Réduction de dimension pour l'analyse de données vidéo

Nicolas Verbeke*, Nicole Vincent*

*Laboratoire CRIP5-SIP, Université René Descartes Paris V,
45 rue des Saints-Pères, 75270 Paris Cedex 06, France
{nicolas.verbeke,nicole.vincent}@math-info.univ-paris5.fr,
http://www.sip-crip5.org

Résumé. Les données vidéo ont la particularité d'être très volumineuses alors qu'elles contiennent peu d'information sémantique. Pour les analyser, il faut réduire la quantité d'information dans l'espace de recherche. Les données vidéo sont souvent considérées comme l'ensemble des pixels d'une succession d'images analysées séquentiellement. Dans cet article, nous proposons d'utiliser une analyse en composantes principales (ACP) pour réduire la dimensionnalité des informations sans perdre la nature tridimensionnelle des données initiales. Nous commençons par considérer des sous-séquences, dont le nombre de trames est le nombre de dimensions dans l'espace de représentation. Nous appliquons une ACP pour obtenir un espace de faible dimension où les points similaires sémantiquement sont proches. La sous-séquence est ensuite divisée en blocs tridimensionnels dont on projette l'ellipsoïde d'inertie dans le premier plan factoriel. Nous déduisons enfin le mouvement présent dans les blocs à partir des ellipses ainsi obtenues. Nous présenterons les résultats obtenus pour un problème de vidéosurveillance.

1 Introduction

De nos jours, le stockage de grands volumes de données est devenu possible et abordable. Ainsi, à des problématiques aussi diverses que l'analyse statistique de la fréquentation d'un lieu, la sécurisation de l'accès à des bâtiments, la surveillance de malades épileptiques dans des hôpitaux, ou encore la facturation des véhicules aux péages des autoroutes, les industriels proposent de plus en plus de solutions techniques basées sur l'acquisition numérique de séquences vidéo. Ces données vidéo sont tridimensionnelles (deux dimensions spatiales, et une dimension temporelle). Il s'agit donc d'un volume 2D+T tel que représenté sur la Figure 1(b)[1]. Quelle que soit l'application, la première tâche d'un système d'analyse de séquences vidéo est toujours la détection de mouvement, et si possible, la détection (segmentation) des objets mobiles. La difficulté de cette tâche est très variable selon les conditions d'acquisition, la précision et la rapidité du traitement escomptées. Une liste relativement exhaustive des difficultés liées à l'acquisition et au contenu de la scène peut être trouvée dans Toyama et al. (1999). Dans cet article, nous ne nous intéresserons qu'au cas d'une acquisition par caméra fixe.

[1] La séquence vidéo utilisée pour créer la Figure 1 a été fournie par le laboratoire CVLAB de l'École Polytechnique Fédérale de Lausanne (Suisse).

(a) Une séquence de 200 trames vue comme une succession d'images 2D. (Seules les trames 1, 27, 93, 151 et 200 sont représentées.)

(b) La même séquence vue comme un volume 2D+T. On peut voir qu'un objet mobile est ici représenté par une composante connexe tridimensionnelle.

(c) Vecteur extrait de la séquence représentant l'évolution de l'apparence d'un point (x, y). Il s'agit du point matérialisé par une flèche blanche dans (b).

FIG. 1 – *Différentes manières de représenter une séquence vidéo.*

La plupart des algorithmes de détection de mouvement présents dans la littérature consistent à bâtir un modèle de l'arrière-plan de la scène filmée (c'est-à-dire, ce que l'on verrait s'il n'y avait aucun objet mobile), puis à comparer ce modèle à l'image visible à un instant t. Autrement dit, les données vidéo ne sont pas traitées comme un volume 2D+T (Figure 1(b)), mais comme une succession de couples d'images à deux dimensions (Figure 1(a)). Cette méthode est appelée soustraction de l'arrière-plan (*background subtraction*).

Le modèle le plus simple consiste à considérer à chaque instant t que l'image au temps $t - 1$ représente l'arrière-plan, et que les zones en mouvement sont celles qui ont changé d'apparence entre $t - 1$ et t. En d'autres termes, la détection de mouvement est obtenue par dérivation temporelle de la séquence. La dérivée ainsi obtenue est très rapide à calculer, mais elle est également très instable du fait de sa sensibilité à tout type de bruit. Par ailleurs, seul le passé immédiat est pris en compte donc les mouvements lents ou saccadés sont mal détectés. Ainsi, les auteurs qui utilisent la dérivée temporelle sont obligés d'ajouter des post-traitements afin de corriger le résultat. Dans Tian et Hampapur (2005), la dérivée est lissée par un opérateur

de moyenne mobile, puis les points dont la direction du flot optique a beaucoup varié dans un passé proche sont éliminés du résultat. Le flot optique est défini comme la projection dans le plan de l'image du vecteur mouvement (3D) réel. Un panorama des différentes méthodes pour calculer le flot optique est présenté dans Barron et al. (1994). Les méthodes utilisant le flot optique prennent davantage en considération la dimension temporelle des séquences vidéo, mais se restreignent à un intervalle de temps de taille 2.

Une alternative à l'utilisation de l'image au temps $t-1$ comme modèle de l'arrière-plan est l'utilisation d'une image de référence. Malheureusement, une telle image n'est pas toujours disponible, et même lorsqu'elle l'est, elle devient vite obsolète, particulièrement en environnement extérieur (changement de luminosité, intempéries, etc.) C'est pourquoi les auteurs utilisant cette méthode proposent toujours une fonction de mise à jour de l'image de référence (*background maintenance*), comme dans Yang et al. (2004), où l'image de référence est continuellement mise à jour aux points où la dérivée temporelle est négligeable.

Souvent, le modèle de l'arrière-plan est un modèle statistique permettant d'évaluer la probabilité d'apparition d'un niveau de gris ou d'une couleur en un certain point. Si celle-ci est élevée, on considère que le pixel appartient à l'arrière-plan, sinon c'est qu'il appartient à un objet. Parfois, il s'agit de l'ensemble des paramètres d'une loi dont la forme est supposée connue. Par exemple, dans McKenna et al. (2000) le modèle de l'arrière-plan comporte la moyenne et l'écart-type des valeurs observées sur chaque canal R, G et B, car on suppose que la distribution de chaque composante couleur est gaussienne. Dans d'autres cas, on ne fait pas d'hypothèse a priori sur la forme de la loi à estimer, et on pratique alors une estimation non paramétrique comme dans Elgammal et al. (2000). Le modèle de l'arrière-plan est alors un ensemble d'observations passées qui permettront d'estimer les densités ponctuellement à l'aide d'une fenêtre de Parzen ou d'une fonction noyau.

La soustraction de l'arrière-plan peut également être vue comme un problème de prédiction. Les méthodes les plus utilisées sont le filtrage de Wiener et celui de Kalman. Dans Toyama et al. (1999), le modèle de l'arrière-plan est l'ensemble des dernières images et des coefficients pondérateurs d'un filtre de Wiener associés à chacune de ces images. Les coefficients sont mis à jour de manière à minimiser l'erreur quadratique entre l'image observée au temps t et sa prédiction qui est la somme des images précédentes pondérée par les coefficients du filtre. Dans Koller et al. (1993), c'est un filtre de Kalman qui sert à prédire le vecteur des paramètres, en l'occurrence l'image observée ainsi que ses dérivées spatiales et sa dérivée temporelle.

Ainsi, rares sont les méthodes qui ne cherchent pas à estimer l'arrière-plan de la scène pour détecter les objets en mouvement. On notera néanmoins les travaux de Ma et Zhang (2001) où l'on considère que les zones en mouvement sont celles où l'entropie spatio-temporelle de la séquence est maximale. Contrairement aux méthodes citées précédemment, la dimension temporelle est pleinement prise en considération par un algorithme d'analyse semi-locale dans le volume 2D+T que constitue la séquence vidéo. Dans Guo et al. (2004), cette approche est légèrement modifiée pour calculer l'entropie de la dérivée temporelle plutôt que celle des images d'entrée afin d'éviter de détecter les contours spatiaux comme étant des zones en mouvement. Les résultats obtenus sont assez proches d'une dérivée temporelle lissée par un opérateur de moyenne mobile. Notre étude se situe dans ce cadre : nous cherchons à détecter les objets mobiles en analysant le volume 2D+T de manière globale puis locale. Dans un premier temps nous préciserons l'espace de représentation choisi pour l'étude des séquences, puis dans une seconde partie nous expliquerons les critères utilisés pour la sélection des objets en mouve-

ment. Enfin nous présenterons les résultats et les évaluerons.

2 Espace de représentation des données

Les données vidéo sont initialement représentées par une fonction définie dans un espace à trois dimensions : deux dimensions spatiales (x, y) et une temporelle (t). A chaque point de cet espace est associé un niveau de gris (ou un vecteur de composantes couleur) en un point (x, y) à l'instant t. Les différentes entités sémantiques (arrière-plan, objets mobiles) sont donc des sous-ensembles de points de cet espace. Afin de les identifier, il convient de les agréger en classes de points présentant des caractéristiques communes. Il va sans dire que le nombre de points à considérer est très important, surtout si l'on veut prendre en compte plus de deux trames pour détecter les objets en mouvement. C'est pourquoi l'approche consistant à bâtir un modèle de l'arrière-plan est si usuelle : les seuls points à considérer sont ceux de la trame courante, tandis que le modèle de l'arrière-plan est censé résumer toutes les observations passées. Nous pensons qu'il est préférable de conserver une connaissance moins synthétique du passé car l'information pertinente à en extraire n'est pas toujours la même. Nous envisageons donc de choisir un espace de représentation adapté davantage à la séquence elle-même plus qu'à chacune des trames et qui permette de prendre en compte le mouvement sans modifier l'information initiale. Comme représenté sur la Figure 1(c), à chaque point de l'espace image (x, y) est associé un vecteur contenant les niveaux de gris en ce point le long de l'intervalle de temps considéré. De plus, dans la perspective de l'utilisation des techniques d'analyse des données, la séquence n'est plus considérée comme une fonction mais comme un ensemble d'individus : les pixels que nous observons quand nous regardons la séquence. Dans cette phase, les relations spatiales entre les pixels sont donc ignorées. Pour éviter de devoir faire une analyse fine, ce ne sont pas les objets que l'on suit mais c'est une position fixe que l'on considère sur la surface de l'image. De chaque pixel on va retenir plusieurs valeurs de niveau de gris au cours du temps. On peut retenir une dizaine de valeurs, soit p et chaque pixel devient un individu caractérisé par un ensemble de paramètres. Les individus sont repérés dans un espace de dimension p. Comme notre méthode traite p trames à la fois, nous pouvons nous permettre d'être p fois plus lents que si nous traitions chaque trame individuellement, et donc d'utiliser des techniques plus coûteuses en temps de calcul. Néanmoins, pour rester dans des temps de traitement raisonnables, presque temps réel, nous devons faire une réduction de la masse des informations. Il existe de nombreuses méthodes de réduction de dimension, telles que l'analyse en composantes principales (ACP), l'analyse factorielle des correspondances (AFC), toute la famille des méthodes d'analyse en composantes indépendantes (ACI), ou encore les algorithmes à base de réseaux neuronaux tels que les cartes de Kohonen. Le lecteur intéressé pourra se référer à Lebart et al. (2006) pour obtenir un panorama détaillé. L'ACP étant connue pour être la meilleure technique linéaire de réduction de dimension au sens des moindres carrés, nous avons choisi cette méthode dans le but de ne préserver que les informations qui permettront au mieux de discriminer les points et de construire des classes.

L'ACP a été développée au début du XIXème siècle pour analyser des données issues des sciences humaines. C'est une technique statistique qui vise à simplifier un ensemble de données en l'exprimant dans un nouveau système de coordonnées de manière à ce que les plus grandes variances soient observées sur les premières coordonnées. Cela permet de réduire la dimensionnalité de l'espace de recherche en ne conservant que les premières dimensions de

l'espace de projection obtenu. Une base de cet espace est composée des vecteurs propres de la matrice de covariance des données, ordonnés par valeurs propres associées décroissantes.

On peut penser que les pixels correspondant au fond ont des composantes à peu près toutes égales alors que les pixels correspondant au passage d'un objet mobile comportent un changement. C'est ce changement que l'on veut mettre en évidence. Pour cela il est intéressant de trouver l'axe, c'est-à-dire la bonne base dans l'espace de dimension p où la variance du facteur est la plus grande.

Dans le cas d'une séquence vidéo, la matrice des données $\mathbf{X}$ contient donc l'ensemble des caractéristiques des points à considérer. Par la suite, nous noterons n le nombre de lignes de $\mathbf{X}$, c'est le nombre de pixels de l'image, et p son nombre de colonnes, c'est le nombre de caractéristiques retenues pour chaque pixel. Les deux premières coordonnées peuvent prendre un nombre fini de valeurs (domaine de définition $\mathcal{D}_P$ des pixels). En revanche, le domaine de définition de la troisième coordonnée (le temps) est *a priori* infini. Il convient donc de choisir une plage de valeurs qui devra contenir toute l'information pertinente. Nous proposons d'utiliser le domaine $\mathcal{D}_t = \{t - \Delta t, \ldots, t\}$ où t est le temps courant. Nous choisissons de remplir la matrice $\mathbf{X}$ en considérant qu'une donnée (une ligne) est un point (x, y), et qu'une variable (une colonne) est un ensemble de niveaux de gris observés à chaque instant de $\mathcal{D}_t$. La nouvelle base de l'espace de représentation est alors associée aux vecteurs propres de la matrice de covariance $\mathbf{C}$ des données :

$$\mathbf{C} = \bar{\mathbf{X}}^T \cdot \mathbf{D}_p \cdot \bar{\mathbf{X}}, \tag{1}$$

où $\bar{\mathbf{X}}$ est la matrice des données centrées, et $\mathbf{D}_p = \frac{1}{p}\mathbf{I}_p$ ($\mathbf{I}_p$ étant la matrice identité d'ordre p.) Comme nous n'avons aucune information supplémentaire, on suppose que chaque variable devrait présenter une variance comparable, et nous utilisons une ACP dite « simple » (données centrées) plutôt qu'une ACP « standard » (données centrées-réduites).

La méthode doit être le plus possible insensible aux diverses conditions dans lesquelles se font les acquisitions : nous nous intéressons plus aux variations de niveau de gris qu'au niveau de gris lui-même du pixel. Nous pouvons, dès le départ, supprimer une dimension de l'espace de représentation des données en choisissant de remplir la matrice de données avec les dérivées temporelles en tout point, on est alors dans un espace de dimension $p - 1$. (Nous appellerons $\mathbf{Y}$ la matrice de données représentée dans cet espace.)

Considérons une séquence de 10 trames de 288 lignes par 720 colonnes dont la première est représentée sur la Figure 2(a). La matrice $\mathbf{X}$ a donc 288×720 lignes et 10 colonnes, tandis que la matrice $\mathbf{Y}$ sur laquelle nous allons appliquer l'ACP a 288×720 lignes et 9 colonnes. La Figure 2 montre les neuf projections de $\mathbf{Y}$ sur les axes principaux issus de l'ACP. Plus précisément nous considérons le domaine de l'image et nous construisons une image dont le niveau de gris correspond à la valeur de la composante du vecteur de caractéristiques sur l'un des facteurs.

D'après la Figure 2, les zones en mouvement apparaissent clairement lorsqu'on projette la matrice $\mathbf{Y}$ sur les deux premiers axes principaux. La différence entre une zone statique et une zone en mouvement est accentuée sur ces axes. Cette observation est confirmée par l'histogramme de la variance expliquée par les facteurs (Figure 3). La variance expliquée par un axe est définie par le rapport entre la valeur propre associée à cet axe, et la somme des valeurs propres de la matrice de covariance.

Ainsi, si l'on choisit de ne retenir que les deux premiers axes principaux, 20% de l'information initialement contenue dans notre matrice de données suffisent à préserver 80% de

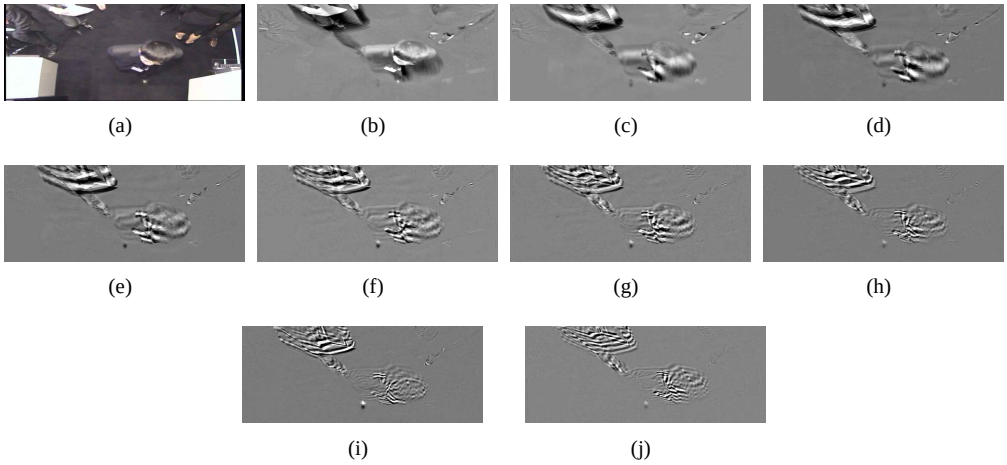

FIG. 2 – *(a) Séquence de travail. (b)—(j) Projections de* $\mathbf{Y}$ *sur chacun des axes mis en évidence par l'ACP. Les sous-figures sont ordonnées selon le rang du facteur correspondant.*

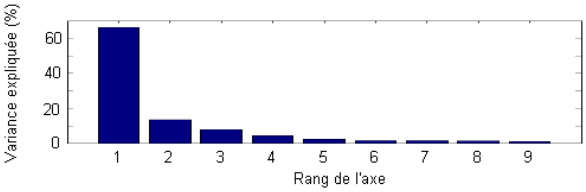

FIG. 3 – *Variance expliquée par les axes principaux.*

la variance observée. Cette première expérimentation confirme donc notre approche qui reste une approche très globale. Dans la section suivante, nous utiliserons donc cet espace de représentation des données. Cela revient à calculer une ACP pour tout ensemble de 10 trames consécutives. De manière à gagner en robustesse nous allons maintenant considérer une approche plus locale qui repose sur cette première étude globale.

3 Détection de zones de mouvement cohérent

La représentation des données telle que dans la Figure 2(b) permet de facilement détecter les mouvements au niveau local (au niveau des pixels). En effet, il suffit de sélectionner les pixels dont la valeur absolue est élevée (les plus sombres et les plus clairs) pour obtenir une segmentation objets mobiles/arrière-plan. La Figure 4(b) représente la segmentation automatique de la projection de $\mathbf{Y}$ sur le premier axe principal.

Nous nous retrouvons alors dans le cas de la plupart des méthodes présentes dans la littérature, une telle image binaire serait étiquetée en composantes connexes pour obtenir une détection par objet mobile. Comme dans la plupart des cas, dans l'exemple de la Figure 4, un

(a) (b) (c)

FIG. 4 – *(a) Projection de* $\mathbf{Y}$ *sur le premier axe principal, (b) segmentation de l'arrière-plan obtenue à partir de (a), (c) segmentation améliorée par des opérations morphologiques.*

prétraitement de l'image serait nécessaire pour supprimer les faux positifs et pour rétablir la connexité des objets (c). Une telle approche fournit une segmentation précise, mais le choix des opérations morphologiques à effectuer est souvent délicat. Une erreur dans le choix d'un élément structurant pourrait effacer un objet intéressant, connecter deux objets différents, valider un faux positif, etc. Une phase d'apprentissage est nécessaire pour adapter la méthode générale au cas particulier de la séquence étudiée. Nous préférons donc éviter d'avoir à effectuer une telle étape, mais nous avons tout de même besoin de définir des zones connexes associées à un unique objet mobile. Pour gagner en cohérence nous allons perdre en précision. A partir de la représentation globale considérée précédemment, dans la population des pixels de l'image, des sous-populations de tailles égales sont isolées et seront comparées. Pour cela, nous commençons par fractionner les données ($\mathbf{Y}$) en plusieurs sous-ensembles. Chaque sous-ensemble correspond à un bloc de $b \times b$ pixels maintenant caractérisés par les neuf valeurs de la matrice de données $\mathbf{Y}$ qui constituent les valeurs des facteurs mis en évidence dans l'étude globale. Sur la séquence initiale ce sont donc des blocs tridimensionnels de taille $b \times b \times 10$ qui sont étudiés au travers de 9 nouvelles caractéristiques. Les blocs que nous avons choisis, de manière à obtenir des résultats plus continus et sans augmenter trop les temps de calcul, se recouvrent par moitié le long des dimensions spatiales.

Les individus des sous-ensembles ainsi obtenus sont représentés dans un espace de dimension $p - 1$. Nous allons étudier les positions relatives de ces ensembles de points. Pour simplifier les calculs, nous représentons chaque ensemble de points par son ellipsoïde d'inertie. De plus nous comparons les projections des ellipsoïdes dans le plan formé par les deux premiers facteurs de la représentation globale.

4 Comparaison des zones détectées

La Figure 5 montre un ensemble d'ellipsoïdes d'inertie projetés sur le premier plan factoriel de l'image globale. Ils correspondent chacun à un bloc spatio-temporel tel que décrits dans la section 3.

Les ellipses observées se différencient par leur position dans le plan, leur surface, et leur orientation. Dans le cadre de cette étude, nous ne nous intéresserons pas à l'orientation des ellipses. Les données, étant centrées, le repère de la Figure 5 a pour origine la moyenne de $\mathbf{Y}$ (ou plus exactement la projection de la moyenne). Par conséquent, une ellipse qui se trouve loin de l'origine représente un bloc dont beaucoup de points sont en mouvement. La surface des ellipses donne une indication sur la variabilité des points du bloc qu'elle représente. On peut ainsi distinguer plusieurs cas :

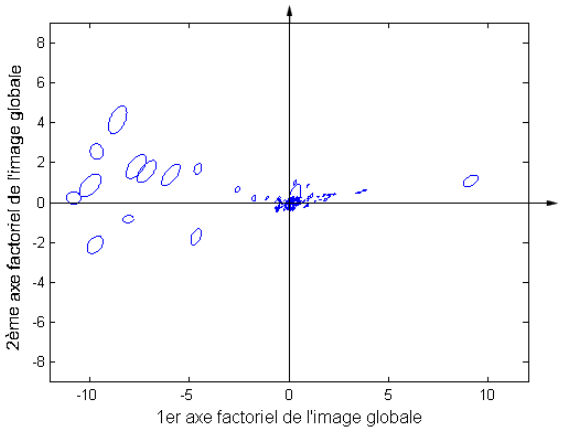

FIG. 5 – *Chaque bloc tridimensionnel est modélisé par l'ellipsoïde d'inertie des points qui le composent, et chaque ellipsoïde est projeté dans le plan formé par les deux premiers facteurs issus de l'étude globale (ACP).*

1. Une petite ellipse proche de l'origine représente un bloc dans lequel aucun mouvement n'est présent.
2. Une grande ellipse proche de l'origine représente un bloc dans lequel les différents points ont des mouvements dissemblables, mais où la moyenne des mouvements est quasi-nulle. Autrement dit, il s'agit de bruit.
3. Une petite ellipse éloignée de l'origine représente un bloc dans lequel le mouvement moyen est important, et dont les points ont quasiment tous le même mouvement. Ce sont les blocs intégralement inclus dans un objet en mouvement.
4. Une grande ellipse éloignée de l'origine représente un bloc dans lequel le mouvement moyen est important, et dont les points présentent des mouvements assez variés. Ce sont les blocs qui peuvent par exemple se trouver à la frontière d'un objet en mouvement.

Pour détecter les objets en mouvement, les blocs les plus intéressants sont donc ceux qui correspondent aux cas 3 et 4, autrement dit, les ellipses éloignées de l'origine. Il faut donc effectuer un seuillage par rapport à la distance à l'origine des centres des ellipses, c'est-à-dire la moyenne (ou la somme) des points appartenant aux blocs correspondants.

5 Expérimentation

La taille des blocs spatio-temporels introduits à la section 3 reste à définir. Des blocs trop larges nuiraient à la précision des contours des objets détectés, tandis que des blocs trop petits impliqueraient des temps de calcul plus élevés, et la connexité des régions pourrait en souffrir. La Figure 6 présente les résultats obtenus pour une même séquence de 10 trames, avec des blocs de taille $b \times b \times 9$, où b vaut successivement 8, 16, 32, 64 et 128. Par ailleurs, nous avons mesuré les temps d'exécution pour obtenir ces résultats, ainsi que leur précision. La précision

est fournie par le nombre de faux positifs et de faux négatifs observés lorsque l'on compare le résultat obtenu avec une segmentation idéale. Ces mesures sont consignées dans le Tableau 1.

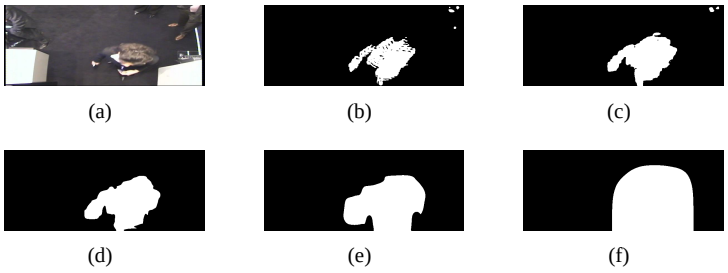

FIG. 6 – *Résultats obtenus à partir d'une même séquence en faisant uniquement varier la taille des blocs spatio-temporels. (a) Séquence de travail. (b)* $b = 8$*. (c)* $b = 16$*. (d)* $b = 32$*. (e)* $b = 64$*. (f)* $b = 128$*.*

	$b = 8$	$b = 16$	$b = 32$	$b = 64$	$b = 128$
Temps d'exécution (%)	100	97,3	94,8	85,4	62,2
Faux positifs (pixels)	516	975	3375	10 372	28 445
Faux négatifs (pixels)	15 245	10 851	6874	2149	506

TAB. 1 – *Performances de l'algorithme avec différentes tailles de blocs.*

Nous constatons que le temps de calcul dépend peu de la taille des blocs (et donc de leur nombre). On peut donc choisir celle-ci uniquement en fonction de la séquence à analyser, sans se soucier du temps de calcul nécessaire. Dans notre cas, les images ont pour dimension 720×288 pixels, et la taille de bloc produisant le moins d'erreurs est 32×32.

Pour évaluer notre algorithme, nous utilisons cinq séquences vidéo qui se différencient par la problématique demandée par l'application et/ou les difficultés intrinsèques de la séquence. La première séquence illustre un problème de comptage de personnes passant par le sas situé en bas de l'image. La difficulté est liée au fait que plusieurs personnes restent longtemps plus ou moins immobiles dans le champ de vision avant de franchir (ou non) le sas. Il faut donc que l'algorithme ne détecte pas les mouvements insignifiants. La seconde vidéo représente également un problème de comptage de personnes, mais là, les personnes ont tendance à se déplacer en groupes connexes. Il faut donc un algorithme suffisamment précis pour pouvoir discerner les différents membres de chaque groupe. La troisième séquence représente une application de détection de passage de véhicules dans le premier plan. La difficulté provient du fait que l'image est très bruitée par le soleil passant à travers les arbres sur la gauche de l'image, et par des véhicules circulant dans l'arrière-plan. Les deux dernières séquences sont des séquences de test classiques utilisées dans de nombreux articles[2]. Elles sont utilisées dans le but de faciliter la comparaison des résultats présentés dans cet article avec d'autres méthodes.

[2]Les séquences présentées dans les deux dernières colonnes de la Figure 7 proviennent du projet CAVIAR/IST 2001 37540 financé par la Commission Européenne, trouvé à l'URL : `http://homepages.inf.ed.ac.uk/rbf/CAVIAR/`.

FIG. 7 – *Résultats obtenus sur 5 séquences avec 5 algorithmes : (1) notre algorithme, (2) dérivée temporelle lissée, (3) modélisation gaussienne, (4) modélisation non paramétrique, (5) entropie spatio-temporelle.*

Sur la Figure 7 sont représentées les segmentations entre objets mobiles et arrière-plan obtenues sur ces cinq séquences vidéo, avec cinq algorithmes différents. La ligne 1 montre les résultats obtenus avec l'algorithme présenté dans cet article ; la ligne 2, la dérivée temporelle lissée de la séquence ; la ligne 3, la soustraction de l'arrière-plan en modélisant celui-ci par une loi gaussienne (McKenna et al., 2000) ; la ligne 4, la soustraction de l'arrière-plan quand celui-ci est modélisé de manière non paramétrique (Elgammal et al., 2000) ; la ligne 5, l'entropie spatio-temporelle de la différence entre images consécutives (Guo et al., 2004). Dans la

littérature, les méthodes concurrentes que nous avons testées sont toujours suivies d'une phase de post-traitement pour faciliter l'extraction des composantes connexes. Pour les lignes 2 à 5, nous avons donc appliqué aux résultats obtenus une fermeture morphologique par un disque de diamètre 5 suivie d'une ouverture morphologique par le même élément structurant.

On constate que les algorithmes utilisant une modélisation statistique de l'arrière-plan (lignes 3 et 4) font apparaître les contours des objets mobiles de manière plus précise. En revanche, ces méthodes sont très sensibles au bruit, donc à moins de choisir très précisément le post-traitement en fonction de la séquence traitée, les résultats obtenus ne constituent pas une bonne segmentation des objets mobiles.

Les contours des objets sont également assez précis avec la méthode de dérivation temporelle (ligne 2). Cela dit, cette méthode a tendance à ne révéler *que* les contours des objets et à en ignorer l'intérieur. Cet inconvénient peut être compensé en augmentant le coefficient de lissage, mais l'on risque alors de créer un effet « fantôme », et de perdre la précision obtenue.

Notre méthode ainsi que celle de l'entropie spatio-temporelle ont en commun le fait de sacrifier la précision des contours au profit d'une plus grande robustesse. Le nombre de composantes connexes est néanmoins plus exact avec la méthode ici présentée (ligne 1).

6 Conclusion

Dans cet article, nous avons présenté une nouvelle méthode de détection de mouvement cohérent dans une séquence vidéo. Contrairement à la plupart des méthodes présentes dans la littérature, nous ne cherchons pas à modéliser l'arrière-plan de la scène pour détecter les objets, mais plutôt à exprimer les données vidéo dans un espace de représentation de dimension réduite, dans lequel la classification entre zones en mouvement et zones statiques est aisée. Pour obtenir cet espace nous appliquons une analyse en composantes principales sur les données d'entrée, et nous ne conservons que les deux premiers facteurs principaux. La séquence est ensuite découpée en blocs spatio-temporels qui sont classifiés par rapport à la position de l'ellipse d'inertie qui les représente dans le plan utilisé. Les résultats obtenus sont satisfaisants dans le sens où le nombre de composantes connexes correspond toujours au nombre attendu. En revanche, les contours des objets sont détectés moins précisément qu'avec les algorithmes de modélisation statistique de l'arrière-plan. Cela dit, dans le contexte d'une utilisation industrielle de la méthode, la précision des contours n'est pas d'une importance capitale ; il est bien plus important de connaître précisément le nombre d'objets présents dans la scène, ainsi que leurs positions et leurs surfaces approximatives. Dans le cas où une grande précision des contours est requise, on pourra utiliser un contour actif (Kass et al., 1988) que l'on initialisera sur le contour fourni par notre méthode. D'autre part, afin de mieux exploiter les informations présentes dans les images d'entrée, nous projetons d'étudier de quelle manière nous pouvons intégrer l'information colorimétrique à notre modèle de données.

Références

Barron, J. L., D. J. Fleet, et S. S. Beauchemin (1994). Performance of optical flow techniques. *Int. J. Computer Vision 12*(1), 43–77.

Elgammal, A. M., D. Harwood, et L. S. Davis (2000). Non-parametric model for background subtraction. In *Proc. European Conf. on Computer Vision (ECCV'00), Volume II*, Dublin, Ireland, pp. 751–767.

Guo, J., E. S. Chng, et D. Rajan (2004). Foreground motion detection by difference-based spatial temporal entropy image. In *Proc. IEEE Region 10 Conf. (TenCon 2004)*, Chiang Mai, Thailand, pp. 379–382.

Kass, M., A. Witkin, et D. Terzopoulos (1988). Snakes : Ative contour models. *International Journal of Computer Vision 1*(4), 321–331.

Koller, D., J. Weber, et J. Malik (1993). Robust multiple car tracking with occlusion reasoning. Technical Report UCB/CSD-93-780, University of California at Berkeley, EECS Department, Berkeley, CA.

Lebart, L., M. Piron, et A. Morineau (2006). *Statistique exploratoire multidimensionnelle : Visualisation et inférence en fouille de données*. Collection Sciences Sup. Dunod.

Ma, Y.-F. et H.-J. Zhang (2001). Detecting motion object by spatio-temporal entropy. In *Proc. IEEE Int. Conf. on Multimedia and Expo (ICME 2001)*, Tokyo, Japan, pp. 265–268.

McKenna, S. J., S. Jabri, Z. Duric, H. Wechsler, et A. Rosenfeld (2000). Tracking goups of people. *Computer Vision and Image Understanding 80*(1), 42–56.

Tian, Y.-L. et A. Hampapur (2005). Robust salient motion detection with complex background for real-time video surveillance. In *IEEE Workshop on Motion and Video Computing*, Volume II, Breckenridge, CO, pp. 30–35.

Toyama, K., J. Krumm, B. Brummit, et B. Meyers (1999). Wallflower : Principles and practice of background maintenance. In *Proc. IEEE Int. Conf. on Computer Vision (ICCV'99)*, Volume 1, Kerkyra, Corfu, Greece, pp. 255–261.

Yang, T., S. Z. Li, Q. Pan, et J. Li (2004). Real-time and accurate segmentation of moving objects in dynamic scene. In *Proc. ACM 2nd Int. Workshop on Video Surveillance & Sensor Networks (VSSN 2004)*, New York, NY, pp. 136–143.

Summary

Video data are known for being very bulky although it contains little semantic information. To analyze it, the amount of information in the search space has to be reduced. Video data is often considered as the set of pixels from an image series that is consecutively analyzed. In this paper, we propose to use principal component analysis (PCA) to reduce the information dimensionality without losing the three-dimensional nature of the input data. We first consider sub-sequences, whose number of frames is the number of dimensions in the feature space. We apply a PCA to obtain a lower-dimensional space, where points with shared features are close to each other. Sub-sequences are then split into three-dimensional blocks whose inertia ellipsoids are projected onto the first factorial plane. Finally we infer the motion occurring within the blocks from the obtained ellipses. We will outline the results achieved for a video surveillance problem.