

# Reconnaissance d'événements dans des vidéos par l'analyse de trajectoires à l'aide de modèles de Markov

Alexandre HERVIEU<sup>1</sup>, Patrick BOUTHEMY<sup>1</sup>, Jean-Pierre LE CADRE<sup>2</sup>

<sup>1</sup>IRISA / INRIA, <sup>2</sup>IRISA / CNRS

Campus Universitaire de Beaulieu, F-35042 Rennes cedex, France

ahervieu@irisa.fr, Patrick.Bouthemy@irisa.fr, lecadre@irisa.fr

**Résumé** – Ce travail présente une méthode originale de classification de trajectoires dans des séquences vidéo pour la reconnaissance d'événements dynamiques. Les Modèles de Markov Cachés (MMC) sont utilisés afin de représenter chaque trajectoire et d'évaluer leurs similarités. Nous avons pu valider notre méthode en la comparant à plusieurs autres méthodes telles que la comparaison d'histogrammes, une distance s'appuyant sur la plus longue sous-séquence commune ainsi qu'avec une méthode utilisant les Séparateurs à Vaste Marge (SVM). Des descripteurs appropriés, invariants à la translation, à la rotation ainsi qu'au facteur d'échelle sont calculés sur les trajectoires, puis exploités dans une représentation par MMC. Nous avons testé notre méthode sur deux ensembles de trajectoires, un premier synthétique composé de classes typiques de trajectoires (telles que les classes de parabole ou de clothoïde), et un second réel contenant des trajectoires obtenues par une méthode de suivi dans une vidéo de grand prix de Formule 1.

**Abstract** – This paper describes an original method for classifying object motion trajectories in video sequences in order to recognize dynamic events. Similarities between trajectories are expressed from Hidden Markov Models representing each trajectory. We have favorably compared our method to several other ones, including histogram comparison, Longest Common Subsequence distance and SVM classification. Trajectory features are computed from the curvature and velocity values at each point of the trajectory, so that they are invariant to translation, rotation and scaling. We have evaluated our method on two sets of data, a first one composed of typical classes of synthetic trajectories (such as parabol or clothoid), and a second one formed with trajectories obtained by tracking cars in a Formula 1 race video.

## 1 Introduction

Le suivi d'objets dans des vidéos est désormais développé au point qu'il est possible d'obtenir des trajectoires fiables d'objets en mouvement dans des situations variées. Les travaux concernant l'analyse de ces trajectoires sont de plus en plus nombreux [1][2], de telles données devant permettre de reconnaître des événements, des actions ou même des interactions entre objets. Le but est alors de fournir une information importante pour l'exploitation automatique de vidéos [3].

Différentes approches ont été étudiées pour la caractérisation de trajectoires. Par exemple, dans [4], le clustering de trajectoires est réalisé en utilisant une distance s'appuyant sur la plus grande sous-séquence commune. D'autres méthodes exploitent les Modèles de Markov Cachés (MMC). F. Porikli a proposé une modélisation par les orientations, invariante aux translations 2D et au facteur d'échelle, et une méthode de classification basée sur les MMC permettant de modéliser la causalité temporelle des trajectoires étudiées [5]. Néanmoins, il s'est appuyé sur des modélisations à l'aide de mélanges de gaussiennes posant des problèmes quant à la représentation des trajectoires (choix difficile du nombre d'états et de gaussiennes dans les mélanges), empêchant cette modélisation de pouvoir traiter les trajectoires de petites tailles.

Notre but est de pouvoir traiter toutes les trajectoires (quelles que soient leurs longueurs et leurs échelles) et d'en extraire des classes correspondant à des mouvements similaires, en terme de chemin suivi et de vitesse de parcours, sans connaissance sur la calibration des caméras et sur la structure de la scène ou sur le mouvement 3D de l'objet considéré. Ainsi, nous proposons une caractérisation des trajectoires invariante à la translation 2D, à la rotation

2D, et au facteur d'échelle, et une méthode originale de reconnaissance à partir de MMC prenant en compte l'évolution temporelle des événements dynamiques observés, et pouvant traiter des trajectoires de toutes tailles.

Le plan de l'article est le suivant. La section 2 présente la caractérisation invariante. En section 3 est décrite la méthode de comparaison de trajectoires à partir des MMC. Dans la section 4, nous introduisons différentes méthodes de classification afin de les comparer à la méthode développée. Enfin, en section 5, nous présentons les ensembles de trajectoires test utilisés, et les résultats expérimentaux sur données synthétiques et réelles.

## 2 Caractérisation de trajectoires

Dans le domaine de l'analyse et de l'interprétation de vidéos, les invariances à un certain nombre de transformations sont souhaitables : invariance à la translation 2D (reconnaissance indépendante de la position dans l'image), à la rotation 2D (indépendance à la direction globale de déplacement) ainsi qu'au facteur d'échelle (indépendance à la distance entre caméra et lieu de l'événement, au moins dans une certaine mesure). Elles seront d'une grande importance dans nombre d'applications vidéo.

### 2.1 Approximation par noyaux

Supposons qu'une trajectoire  $T_i$  soit définie par un ensemble de  $M$  points correspondant aux positions successives de l'objet suivi dans la séquence d'images, on note  $T_i = \{(x_1, y_1), \dots, (x_M, y_M)\}$ . Avant de calculer les descripteurs de cette trajectoire, qui seront des valeurs différentielles, il est préférable d'avoir une représentation continue des courbes formées par les trajectoires. Nous avons donc effectué une approximation par noyaux gaussiens de  $T_i$ , nécessitant le choix de  $h$  (paramètre de lissage de la

courbe), définie par :

$$u_t = \frac{\sum_{j=1}^M e^{-\left(\frac{t-j}{h}\right)^2} x_j}{\sum_{j=1}^M e^{-\left(\frac{t-j}{h}\right)^2}}, \quad v_t = \frac{\sum_{j=1}^M e^{-\left(\frac{t-j}{h}\right)^2} y_j}{\sum_{j=1}^M e^{-\left(\frac{t-j}{h}\right)^2}}.$$

Les expressions explicites  $\dot{u}_t$ ,  $\dot{v}_t$ ,  $\ddot{u}_t$  et  $\ddot{v}_t$  correspondant aux dérivées temporelles à l'ordre un et à l'ordre deux de  $u$  et  $v$  peuvent alors être obtenues.

## 2.2 Invariance de la caractérisation

La plupart des méthodes pour le clustering de trajectoires développées jusqu'à maintenant utilisent les coordonnées spatiales dans l'image comme caractérisation (excepté [4]). Ces coordonnées sont en effet utiles afin d'étudier les ressemblances exactes entre trajectoires, mais notre approche est de considérer l'aspect général des trajectoires (en terme de forme et de vitesse). Prendre en compte les orientations locales successives des trajectoires est plus intéressant et cela permet de comparer leurs formes générales. Considérons les valeurs  $\gamma_t = \arctan(\dot{v}_t/\dot{u}_t)$ , nous avons alors une caractérisation invariante à la translation ainsi qu'au facteur d'échelle. Afin d'avoir une caractérisation également invariante aux rotations, considérons maintenant sa dérivée temporelle  $\dot{\gamma}_t$ . A l'aide de quelques calculs simples de dérivation et de trigonométrie, on prouve que :

$$\dot{\gamma}_t = \frac{\ddot{v}_t \dot{u}_t - \ddot{u}_t \dot{v}_t}{\dot{u}_t^2 + \dot{v}_t^2} = \kappa_t \cdot \|V_t\|$$

avec  $\kappa_t = \frac{\ddot{v}_t \dot{u}_t - \ddot{u}_t \dot{v}_t}{(\dot{u}_t^2 + \dot{v}_t^2)^{\frac{3}{2}}}$  la courbure locale de la trajectoire et  $\|V_t\| = (\dot{u}_t^2 + \dot{v}_t^2)^{\frac{1}{2}}$  la vitesse locale au point  $(u_t, v_t)$ . Le numérateur de  $\dot{\gamma}_t$ ,  $\ddot{v}_t \dot{u}_t - \ddot{u}_t \dot{v}_t = \det \begin{pmatrix} \ddot{v}_t & \dot{v}_t \\ \ddot{u}_t & \dot{u}_t \end{pmatrix}$  est un déterminant, donc invariant à la rotation 2D. Le dénominateur de  $\dot{\gamma}_t$  est  $\dot{u}_t^2 + \dot{v}_t^2 = \|V_t\|^2$ , invariant aux rotations 2D (norme de la vitesse). Par conséquent  $\dot{\gamma}_t$  est invariant aux rotations 2D. Le vecteur de descripteurs utilisé pour représenter une trajectoire donnée est alors le vecteur contenant les valeurs successives de  $\dot{\gamma}(t)$  :  $\phi = [\dot{\gamma}_1, \dot{\gamma}_2, \dots, \dot{\gamma}_{n-1}, \dot{\gamma}_n]$ .

## 3 Distance entre trajectoires et classification

### 3.1 Approche par Modèles de Markov Cachés

Pour modéliser la distribution des  $\dot{\gamma}$ , nous avons choisi de fixer un intervalle contenant un certain pourcentage  $P$  des valeurs de  $\dot{\gamma}$  mesurées (pour l'ensemble des trajectoires à classer) afin d'éliminer les mesures aberrantes et de maintenir les chaînes de Markov à un nombre d'états limité et représentatif. Ensuite, l'intervalle obtenu  $[-S, S]$  est divisé en un nombre  $N$  de bins  $q_i = [\dot{\gamma}_{i,inf}, \dot{\gamma}_{i,sup}]$  (Fig. 1 présente un histogramme de  $\dot{\gamma}$  correspondant à une trajectoire de la classe des sinusoides dans  $[-S, S]$ ).

Chaque trajectoire est modélisée à l'aide d'un MMC dans  $[-S, S]$  avec un nombre d'états donné par le nombre  $N$  de bins. Un MMC à  $N$  états est caractérisé par :

- la matrice de transition  $A = \{a_{ij}\}$  avec

$$a_{ij} = P[q_{t+1} = S_j | q_t = S_i], \quad 1 \leq i, j \leq N,$$

où  $q_t$  est l'état à l'instant  $t$  et  $S_i$  correspond à l'index de l'état  $i$ .

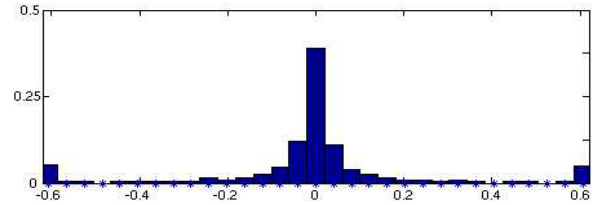


FIG. 1 – Exemple d'histogramme normalisé (histogramme des  $\dot{\gamma}$  d'une trajectoire sinusoïdale avec  $h = 3$ ,  $P = 90\%$ , et  $N = 31$ ).

- la distribution initiale des états  $\pi = \{\pi_i\}$ , où

$$\pi_i = P[q_1 = S_i], \quad 1 \leq i \leq N.$$

- les probabilités d'observation conditionnelle  $B$ , où

$$b_i(X_t) = P[X_t | q_t = S_i],$$

avec  $X_t$  est l'observation au temps  $t$ .

F. Porikli a proposé une méthode de classification basée sur les MMC permettant de modéliser la causalité temporelle des trajectoires étudiées [5]. Néanmoins, il s'est appuyé sur une modélisation par MMC ayant une topologie de type "left-to-right" (*i.e.*, les transitions temporelles entre états du MMC se font uniquement de droite à gauche) et des probabilités d'observation conditionnelle basées sur des mélanges de gaussiennes. Cette modélisation nécessite, pour chaque trajectoire, de choisir le nombre d'états ainsi que le nombre de composantes des mélanges de gaussiennes. De plus, cette méthode est sensible aux tailles des trajectoires, ce qui signifie que des trajectoires de tailles très différentes seront considérées comme différentes. Notre but est d'avoir une méthode pouvant traiter des trajectoires de toutes tailles (notamment les petites trajectoires) et invariante aux tailles des trajectoires.

Ainsi, dans la modélisation choisie, les états des chaînes de Markov sont les bins des histogrammes (le même nombre  $N$  pour toutes les trajectoires), et les observations  $X_t$  sont les  $\dot{\gamma}_t$  calculés. Pour modéliser les probabilités conditionnelles d'observation  $b_i(X_t)$ , nous avons utilisé des gaussiennes centrées en  $\mu_i$  (*i.e.*, le centre du bin  $S_i$  considéré). L'écart type  $\sigma$  de ces gaussiennes est choisi tel que l'intervalle  $[\mu_i - \sigma, \mu_i + \sigma]$  correspond à la largeur des bins. Cette modélisation à l'aide de MMC permet de traiter des trajectoires de toutes tailles, tout en prenant en compte les erreurs dues à l'extraction des mesures.

Afin d'estimer les paramètres du modèle  $A$  et  $\pi$ , nous avons adopté la méthode par moindres carrés définie dans [6] (méthode définie par un processus de comptage). Soit  $H_t^{(i)} = P(\dot{\gamma}_t | q_t = i)$  (correspondant à un poids pour le processus de comptage effectué), les estimations de  $A$  et  $\pi$  sont, pour une trajectoire de longueur  $M$ , données par :

$$a_{ij} = \frac{\sum_{n=1}^{M-1} H_n^{(i)} H_{n+1}^{(j)}}{\sum_{n=1}^{M-1} H_n^{(i)}} \quad \text{et} \quad \pi_i = \frac{\sum_{n=1}^M H_n^{(i)}}{M}.$$

### 3.2 Mesure de similarité et reconnaissance

La distance utilisée pour comparer deux MMC associés à deux trajectoires est celle proposée par Rabiner [7]. Etant donné deux MMC de paramétrage  $\lambda_1$  et  $\lambda_2$  ( $\lambda_i = (A_i, B_i, \pi_i)$ ,  $i = 1, 2$ ), on a

$$D(\lambda_1, \lambda_2) = \frac{1}{T} [\log P(O^{(2)} | \lambda_2) - \log P(O^{(2)} | \lambda_1)]$$

et la version symétrisée de cette distance est

$$D_s(\lambda_1, \lambda_2) = \frac{1}{2}[D(\lambda_1, \lambda_2) + D(\lambda_2, \lambda_1)].$$

où  $O^{(j)} = \dot{\gamma}_1 \dot{\gamma}_2 \dots \dot{\gamma}_T$  est la séquence des états associée à la trajectoire  $T_j$  sachant  $\lambda_j$  (estimée à l'aide d'un algorithme de Viterbi) et  $P(O^{(j)}|\lambda_i)$  exprime la probabilité d'observer  $O^{(j)}$  avec le modèle  $\lambda_i$ . Une classe est alors représentée par un ensemble de MMC correspondant chacun à une des trajectoires utilisées pour apprendre cette classe. La reconnaissance est alors réalisée à l'aide d'une méthode d'agrégation par lien moyen, *i.e.*, en calculant la moyenne des distances entre la trajectoire considérée  $T_i$  et toutes les trajectoires  $T_{l_j}$  de la classe  $G_j$  par :

$$D(T_i, G_j) = \frac{\sum_{T_{l_j} \in G_j} D_s(T_i, T_{l_j})}{\#G_j}.$$

Cela permet d'affecter la trajectoire considérée à la classe associée à la trajectoire d'entraînement la plus proche de la trajectoire traitée.

### 3.3 Classification non supervisée de trajectoires

Nous avons également mis en place une technique de classification non supervisée. Notre méthode permet en effet d'effectuer un clustering à l'aide d'une classification hiérarchique ascendante binaire classique avec méthode d'agrégation par lien moyen entre groupes de trajectoires  $G_i$  et  $G_j$  défini par :

$$D(G_i, G_j) = \frac{\sum_{T_{k_i} \in G_i, T_{l_j} \in G_j} D_s(T_{k_i}, T_{l_j})}{\#G_i \#G_j}.$$

La difficulté supplémentaire est de savoir quand stopper le processus d'agrégation. Deux possibilités sont envisageables, soit on connaît a priori le nombre de classes que l'on désire obtenir, soit l'on fixe un seuil d'arrêt d'agrégation des classes (dans les expérimentations effectuées, le nombre de classes désirées est connu).

## 4 Autres méthodes de classification

### 4.1 Distance de Bhattacharyya entre histogrammes

Afin de démontrer l'importance de la prise en compte de la causalité temporelle, *i.e.*, les transitions entre bins, nous avons implanté une méthode de classification basée sur la distance de Bhattacharyya entre histogrammes. La distance de Bhattacharyya  $D_b$  entre deux histogrammes normalisés  $h_i$  et  $h_j$  (voir Fig. 1) est définie par :

$$D_b(h_i, h_j) = 1 - \sum_{k=1}^N \sqrt{h_i^k h_j^k}$$

où  $h_i^k$  est la valeur dans le bin  $k$  de l'histogramme associé à la trajectoire  $i$ . La reconnaissance est ensuite obtenue de façon analogue à l'approche par MMC, *i.e.*, à l'aide d'une méthode d'agrégation par lien moyen.

### 4.2 Méthode de classification par SVM

Un outil efficace de classification est le SVM (Séparateur à Vaste Marge). En entrée des SVM, nous avons choisi d'utiliser les paramètres des MMC associées aux trajectoires. Les paramètres en entrée des SVM doivent être sous formes de vecteurs. Par conséquent, pour chaque trajectoire, nous avons créé un vecteur contenant les paramètres du MMC correspondant. Par exemple, considérons

le MMC  $\lambda_i$  correspondant à la trajectoire  $T_i$  (pour des facilités de présentation, nous développons un exemple avec  $N = 3$ ) avec

$$A_i = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}, \quad \pi_i = [a_1 \ a_2 \ a_3].$$

Alors  $X_i = [a_{11} \ a_{12} \ a_{13} \ a_{21} \ a_{22} \ a_{23} \ a_{31} \ a_{32} \ a_{33} \ a_1 \ a_2 \ a_3]$  sera le vecteur caractérisant la trajectoire  $T_i$ . Nous utilisons une technique de classification par SVM à l'aide d'un noyau gaussien. Les résultats obtenus sont issus d'un schéma de classification "un contre tous", les paramètres du SVM ayant été fixés à l'aide d'une validation croisée.

## 5 Expérimentations

### 5.1 Trajectoires synthétiques

Pour tester la méthode développée, nous avons tout d'abord généré un ensemble de trajectoires typiques. Plus précisément, nous avons considéré 8 classes (sinusoïdes, paraboles, hyperboles, ellipses, cycloïdes, spirales, droites et clothoïdes) et nous avons simulé 8 trajectoires différentes par classe, correspondant à différentes paramétrisations des courbes, auxquelles nous avons appliqué différentes transformations géométriques (rotation, échelle).

### 5.2 Trajectoires vidéos réelles

Ensuite, nous avons traité des trajectoires réelles extraites d'une vidéo correspondant à un programme TV de course de Formule 1 filmée par plusieurs caméras. Les trajectoires ont été obtenues à l'aide d'une méthode de suivi basée sur le calcul du déplacement associé à des points d'intérêt extraits sur l'objet suivi. Le mouvement de l'arrière plan dû à un mouvement de la caméra (panoramique, zoom ou rotation) est compensé (Fig. 2 et 3). Ainsi les trajectoires obtenues par cette méthode sont visuellement assez similaires aux trajectoires 3D réelles des formules 1 (à une homographie près, le mouvement 3D étant planaire).



FIG. 2 – Images de plans vidéo acquies par deux caméras différentes placées à deux endroits différents du circuit. Les trajectoires extraites sont surlignées dans les images.

### 5.3 Résultats expérimentaux

Pour évaluer les performances de reconnaissance, nous avons adopté la technique de validation croisée "leave-one-out", pour un grand nombre de valeurs de  $h$  (paramètre de lissage),  $P$  (pourcentage de données considéré pour définir l'intervalle  $[-S, S]$ ) et  $N$  (nombre d'états).

En ajoutant du bruit aux trajectoires synthétiques, nous avons évalué l'influence du paramètre  $h$ . Comme attendu, une grande valeur de  $h$  est requise pour les données très bruitées (Tab. 1).

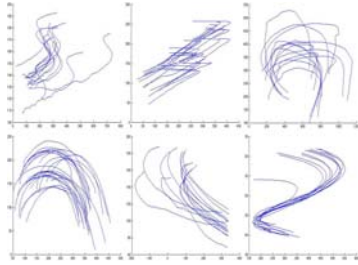


FIG. 3 – Tracé de 6 classes de trajectoires tirées d'une vidéo de course de Formule 1. Une classe est composée de trajectoires extraites de plans filmés par une même caméra. Les différentes classes correspondent à des caméras placées le long du circuit à différents virages stratégiques, la forme du virage induisant celle de la trajectoire des voitures.

$\sigma$ \ $h$	1	2	3	5	8	10	15	20	25
0.1	0.53	0.73	0.83	0.92	0.98	0.98	0.95	0.98	1.00
0.2	0.47	0.73	0.78	0.74	0.94	0.94	0.94	1.00	1.00
0.5	0.33	0.56	0.67	0.78	0.78	0.89	0.88	0.94	0.97
1	0.22	0.52	0.67	0.60	0.69	0.72	0.86	0.90	0.94
2	0.20	0.41	0.44	0.67	0.59	0.67	0.73	0.84	0.91

TAB. 1 – Taux de bonne reconnaissance sur les trajectoires synthétiques, avec la méthode basée sur les MMC, en utilisant la validation croisée "Leave-one-out", pour différentes valeurs de  $h$  et  $\sigma$  ( $\sigma$  est l'écart-type du bruit ajouté) avec  $N = 11$  et  $P = 0.9$ .

Des résultats parfaits de reconnaissance sur les trajectoires synthétiques ont été obtenus avec les méthodes basées sur les MMC et les SVM, et ce pour de nombreuses paramétrisations (de  $N$ ,  $h$  et  $P$ ), et seulement pour quelques paramétrisations avec la technique utilisant la distance de Bhattacharyya. La causalité temporelle est donc bien une donnée importante à intégrer dans la modélisation.

Pour la tâche de reconnaissance appliquée à l'ensemble de trajectoires réelles, deux cas ont été considérés : 4 et 6 classes de trajectoires de formules 1 (Fig. 3). Une classe est composée de trajectoires extraites de plans filmés par une même caméra. Les différentes classes correspondent à des caméras placées le long du circuit à différents virages, la forme de ces virages induisant celle de la trajectoire des formules 1. Des résultats très satisfaisants ont été obtenus avec les méthodes utilisant MMC et SVM (celle basée sur l'utilisation de la distance entre MMC donne les meilleurs résultats, montrant l'importance de l'utilisation de l'algorithme de Viterbi, là où la méthode basée sur les SVM utilise seulement les paramètres des MMC), et des résultats moins précis dans le cas de la méthode basée sur la distance de Bhattacharyya (Tab. 2). Nous avons également comparé notre méthode à celle utilisant la plus longue sous-séquence commune [4] qui a donné de moins bons résultats avec un temps de calcul bien plus élevé. Notre méthode basée sur les MMC pourra être utilisée pour la comparaison de trajectoires ayant des tailles très différentes. De plus, elle est plus flexible que la méthode utilisant les SVM (*e.g.*, possibilité de faire de la détection de trajectoires anormales ainsi que du clustering multi-classes direct, d'ajouter des classes avec un apprentissage complémentaire relatif uniquement aux classes ajoutées).

La technique de classification développée à l'aide de la distance entre MMC a permis d'effectuer, pour plusieurs paramétrisations données (*e.g.*, valeur de  $h$ ,  $N$  et  $P$ ), un clustering non supervisé parfait des quatre classes de trajectoires réelles issues de la vidéo de Formule 1 (Tab. 3), ainsi qu'un clustering satisfaisant avec 6 classes. La comparaison avec les autres méthodes considérées montrent l'efficacité de la méthode développée basée sur la compa-

raison entre MMC.

	Pourcentage de bonne reconnaissance	
	4 classes	6 classes
HMM	100	99.0
SVM	100	96.1
Bhattacharyya	100	93.1
LCSS	97.1	91.2

TAB. 2 – Comparaison des résultats de reconnaissance pour les trajectoires extraites de vidéos réelles, en utilisant la technique de validation croisée. Les 6 classes considérées correspondent à celles présentées en Fig. 3 (les 4 classes considérées étant les 4 classes à gauche de la Fig. 3).

	Pourcentage de bonne classification	
	4 classes	6 classes
HMM	100	92.2
Bhattacharyya	58.4	53.9
LCSS	84.4	72.3

TAB. 3 – Résultats de classification non supervisée obtenus à l'aide d'une classification hiérarchique ascendante binaire avec les méthodes basées sur la comparaison de MMC, sur la distance LCSS et sur la distance de Bhattacharyya pour les trajectoires de formules 1. Les 6 classes considérées correspondent à celles présentées en Fig. 3 (les 4 classes également considérées correspondant aux 4 classes à gauche de la Fig. 3). Le nombre de classes à créer étant connu, il a été utilisé comme critère d'arrêt dans l'agrégation des groupes de trajectoires.

Les meilleurs résultats, pour les tests de reconnaissance et de classification effectués sur les classes de trajectoires synthétiques (sans bruit ajouté) et sur les trajectoires extraites de vidéos réelles, ont été obtenus avec des valeurs de  $P$  de 95%, et de faible valeur de  $h$ . Le choix du nombre d'états  $N$  est lui moins immédiat. Pour certaines configurations de  $h$  et  $P$ , le choix d'un faible nombre d'états (de l'ordre de 5 à 10 états) est le meilleur alors que pour d'autres configurations, un  $N$  plus grand (30 à 50 états) a donné de meilleurs résultats.

## 6 Conclusion

Nous avons proposé une méthode de reconnaissance et de classification d'événements dans une vidéo exploitant les trajectoires dans l'image des objets mobiles de la scène. Nous avons introduit une caractérisation appropriée des trajectoires, invariante à la translation, à la rotation et au facteur d'échelle, et exploité une modélisation causale par MMC. En comparant la méthode développée avec d'autres méthodes, nous avons justifié le choix de cette représentation ainsi que le schéma de classification par MMC. Des premiers résultats très encourageants ont été obtenus, tant sur des données synthétiques que réelles.

## Références

- [1] F. Bashir, A. Khokhar et D. Schonfeld. *Real-time motion trajectory-based indexing and retrieval of video sequences*. IEEE Trans. on Multimedia, vol.9, no.1, pp. 58-65, 2007.
- [2] W. Hu, X. Xiao, D. Xie, T. Tan et S. Maybank. *A system for learning statistical motion patterns*. Pattern Analysis and Machine Intelligence, vol.28, no.9, pp 1450-1464.
- [3] G. Piriou, P. Bouthemy, J.-F. Yao. *Recognition of dynamic video contents with global probabilistic models of visual motion*. IEEE Trans. on Image Processing, vol.15, no.11, pp 3417-3430, 2006.
- [4] D. Buzan, S. Sclaroff et G. Kollios. *Extraction and clustering of motion trajectories in video*. Proc. Int. Conf. Pattern Recognition, Cambridge, UK, août 2004, pp 521-524.
- [5] F. Porikli. *Trajectory distance metric using hidden Markov model based representation*. PETS Workshop, Prague, mai 2004.
- [6] J. Ford et J. Moore. *Adaptive estimation of HMM transition probabilities*. IEEE Trans. on Signal Proc., vol. 46, no. 5, mai 1998.
- [7] L. Rabiner. *A tutorial on hidden Markov models and selected applications in speech recognition*. Proc. IEEE, vol. 77, no. 2, pp. 121-167, 1989.