2005.

[6] P. Dollar, C. Wojek, B. Schiele and P. Perona, Pedestrian detection : A benchmark. 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 304-311, 2009.

[7] P. Dollar, C. Wojek, B. Schiele and P. Perona. Pedestrian Detection : An Evaluation of the State of the Art. In IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, no. 4, pp. 743-761, 2012.

[8] M. Grand-Brochier. Descripteurs 2D et 2D+t de points d'intérêt pour des appariements robustes. Autre. Université Blaise Pascal - Clermont-Ferrand II, 2011.

[9] T. Lindeberg. Scale invariant feature transform. Vol. 7, no. 5. p. 10491, 2012.

[10] T. Lindeberg. Feature detection with automatic scale selection. International Journal of Computer Vision 30(2) : 77-116, 1998.

[11] P. Mazzeo, P. Spagnolo, M. Leo and T. D'Orazio. Visual Players Detection and Tracking in Soccer Matches. IEEE Fifth International Conference on Advanced Video and Signal Based Surveillance, pp. 326-333, 2008.

[12] B. Schölkopf and A. Smola. Learning With Kernels : Support Vector Machines, Regularization, Optimization and Beyond. MIT Press, 2002.

[13] J. Shi and C. Tomasi. Good features to track. IEEE Conference on Computer Vision and Pattern Recognition, pages 593–600, 1994.

[14] M. Smith and J. Brady. Susan - a new approach to low level image processing. International Journal of Computer Vision, 23 :45–78, 1997.

[15] L. Sun and G. Liu. Field lines and players detection and recognition in soccer video. IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 1237-1240, 2009.

[16] S. Sergyán. Color Content-based Image Classification. 5th Slovakian-Hungarian Joint Symposium on Applied Machine Intelligence and Informatics January 25-26. Poprad, Slovakia, 2007.

[17] V. Vapnik and A. Lerner. Pattern Recognition using Generalized Portrait Method. Automation and Remote Control, 1963.

[18] P. Viola and M.Jones. Robust Real-time Object Detection. Second international workshop on statistical and computational theories of vision – Modeling, learning, computing, and sampling Vancouver, Canada, July 13, 2001.

[19] X. Wang, V. Ablavsky, H. Shitrit and P. Fua. Take your eyes off the ball : Improving ball-tracking by focusing on team play, Computer Vision and Image Understanding, Volume 119, Pages 102-115, ISSN 1077-3142, 2014.