Titre: Title:	Suivi multi-objets par la détection : application à la vidéo surveillance
Auteur: Author:	Dorra Riahi
Date:	2016
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Riahi, D. (2016). Suivi multi-objets par la détection : application à la vidéo surveillance [Thèse de doctorat, École Polytechnique de Montréal]. PolyPublie. https://publications.polymtl.ca/2326/

URL de PolyPublie: PolyPublie URL:	https://publications.polymtl.ca/2326/
Directeurs de recherche: Advisors:	Guillaume-Alexandre Bilodeau
Programme: Program:	Génie informatique

UNIVERSITÉ DE MONTRÉAL

SUIVI MULTI-OBJETS PAR LA DÉTECTION : APPLICATION À LA VIDÉO SURVEILLANCE

DORRA RIAHI
DÉPARTEMENT DE GÉNIE INFORMATIQUE ET GÉNIE LOGICIEL
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

THÈSE PRÉSENTÉE EN VUE DE L'OBTENTION
DU DIPLÔME DE PHILOSOPHIÆ DOCTOR
(GÉNIE INFORMATIQUE)
SEPTEMBRE 2016

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Cette thèse intitulée :

SUIVI MULTI-OBJETS PAR LA DÉTECTION : APPLICATION À LA VIDÉO SURVEILLANCE

présentée par : RIAHI Dorra
en vue de l'obtention du diplôme de : Philosophiæ Doctor
a été dûment acceptée par le jury d'examen constitué de :

M. LANGLOIS J.M. Pierre, Ph. D., président
M. BILODEAU Guillaume-Alexandre, Ph. D., membre et directeur de recherche
M. KADOURY Samuel, Ph. D., membre
M. MALDAGUE Xavier, Ph. D., membre externe

DÉDICACE

Le rêve du héros, c'est d'être grand partout et petit chez son père.
Victor Hugo...

À celui qui m'a aidé à découvrir le trésor de 'savoir', mon père Mahjoub Riahi

Ce travail est le vôtre ...

À ma très chère mère Leila
À Ghazi, mon amour
À mes chères sœurs Mariem et Houda
À mon frère bien aimé Karim
À la mémoire de ma belle grand-mère Maltifa, tu restes présente dans mon cœur
Au petit ange Yahya

Je vous aime beaucoup ...

REMERCIEMENTS

Cette thèse de doctorat représente un chapitre important de ma vie. J'ai vécu au cours de ces dernières années des satisfactions et des souffrances, des rencontres et des départs, de beau et de décourageant aussi. Ce parcours, jamais linéaire, marque un changement de mes habitudes et de la façon de mes pensées. Avant d'exposer les résultats de mes travaux, je tiens à remercier toutes les personnes qui ont participé de près ou de loin le déroulement de cette thèse de doctorat. Au terme de ces années de doctorat, mes sincères remerciements vont particulièrement à l'inspirateur de ce travail, mon directeur de recherche, le Dr. Guillaume-Alexandre Bilodeau, qui est un modèle professionnel pour sa gestion de temps et bien sûr pour sa maitrise de domaine. Merci monsieur pour votre disponibilité et pour la patience que vous m'avez accordés tout au long de ces années. Je garderais dans mon cœur votre générosité, votre gentillesse, vos précieux conseils et votre compréhension. Vous êtes pour moi un exemple à suivre dans ma vie. Je tiens à exprimer mon profond respect et ma gratitude à M. Pierre Langlois qui m'a fait l'honneur de présider ce jury. Je remercie également M. Samuel Kadoury et M. Xavier Maldague qui ont accepté de server comme member du jury. Merci messieurs d'avoir lu ma thèse et pour vos suggestions de correction. Je tiens également à adresser mes sincères remerciements à l'équipe du laboratoire LITIV dans son ensemble et tout spécialement Wassim, Pierre-Luc, Tanushri et Jean-Philippe pour leur sympathie et la bonne ambiance de travail. Ce fut un grand plaisir de vous connaitre. Je vous souhaite une bonne continuation.

RÉSUMÉ

Dans notre société, les systèmes intelligents ont retenu une attention considérable. En particulier, la vidéosurveillance représente un outil indispensable pour les systèmes de surveillance. Ces systèmes exploitent des données provenant de différents capteurs dans le but d'extraire des informations qui servent à prendre une décision (tel que des événements de menaces). Dans ce contexte, les algorithmes de suivi représentent un vaste sujet important parmi les algorithmes de traitement des systèmes intelligents de vidéosurveillance. En particulier, dans ce travail, on est intéressé par le suivi de plusieurs objets MOT (*Multi Object Tracking*). Bien que de nombreuses approches de suivi ont été proposées, ce sujet reste un défi. Ce travail présente une nouvelle approche de MOT. L'algorithme de MOT que nous avons développé est basé sur l'utilisation de plusieurs descripteurs visuels dans le cadre de l'association des données. Notre approche est capable de gérer certaines problématiques liées au suivi à savoir les occultations à long terme et la similarité entre les modèles d'apparence des objets cibles. L'algorithme MOT repose sur le concept de la fusion de plusieurs descripteurs. Il consiste à sélectionner la position exacte de l'objet à suivre en construisant une représentation robuste du modèle d'apparence des objets cibles. Le modèle d'apparence est extrait en utilisant le descripteur de couleur, le descripteur épars, le descripteur de mouvement et le descripteur de l'information spatiale. Dans le but de sélectionner l'objet candidat optimal (une détection) pour un objet cible, une fonction d'affinité linéaire est estimée. Cette fonction combine les différents scores de similarité qui sont calculés pour chaque descripteur mentionné ci-dessus. Dans notre système de MOT, le processus de suivi est formulé comme un problème d'association des données entre un ensemble des objets candidats (résultats d'un détecteur d'objets) et un ensemble des objets cibles en fonction de la valeur de leur probabilité jointe. Dans la partie expérimentale de ce travail, nous réalisons plusieurs expérimentations dans le but d'évaluer et de confirmer la robustesse de l'approche proposée. Cette dernière a été évaluée en utilisant des séquences vidéo publiques à savoir *TUD* et *PETS2009*. Ces évaluations prouvent la pertinence de notre approche en démontrant que notre approche de MOT surpasse plusieurs algorithmes récents de la littérature.

ABSTRACT

In our society, intelligent systems have attracted considerable attention. In particular, video surveillance is an essential tool for monitoring systems. These systems use data from different sensors in order to extract information used to derive a decision (such as events of threat). In this context, the tracking algorithm is a vast and important subject for video surveillance systems. In particular, in this work, we are interested in tracking multiple objects, MOT (*Multi Object Tracking*). Although numerous tracking approaches have been proposed, this remains a challenging task at the heart of video surveillance applications. This work presents a new MOT approach. The MOT algorithm we developed is based on the use of several visual features as part of the data association. Our approach is able to handle some issues related to tracking namely the long-term occlusions and the close similarity between the appearance model of the target objects. The proposed MOT algorithm is based on the concept of multi-feature fusion. It is based on selecting the exact position of the tracked object by constructing a robust representation of appearance model of the target objects. The appearance model is extracted using the color descriptor, the sparse appearance model, the motion descriptor and the spatial information model. In order to select the optimal candidate object (detection) to a target object, a linear affinity function is estimated. This function combines the different similarity scores which are calculated for each descriptor mentioned above. In our MOT system, the tracking process is formulated as a data association problem between a set of candidate objects (results of a detector of objects) and a set of target objects based on their joint probability value. In the experimental part of this work, we perform several experiments in order to evaluate and to confirm the robustness of the proposed approach. It was evaluated using public video sequences namely *TUD* and *PETS2009*. These evaluations demonstrate the relevance of our approach by demonstrating that our MOT approach outperforms several recent algorithms of the state-of-the-art.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

LISTE DES SIGLES ET ABRÉVIATIONS

MOT	Multi-Object Tracking
ROI	Region Of Interest
CRF	Conditional Random Field
LSH	Locality Sensitive Histogram
TBMOD	Texture-based Moving Object Detection
ISM	Implicit Shape Model
RGB	Red Green Blue
HSV	Hue Saturation Value
HOG	Histogram of Oriented Gradients
SVM	Support Vector Machine
SIFT	Scale Invariant Feature Transform
LSS	Local Self-Similarities
LBP	Local Binary Patterns
SPM	Spatial Pyramid Matching
CNN	Convolutional Neural Network
RMN	Relative Motion Network
JPDAF	Joint Probabilistic Data Association Filter
MHT	Multiple hypothesis Tracking
HOOF	Histograms of Oriented Optical Flow
MOTA	Multi-Object Tracking Acuracy
MOTP	Multi-Object Tracking Precision
TP	True Positive
FP	False Positive
FN	False Negative

CHAPITRE 1 INTRODUCTION

1.1 Contexte technologique

Depuis quelques années, on observe un besoin croissant pour les systèmes de vidéosurveillance. Ces derniers sont de plus en plus présents dans la vie quotidienne tel que dans les banques, les stations de métro, les aéroports, les centres d'achats, les universités, etc. L'analyse du mouvement dans les vidéos est un outil indispensable pour des applications aussi diverses que la vidéosurveillance, l'imagerie médicale, la robotique, l'analyse de séquences sportives, etc. En plus, elle est nécessaire pour explorer des domaines parfois inconnus par l'homme à savoir : les robots de fonds marins ou encore de l'espace. L'application qui nous intéresse dans ce travail est la vidéosurveillance. À cet égard, il est intéressant de noter que le gouvernement canadien a consacré 7.7 milliards de dollars pour la sécurité nationale en 2002 (Canada). Pour utiliser efficacement des caméras de surveillance, un agent entraîné doit regarder continuellement les séquences vidéo et répondre à des activités suspectes. Ceci est difficile en pratique étant donné les limites de l'attention humaine. En fait, d'après une étude développée par le CRIM (Gouaillier (2009)) (Centre de recherche informatique de Montréal), il a été démontré que l'attention de l'opérateur humain baisse significativement après quelques minutes (environ 20 minutes) de concentration. De ce fait, l'automatisation de la vidéosurveillance est devenue un défi réel en vision par ordinateur. Elle consiste à reproduire automatiquement, à l'aide d'un ordinateur, l'analyse de séquence vidéo provenant des caméras de surveillance. Grâce à l'automatisation de la vidéosurveillance, le nombre d'opérateurs humains sera diminué. Pour les systèmes de vidéosurveillance, l'analyse de séquence vidéo se fait sur différents niveaux hiérarchiques, à partir du niveau des pixels, jusqu' à atteindre l'étude des comportements. Ainsi, il est nécessaire de prendre en considération les tâches suivantes :

- **La détection des régions d'intérêts**. Elle est à la base de toute analyse intelligente. Elle consiste à éliminer les zones sans intérêt avant les modules d'analyse du niveau supérieur. En fait, la détection des objets en mouvement consiste à diviser la séquence vidéo en deux parties (une partie statique et une partie dynamique) selon les zones cibles dans l'image.
- **La segmentation de la séquence vidéo**. Elle consiste à diviser l'image en des régions homogènes afin de décrire la séquence vidéo.
- **Le suivi des objets en mouvement**. C'est un processus de haut niveau qui permet d'étiqueter les régions d'intérêts. En d'autre termes, déterminer la position d'un objet en mouvement dans la séquence vidéo à chaque instant. Ce processus de localisation se

fonde sur la représentation de l'objet cible en utilisant un ensemble de caractéristiques discriminantes telles que la couleur, le mouvement, etc.

— **La reconnaissance des activités ou des objets en mouvement**. C'est un processus de haut niveau qui vise à analyser la scène dans le but de déterminer les événements spéciaux tels qu'une personne qui marche, une personne qui court, une personne qui prend un objet, etc.

Le processus du suivi fait l'objet des travaux de cette thèse. La problématique en question représente un domaine de recherche très actif et complexe dans la vision par ordinateur. Parmi les utilisations d'un tel système de suivi, on peut citer : dans la sécurité publique, le système envisagé pourrait être commandé d'une manière autonome pour révéler les menaces cachées et identifier les criminels surtout dans les grands volumes d'observations tels que les sous-sols, les stationnements publics, etc. Dans l'utilisation domestique, un système du suivi domestique peut être installé pour le contrôle automatique d'une maison surtout s'il y a des personnes à mobilité réduite.

D'une manière générale le processus du suivi consiste à localiser les objets cibles sur toute la vidéo. Si le nombre des cibles à suivre est supérieur à 1, on est dans le cas du suivi multi objets. Ce dernier fait l'objet de notre travail. Initialement, les objets candidats doivent être extraits (par exemple, en utilisant la soustraction de l'arrière-plan ou appliquer un détecteur d'objets). Par la suite, pour chaque objet cible, il faut extraire des caractéristiques de haut niveau (fiables et discriminantes) afin de créer le modèle d'apparence. Finalement, la mise en correspondance aussi connue comme l'association des données est faite dans le but de construire des trajectoires continues pour chaque objet cible. Les modèles d'apparence pour chaque objet cible doivent être mis à jour afin de prendre en considération tout changement pouvant affecter l'apparence. L'architecture générale d'un algorithme de suivi MOT est décrit à la figure 1.1.

Le suivi multi objets est un processus complexe qui consiste à estimer les trajectoires de déplacements pour les objets en mouvement dans une séquence vidéo. Il dépend ainsi de :

— La représentation utilisée pour décrire les objets cibles tout en se basant sur l'extraction de caractéristiques discriminatives. Ces dernières dépendent à leur tour de la précision et de la fiabilité des détections.

— L'association des détections avec des trajectoires existantes (*Data Association*). En fait, l'association des données consiste à mettre en correspondance des objets détectés à l'instant courant avec des objets détectés aux instants précédents dans le but de construire des trajectoires continues au cours du temps.

Du coup, les principaux problèmes à gérer dans les systèmes du suivi multi objets sont d'une part liés au nombre des objets cibles et aux ambiguïtés entre eux (par exemple, les

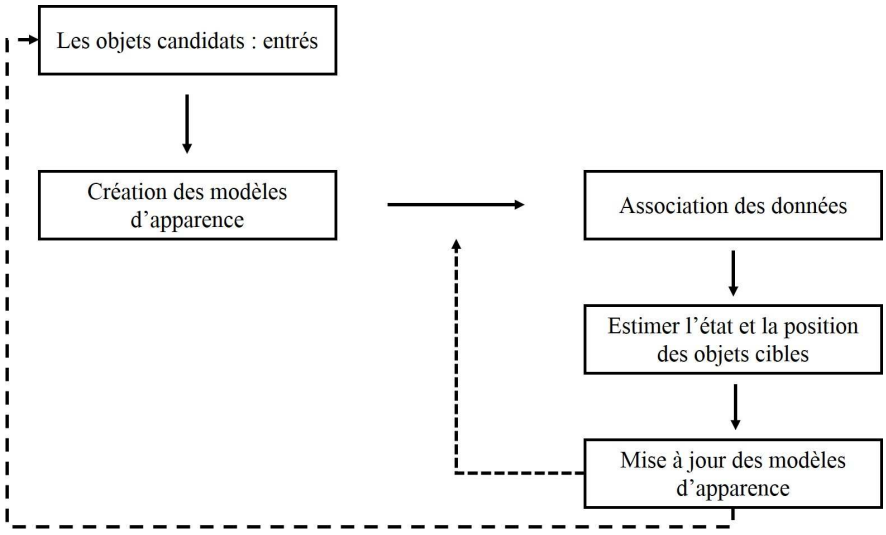


Figure 1.1 Un algorithme du suivi multi objets : approche générale.

occultations) et d'autre part liées à la discrimination des objets cibles. Dans la littérature scientifique, les méthodes du suivi multi objets sont en général classées selon la représentation utilisée pour décrire les objets cibles. Il y a les méthodes du suivi basées sur un modèle d'apparence génératif ou basées sur un modèle d'apparence discriminatif (vu en détail dans le chapitre suivant). Ou bien, une autre taxinomie a été proposée dans Yilmaz et al. (2006) où les méthodes du suivi sont classées selon le type des objets cibles (le format de la région qui englobe l'objet cible) : suivi par point, suivi par contour ou suivi par une fenêtre rectangulaire. Dans ce travail, on propose une nouvelle taxinomie pour les méthodes du suivi en se basant sur les éléments principaux qui définissent un tel algorithme du suivi. Le reste de ce chapitre est organisé de la façon suivante : la problématique est présentée à la section 1.2, les objectifs sont énoncés à la section 1.3. Les contributions et le contenu de cette thèse sont présentés respectivement aux sections 1.4 et 1.5.

1.2 Problématique

Le processus du suivi d'objets peut s'exprimer en fonction de la détection des objets. Le suivi s'attache à détecter les objets d'intérêts à chaque trame de la séquence vidéo puis à mettre en correspondance les objets détectés de façon à obtenir des trajectoires cohérentes pour chaque objet. Ainsi, d'un point de vue fonctionnel, le suivi multi objets fait intervenir trois étapes principales :

1. **La détection des objets candidats**. C'est une étape primordiale pour la plupart des applications de traitement des images. Elle peut être effectuée en utilisant des

techniques de soustraction de l'arrière-plan ou des détecteurs d'objets.

2. **La construction du modèle d'apparence**. Consiste à associer à chaque objet détecté des descripteurs qui permettent de caractériser son modèle d'apparence afin de le comparer avec d'autres objets dans les trames suivantes.
3. **L'association des données**. Consiste à mettre en correspondance un objet détecté à l'instant courant avec des objets détectés aux trames précédentes. Ceci permet de déterminer la position et l'état de l'objet cible à chaque instant.

Afin d'être à la fois plus robuste et performant, un algorithme du suivi multi objets doit traiter les problèmes énoncés ci-dessous :

— **Détections non fiables et imprécises.** Les détections représentent les données d'entrée pour un algorithme MOT. En fait, l'étape de la détection est primordiale pour l'initialisation du processus de suivi. Ces détections sont incertaines dans le sens où il y a des détections manquantes (objets partiellement détectés ou totalement non détectés) ou de fausses alarmes (des objets détectés, mais ne correspondant pas aux objets d'intérêts). Cette imprécision de détections affecte d'une façon directe la performance de suivi MOT. Plusieurs raisons expliquent l'imperfection des détections à savoir le bruit de l'arrière-plan (des objets de l'arrière-plan seront détectés comme des objets d'intérêts) causé par des variations soudaines de la luminosité, vibrations de la caméra, des mouvements quasi statiques des objets cibles. D'autre part, les régions qui englobent les objets détectés peuvent contenir des pixels bruités (qui appartiennent à l'arrière-plan). Il est possible que ces détections soient fausse. Par exemple, dans la figure 1.2, il y a des détections qui ne présentent pas des objets d'intérêts (Les fausses détections sont marqués par des flèches).

— **Changement du modèle d'apparence.** Les objets non rigides peuvent subir des changements au niveau de leurs apparences (voir exemple dans les figures 1.3 et 1.4). En fait, les objets cibles sont généralement représentés par des formes géométriques avec une orientation initiale. Dans le monde réel, ces objets peuvent subir des varia-

Figure 1.2 Exemples de fausses détections.

tions de rotations importantes soit sur le plan de l'image ou des rotations sur le plan réel tridimensionnel (par exemple, une personne vue de face initialement après une rotation, on aura sa vue de dos). Dans le cas du suivi MOT qui utilise seulement des données proviennent d'un seul capteur, il est toujours difficile de qualifier le modèle d'apparence en cas d'une rotation. En outre, les conditions d'éclairage d'une séquence vidéo dépendent de son contenu physique (pièce peu éclairée, scène filmée à l'extérieur, etc.). Si un MOT utilise des descripteurs basés sur l'intensité de couleurs, dans le cas de changements d'illumination, la discrimination des objets cibles peut être perturbée. Un autre facteur aussi important peut influer sur le changement du modèle d'apparence, c'est le changement d'échelle. L'échelle dépend de la distance dans le plan réel entre les objets cibles et la caméra. Plus l'objet est loin de la caméra, plus les détections obtenues sont petites (en nombre de pixels). Du coup, les objets cibles contiennent des informations moins fiables.

— **Apparence similaire des objets cibles.** En général, dans les applications de MOT, les objets à suivre sont de la même nature (soit des personnes, soit des véhicules, etc.). Du coup, la séquence vidéo contient des objets qui sont similaires du point de vue de l'apparence (un exemple est illustré dans la figure 1.5). Ce problème affecte l'efficacité d'un algorithme de suivi plus spécifiquement l'étape d'association des données. En fait, la similarité entre les objets candidats et les objets cibles est presque la même. Ceci augmente l'ambiguité lors de la mise en correspondance.

— **Occultation partielle et totale.** Les problèmes des occultations sont très connus et liés aux applications du suivi d'objet(s). Ils représentent un facteur difficile pour le suivi (voir figure 1.6). L'occultation est définie lorsqu'un objet est partiellement ou totalement masqué par un ou plusieurs autres objets (en mouvement ou fixe). Il existe trois types pour les occultations partielles : auto-occultation lorsque l'objet cible est occulté par une partie de lui-même (par exemple, dans le cas du suivi de visage,

Figure 1.3 Exemples de changement d'échelle.

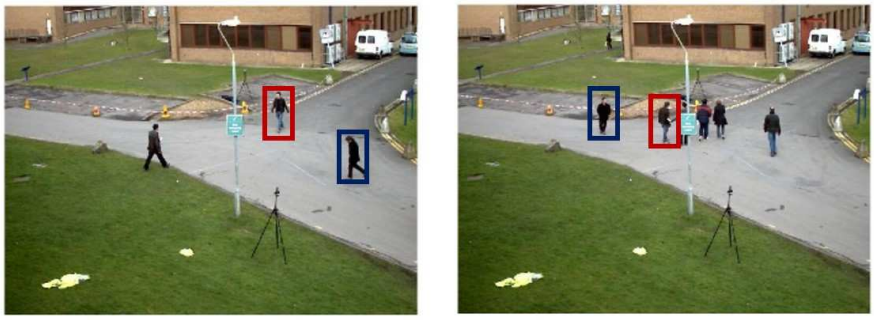


Figure 1.4 Changement du modèle d'apparence dû aux rotations.

Figure 1.5 Objets cibles à apparences similaires.

le visage est occulté par les mains), l'élément occultant est un autre objet d'intérêt présent dans la séquence vidéo ou l'élément occultant est une partie de l'arrière-plan. Pour les séquences vidéo moyennement chargées (qui contiennent un nombre important d'objets à suivre), les situations d'occultations partielles sont fréquentes surtout dans le cas où l'objet occultant est un objet en mouvement. En fait, les objets cibles interagissent entre eux ce qui augmente l'ambiguïté de la discrimination. En plus, une telle situation peut affecter la construction du modèle d'apparence des objets cibles. En effet, le modèle d'apparence contient plus de caractéristiques provenant de l'objet occultant et aussi ne dispose plus des caractéristiques de la (ou les) partie(s) occultée(s). Ainsi le modèle d'apparence risque de changer d'une façon qui ne correspond plus au modèle original. Un autre type d'occultation aussi important, c'est l'occultation totale de l'objet. Un objet cible est totalement invisible soit dû à un objet occultant plus grand (en termes de taille) que l'objet occulté ou si l'objet cible quitte temporairement le champ de vue de la caméra. Lors d'un tel cas, un algorithme MOT doit être capable de prédire la position de l'objet cible lorsqu'il est totalement occulté. La durée de l'occultation a une influence importante sur la stratégie du traitement de ces situations. En fait, plus l'occultation est longue, plus il est probable de perdre le

modèle d'apparence. Un exemple est illustré dans la figure 1.7.

— **Nombre des objets cibles inconnu.** La difficulté d'un algorithme MOT réside dans le fait que le nombre des objets cibles est inconnu et peut varier au cours du temps (voir figure 1.8). Ceci est expliqué par le fait que les objets cibles peuvent apparaître ou disparaître à tout moment. Un algorithme MOT doit être capable de gérer la variabilité du nombre d'objets.

Un bon algorithme de suivi MOT doit être capable de gérer les problèmes décrits ci-dessus.

1.3 Objectifs

L'objectif principal de ce travail est de développer une solution algorithmique fiable et performante qui permette de suivre des objets en mouvements dans une séquence vidéo. Cette solution doit faire face aux difficultés discutées dans la section 1.2 appliquées à des scènes de vidéosurveillance. Afin d'aborder l'objectif général, il faut établir les objectifs spécifiques suivants :

— Le premier objectif de ce travail porte sur l'extraction de caractéristiques discriminatives de plusieurs objets cibles. Dans le cadre de multi descripteurs, chaque objet cible est modélisé en utilisant la combinaison de probabilités multiples. Ces dernières correspondent aux différents scores de similarité basés sur différents descripteurs d'apparence. Ces descripteurs doivent être robustes aux différents problèmes perturbateurs du suivi (occultations, changement d'illumination, etc.). Cette représentation doit assurer la distinction entre les différents objets suivis et être adaptable aux changements de modèle d'apparence.

— Après la description des objets à suivre, chacun doit être associé à une trajectoire en prenant note que le nombre des objets et des trajectoires change dans le temps. Ce second objectif fait référence au problème d'association des données.

— Le troisième objectif concerne la correction des associations des cibles dans le but de gérer efficacement les apparitions et les disparitions des objets en plus des cas

Figure 1.6 Exemple d'occultation partielle et totale.

Figure 1.7 Exemple d'occultation longue. Les trames 2, 33 et 40 (de gauche à droite)

Figure 1.8 Le nombre des objets cibles diminue et augmente au cours du temps.

d'occultations.

— Le dernier objectif est de valider la solution algorithmique en l'appliquant sur plusieurs scénarios et quantifier les résultats en utilisant des métriques standards.

1.4 Contributions

Les algorithmes du suivi sont en général classés en deux catégories : les méthodes en ligne et les méthodes hors ligne. Ces dernières utilisent les informations provenant à la fois des trames déjà passées et des trames futures afin de prédire la position courante des objets à suivre tandis que les approches MOT en ligne utilisent seulement les informations qui existent déjà (dans le passé). Les approches en ligne conviennent aux applications en temps réel. Notre approche est une méthode de suivi en ligne.

D'autre part, ce travail s'inscrit principalement dans le cadre d'une approche du suivi MOT basé sur les détections. L'aspect de détection signifie le fait qu'un détecteur d'objets est indispensable dans le processus du suivi. Un algorithme de suivi MOT se compose de deux étapes principales : la construction du modèle d'apparence et l'association des données pour sélectionner le meilleur objet candidat pour chaque objet cible. La conception d'un algorithme de suivi consiste ainsi à répondre aux questions suivantes : comment décider quel est le meilleur objet candidat ? Quand un objet cible peut-il être considéré en état d'occultation ? L'occultation est elle partielle ou totale ? Cela nécessite une description robuste et efficace

du modèle des objets cibles qui sont a priori inconnus.

Les contributions de cette thèse sont liées à ces deux composantes. Répondant aux objectifs de cette thèse, on propose une approche en ligne de suivi multi objets par détection dans le cadre de plusieurs descripteurs afin de gérer les difficultés mentionnées ci-dessus. Dans ce travail, on a démontré que la meilleure façon d'améliorer une telle approche de MOT est d'utiliser un modèle d'apparence robuste et fiable dans le cadre d'une stratégie d'association des données « intelligente ». Ceci est justifié par le fait que le modèle d'apparence présente un élément crucial pour l'étape d'association des données parce que le modèle d'apparence est souvent très dynamique et les interactions entre des objets similaires peuvent causer des ambiguïtés.

D'abord, le modèle d'apparence de l'objet est construit en utilisant un ensemble de descripteurs indépendants et complémentaires : l'histogramme de couleur, l'histogramme de mouvement, la représentation éparse et l'information spatiale. Un modèle d'apparence robuste est ainsi obtenu qui permet de distinguer d'une manière fiable les objets cibles à chaque instant. En ce qui concerne l'étape de l'association des données, l'algorithme d'optimisation hongrois est appliqué afin d'optimiser les associations entre les objets candidats et les trajectoires des objets cibles à chaque trame. Par ailleurs, afin de gérer les cas d'occultations ainsi que l'apparition et la disparition des objets, les associations seront filtrées de façon à éliminer ceux qui ne sont pas fiables et à en ajouter de nouvelles si nécessaire selon les états des objets cibles (objet en occultation, objet actif, nouvel objet, etc.) et les scores de similarités.

Ainsi, les principales contributions apportées peuvent se décliner en quatre points :

- En réponse au premier objectif, une nouvelle approche du suivi MOT qui combine plusieurs descripteurs récents tels que la représentation éparse et l'histogramme de couleur appelée Locality Sensitive Histogram (LSH) a été développée.
- Une approche d'association des données hiérarchique basée sur l'estimation d'une fonction d'affinité qui combine linéairement les scores de similarité provenant des différents descripteurs.
- Une étape de gestion d'association des données a été proposée afin de répondre au troisième objectif. En plus de gérer les objets entrants et sortants dans la séquence vidéo, une stratégie d'interpolation est utilisée afin d'estimer la position perdue des objets cibles (un objet qui est invisible durant certain temps). Cette interpolation est basée sur les relations spatiales entre les différentes positions des objets (l'historique du mouvement).
- Des résultats expérimentaux démontrent que l'approche proposée testée sur une variété de bases de données fournit des résultats prometteurs en les comparants avec d'autres

méthodes MOT récentes de la littérature.

Dans le cadre de mon projet de doctorat, les contributions décrites ci-dessus ont été publiées dans deux articles de conférences et un autre article de journal soumis :

- Un article intitulé *Multiple feature fusion in the Dempster-Shafer framework for multi-object tracking*. Il a été publié à *Conference on Computer and Robot Vision (CRV 2014)*. Il propose une méthode du suivi multi objets basée sur l'utilisation de plusieurs descripteurs dans le but de construire un modèle d'apparence robuste aux problèmes de MOT.
- Un article intitulé *Multiple object tracking based on sparse generative appearance modeling*. Il a été publié à *IEEE International Conference on Image Processing(ICIP 2015)*. Une version améliorée de méthode de MOT a été présentée dans cet article qui est basé sur l'association des données hiérarchique dans le cadre du suivi en utilisant des descripteurs multiples.
- Un article intitulé *Online multi-object tracking by detection based on generative appearance models*. Il a été accepté pour publication à *Computer Vision and Image Understanding (CVIU)*. Cet article de journal présente un raffinement des deux approches présenté dans les articles déjà publiés.

1.5 Structure de la thèse

Dans le chapitre 2, on présente une revue critique de l'état de l'art des différents aspects de la problématique. Il traite la littérature sur les méthodes de suivi classique. Ensuite, les différentes approches de suivi MOT récent sont abordées. Le chapitre 3 traite l'approche développée pour le suivi des objets en mouvement dans une séquence vidéo. Ceci inclut la présentation en détail des contributions proposées à savoir la construction des modèles d'apparence des objets cibles et l'association des données entre les objets candidats et les objets cibles. Par la suite vient le chapitre 4 qui décrit les expérimentations menées afin de montrer un aperçu qualitatif et quantitatif des différents aspects de la méthodologie proposée. Finalement, le chapitre 5 conclut avec les contributions et propose des perspectives pour ce travail.

CHAPITRE 2 REVUE DE LA LITTÉRATURE

2.1 Concepts de base de suivi multi objets

Le suivi multi objets (MOT) est un processus important pour de nombreuses applications en vision par ordinateur, telles que la robotique, la vidéosurveillance et la reconnaissance d'activités. Par conséquent, une variété d'algorithmes de suivi multi objet a été proposée dans la littérature. Parmi les approches de suivi qui sont très connues et répandues et qui sont utilisées jusqu'à nos jours dans les systèmes de suivi mono objet et dans les systèmes de suivi multi objets, on peut citer les méthodes Mean-Shift, Filtre de Kalman et Filtre de Particules.

— **Mean-Shift**. La procédure Mean-Shift est basée sur la distribution globale des caractéristiques d'intensité (des couleurs ou des niveaux de gris) de l'objet cible. Initialement, un modèle pour l'objet à suivre est sélectionné et l'histogramme de couleur pour la région qui englobe l'objet cible est ainsi construit. Ensuite, la fonction de densité de probabilité est calculé de la façon suivante :

$$q_u = C \sum_{i=1}^{n} k(\|x_i\|)^2 \delta \left[b(x_i) - u \right] \tag{2.1}$$

Où C est une constante de normalisation, k est un noyau qui favorise les centroïdes d'objets, u est l'ensemble d'intervalles de l'histogramme, n est le nombre de pixels de l'objet, b est la valeur de l'intervalle de l'histogramme au pixel x_i et enfin δ est la fonction de Kronecker. Par la suite, la distance de Bhattacharyya est utilisée afin de comparer les histogrammes associés aux objets. Cette opération est répétée jusqu'à ce que la valeur de similarité ne dépasse pas un seuil ou que le nombre limite d'itérations suite atteint (de quatre à six itérations en général). Une phase de réinitialisation du modèle pour les objets cibles permet de gérer les occultations entre eux.

— **Filtre de Kalman**. Cette approche a été proposée par Kalman au début des années 1960. Elle est basée sur un principe récursif. En fait, la position courante de l'objet cible est calculée à l'aide de l'estimation de la dernière position de l'objet. Le filtre de Kalman se fait sur deux étapes : prédiction de l'état courant et mise à jour. La phase de prédiction permet de prédire la position courante de l'objet cible en se basant seulement sur son état précédent. Tandis que l'étape de mise à jour consiste à corriger l'estimation de l'état à l'instant courant afin d'obtenir plus de précision. Mathématiquement, le filtre de Kalman consiste à estimer l'état de l'objet X^t à l'instant

t en fonction de l'estimation de l'état de l'objet (X^{t-1}) à l'instant $t-1$ et d'un terme d'ajustement qui dépend du bruit (le bruit suit la loi de probabilité normale) selon les équations suivantes :

$$X^t = A^t X^{t-1} + W^t \tag{2.2}$$

$$D^t = AD^{t-1}A + Q^t \tag{2.3}$$

Où X est le vecteur des états (les états suivent une distribution gaussienne), A est une matrice de transition qui relie l'état actuel à l'état précédent, D est la matrice de covariance de l'erreur d'estimation, W est le vecteur des entrées déterministes et Q est la matrice de covariance du bruit. Pour l'étape de correction, il faut estimer tout d'abord le gain du filtre de Kalman. Soit l'équation de mesure suivante (ou appelée aussi innovation) qui décrit la dépendance des observations avec l'ensemble des états X :

$$Y^t = B^t X^t + V^t \tag{2.4}$$

Où : Y^t est le vecteur d'observation à l'instant t, B^t est une matrice de mesure à l'instant t, V^t est le bruit de mesure l'instant t. La mise à jour de l'état prédit X^t et de la covariance prédite D^t est calculée selon les deux équations suivantes :

$$\tilde{X}_t = X_t + G_t B_t \tag{2.5}$$

$$\tilde{D}_t = (I - G_t B_t) D_t \tag{2.6}$$

Et G_t est le Gain de Kalman :

$$G^t = D^t B^t (B^t D^t B^t + R^t) \tag{2.7}$$

Où R^t est la matrice de covariance du bruit.

— **Filtre de Particules.** La méthode du Filtre de Particules consiste à estimer la localisation de l'objet cible en fonction de particules pondérées. Initialement, des particules sont générées aléatoirement autour de la dernière position de l'objet cible. Ces particules doivent être comparées (en utilisant un modèle d'apparence comme l'histogramme de couleur) avec le modèle de l'objet cible. Ainsi, la particule la plus similaire peut être choisie comme le nouvel état de l'objet suivi. Soit un objet cible représenté par un ensemble de N particules. Chaque particule fp_i^t, avec $i = 1, ..., N$ est définie par sa position géométrique dans la trame (coordonnés scalaires) (x, y) et un poids (qui reflète l'importance de la particule dans l'estimation de l'objet cible) wf_i^t. Les particules sont sélectionnées après chaque itération selon le principe suivant : sélec-

tionner la plus petite particule (la particule qui a le plus petit numéro) fp_i^t qui a un poids plus grand qu'un seuil k (avec k est nombre aléatoire). Ainsi, les poids des particules seront mis à jour selon les nouvelles caractéristiques (à l'instant courant t). Le temps d'exécution peut être également contrôlé par le nombre de particules générées à chaque itération. Cette propriété permet à la méthode de fonctionner en temps réel. Les résultats de cette méthode vont permettre une meilleure interprétation d'une séquence vidéo (exemple : identification des comportements des personnes, reconnaissance des activités, etc.). L'approche du Filtre de Particules est utilisée souvent pour les méthodes de suivi multi objets appelées *tracking-by-detection*.

Récemment, les méthodes de suivi peuvent être classées selon le modèle d'apparence utilisé : méthodes basées sur un modèle génératif, méthodes basées sur un modèle discriminatif.

— **Méthode générative**. L'objet cible est représenté dans un espace de caractéristiques puis une recherche est effectuée afin de trouver le meilleur score de correspondance avec les objets candidats. En général, cette catégorie de méthode ne nécessite pas un grand ensemble de données pour l'apprentissage.

— **Méthode discriminative**. Le suivi sera traité comme un problème de classification binaire (classe de l'objet cible et classe pour les autres objets). En général, cette catégorie de méthode nécessite un grand ensemble de données pour l'apprentissage.

Une autre catégorisation pour les méthodes de suivi a été proposé par Yilmaz et al. (2006), tout en se basant sur la forme et l'apparence de l'objet : suivi par points, suivi par fenêtres englobantes et suivi par silhouette.

— **Suivi par point**. L'objet cible est représenté par son centroïde ou par un ensemble de points caractéristiques.

— **Suivi par fenêtre**. L'objet cible est représenté par des formes géométriques telles qu'un rectangle, une ellipse ou un cercle (peut être une forme qui entoure ou qui est à l'intérieur de l'objet cible).

— **Suivi par silhouette**. L'objet cible est représenté par son contour ou sa silhouette.

Différemment des stratégies de suivi traditionnel, dans ce travail, on est intéressé par les méthodes de suivi basées sur les détections. Ces méthodes sont récentes et ont eu un grand succès. Un algorithme de suivi multi objets basé sur les détections peut être amélioré soit en améliorant les réponses de détection, le modèle d'apparence de l'objet cible ou la stratégie d'association de données. Des travaux ont été faits récemment sur tous ces aspects. Ainsi, afin d'améliorer les performances d'un algorithme de suivi multi objets, il faut absolument tenir en compte des éléments présentés ci-dessus. En s'inspirant de cela, on propose une nouvelle classification pour les méthodes de suivi multi objets. La figure 2.1 présente la taxinomie proposée en illustrant les sous-catégories de chaque classe. Notre revue de littérature sera

axée sur trois éléments principaux : les objets candidats utilisés (peuvent être le résultat d'un détecteur d'objets ou la combinaison du résultat d'un détecteur et d'un traqueur d'objets), la représentation du modèle d'apparence de l'objet cible (représentation basée sur un seul descripteur ou représentation basée sur plusieurs descripteurs) et la technique utilisée pour l'association des données (des techniques basées sur les algorithmes d'optimisation ou des techniques récemment proposées basés sur la notion de *Tracklet*). Grâce à cette catégorisation, les méthodes de suivi multi objets sont classées selon le type d'amélioration utilisé afin de perfectionner un algorithme de suivi.

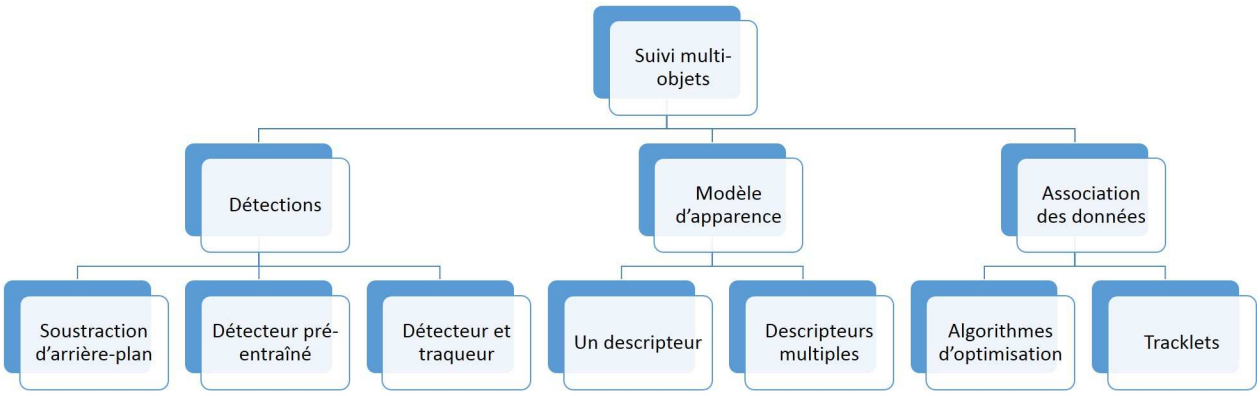


Figure 2.1 Classification des méthodes de suivi multi objets.

2.2 Les détections

Le processus de suivi dépend nécessairement de la qualité des données fournies pour les régions d'intérêts. Ces données peuvent être obtenues soit en utilisant une technique de soustraction d'arrière-plan ou un détecteur d'objets pré entaîné. Les détections peuvent être utilisées uniquement dans l'association des données ou combinées avec d'autres informations : des traqueurs d'objets.

2.2.1 Soustraction d'arrière-plan

Les méthodes traditionnelles de suivi multi objets utilisent des approches basées sur l'extraction des objets en mouvement telle que la soustraction de l'arrière-plan afin d'obtenir les régions d'intérêts [Felzenszwalb et al. (2010), Vijayanarasimhan and Grauman (2014)]. Ces méthodes sont basées sur le fait que s'il y a du mouvement dans une image, la valeur d'intensité d'un pixel sera différente par rapport à celle correspondante dans la trame suivante ou la trame de référence. Généralement, la soustraction de l'arrière-plan se fait de la façon suivante : un modèle initial est créé pour modéliser l'image de l'arrière-plan, la différence entre l'image courante et le modèle de l'arrière-plan est estimée afin d'obtenir les objets en mouvement et finalement le modèle de l'arrière-plan est mis à jour. Les régions ainsi détectées forment probablement les objets d'intérêt. La méthode la plus connue de soustraction de l'arrière-plan est celle de la distribution gaussienne proposée par McKenna et al. (2000). Elle consiste à modéliser l'arrière-plan par une distribution gaussienne. Un pixel est modélisé par sa moyenne μ et sa variance σ^2 qui sont calculées pour chaque canal de l'espace de couleurs (R, G, B). À l'itération t, un pixel est classé comme un pixel en mouvement si $|X_{t,i} - \mu_{t-1,i}| > 3\sigma_{t-1,i}$ où i est un canal parmi R, G ou B. Une mise à jour du modèle de l'arrière-plan est par la suite nécessaire. Une autre approche améliorée a été proposée par Stauffer and Grimson (2000) qui consiste à modéliser l'arrière-plan en appliquant un mélange de distributions gaussiennes. Cette approche est plus robuste aux bruits et aux petits mouvements de l'arrière-plan. Elle vise à modéliser chaque pixel par K distributions gaussiennes. Afin de mettre à jour le modèle d'arrière-plan, chaque distribution est ajustée par un poids. D'autres approches sont basées sur la texture de l'image comme la méthode TBMOD (Texture-based Moving Object Detection) [Heikkila and Pietikainen (2006)]. Elle modélise chaque pixel par un patron binaire calculé par une comparaison entre un pixel et son voisinage. Plus précisément, un histogramme est associé aux patrons locaux sur une région circulaire. Après, la décision est déterminée en comparant les différents histogrammes. Par ailleurs, le défaut principal de cette catégorie est le manque de précision et la sensibilité aux bruits. Pour cela, il est parfois nécessaire de développer des outils de post-traitement afin d'améliorer la qualité des résultats obtenus. Dans ce cadre, nous avons participé à une compétition d'évaluation des différentes méthodes de soustraction d'arrière-plan [Riahi et al. (2012)].

2.2.2 Détecteur d'objets pré entraîné

Pour surmonter ces inconvénients des méthodes de soustraction d'arrière-plan, des approches récentes utilisent un détecteur d'objet pré entraîné (avec un apprentissage selon le type des objets à suivre) afin d'obtenir les régions d'intérêts dans chaque trame. Dans la suite, on

va présenter quelques exemples de ces approches. Dans Wu and Nevatia (2007), les auteurs utilisent un détecteur d'objet partiel. En fait, un détecteur est utilisé pour chaque sous-région de l'objet cible. Les résultats de détection sont par la suite combinés afin d'obtenir des détections globales. Les détecteurs utilisés sont basés sur les caractéristiques de silhouette. Tous les deux : les détections combinées et les détections partielles sont utilisées comme entrées dans le processus de suivi multi objets. Une approche du suivi multi objets a été proposée par Stalder et al. (2010) qui est basée sur un filtrage en cascade de détecteurs d'objets. Ce filtre est composé des contraintes suivantes : la taille des objets, la pondération de l'arrière-plan et la régularité des trajectoires des objets cibles. Un objet détecté est défini par son score de confiance (la probabilité qu'un objet puisse être un objet d'intérêt) $S(I, x, y, s)$ où l'objet situé à la position (x, y) à une échelle s dans la trame I. Les contraintes sont :

— La contrainte géométrique. Cette contrainte est basée sur le fait que tous les objets appartiennent au même plan et que leurs tailles restent immuables durant le suivi. En appliquant ce filtre, le score de confiance devient : $S^{'} = S(I, x, y, s)$.

— La contrainte de l'arrière-plan. Cette contrainte est basée sur le fait que la région de l'arrière-plan est plus souvent présente qu'une région en mouvement. Ceci implique que la variance d'un objet détecté est plus petite que la variance d'un objet d'arrière-plan. Afin d'appliquer la contrainte de l'arrière-plan, le score de confiance $S^{'}$ est modélisé en utilisant une mixture gaussienne.

— Filtre de trajectoires. Ce filtre est basé sur le fait qu'un vrai objet détecté ne peut disparaître tout d'un coup d'une trame à la trame suivante. En plus, le modèle d'apparence d'un objet cible ne peut pas être similaire à d'autres régions dans la même trame.

Cette approche a été appliquée aussi dans le cadre des vidéos de sport. Dans Yao et al. (2010), les auteurs utilisent un détecteur des athlètes (les objets cibles). Le détecteur Hough Forest [Gall and Lempitsky (2013)] a été utilisé afin de détecter les objets cibles. Il y a un classificateur qui permet de détecter une classe d'objets spécifiques telle que les athlètes. Dans cette classification, il y a seulement deux classes : la classe des athlètes (échantillons positifs) et la classe de l'arrière-plan (échantillons négatifs). Une carte de confiance est obtenue qui reflète la valeur de confiance de chaque particule candidate. Cette méthode est hors ligne donc elle ne peut pas être utilisée pour des applications temps réel. Kuo and Nevatia (2011) proposent une approche MOT basée sur l'idée de relier les détections pour former des trajectoires locales appelées *tracklets*. Les détections sont obtenues en utilisant un détecteur de personnes. Par exemple, dans Segal and Reid (2013a), les auteurs utilisent les détections disponibles avec les séquences vidéo. Similairement, le modèle d'observation dans Milan et al. (2014) est déterminé par un détecteur d'objet. Ce dernier est basé sur l'histogramme HOG

ainsi que l'histogramme de flux optique. Les méthodes de suivi basées sur les détections ont montré une amélioration remarquable pour la qualité des résultats de suivi multi objets. Ces méthodes intègrent l'utilisation de détecteurs d'objets cibles (par exemple un détecteur de personnes, un détecteur des voitures, etc.) en combinaison avec un traqueur d'un seul objet ce qui permet de guider le système de suivi. L'inconvénient de ces approches est le fait que la performance de l'algorithme de suivi est proportionnelle à la qualité du détecteur d'objet. De ce fait, des processus supplémentaires sont indispensables afin d'obtenir de hautes performances du suivi multi objets.

2.2.3 Détecteur et traqueur d'objets

Les détecteurs d'objets souffrent de plusieurs problèmes à savoir : les fausses détections, les détections manquantes, la variation de l'échelle, le format de la région qui englobe l'objet cible, etc. Afin de gérer les problèmes de détecteurs d'objets, certaines approches plus récentes utilisent les résultats provenant d'un détecteur d'objets en les combinant avec les résultats provenant d'un traqueur d'objets. Afin d'améliorer la qualité des détections, des approches de suivi par la détection ont été récemment proposées [Breitenstein et al. (2011), Yao et al. (2010), Yang et al. (2009a), Milan et al. (2014)]. Ces méthodes de suivi multi objets sont basées principalement sur l'utilisation d'un traqueur et d'un détecteur d'objets simultanément.

Le traqueur est utilisé pour suivre l'objet cible dans le temps, tandis que le détecteur sert à localiser tous les objets courants qui ont été observés. L'utilisation d'un détecteur et d'un traqueur ensemble permet d'améliorer les lacunes de chacune. En fait, le résultat du traqueur peut être corrigé par le détecteur et une imprécision du détecteur d'objet peut être rectifiée par la prédiction du traqueur d'objet.

Le traqueur et le détecteur seront utilisés d'une façon complémentaire. Dans Breitenstein et al. (2009), afin d'améliorer le détecteur d'objets, les auteurs utilisent un traqueur et un classificateur d'objets. Initialement, pour chaque objet détecté avec un score de classification élevé, un ensemble de particules est généré. Les particules sont générées selon une distribution normale autour du point du centre de la détection. Les particules sont ainsi évaluées en se basant sur un score de similarité. Afin de sélectionner la particule la plus similaire, le principe du filtre de particules est utilisé [Nummiaro et al. (2003)]. En fait, chaque particule est définie par son poids (qui reflète son importance).

Dans Breitenstein et al. (2011), les auteurs utilisent un détecteur d'objet basé sur un classificateur et un traqueur basé sur le filtre de particules. En fait, pour chaque personne détectée, un algorithme de suivi Filtre de Particules est appliqué dans le but de gérer les cas d'occultation et les détections manquantes. Pour chaque objet ayant un score de confiance élevé, un

traqueur Filtre de Particules est créé. Le score de confiance du détecteur est utilisé de deux façons : un score de confiance pour un vote probabiliste pour l'appariement (détecteur ISM) et un score pour localiser les objets cibles (détecteur HOG).

En plus, dans le cas où la détection est classifiée comme étant une détection fiable, cette dernière sera utilisée pour guider le traqueur associé. L'algorithme de suivi est initialisé par seulement les détections qui apparaissent pour une période de temps (des détections imbriquées). Initialement, pour chaque objet détecté avec une probabilité élevée, un traqueur est initialisé. Ils utilisent deux détecteurs d'objets : HOG (Dalal and Triggs (2005a)) et ISM (Leibe et al. (2008)). Pour chaque détecteur utilisé, une carte de confiance est calculée qui reflète la fiabilité de chaque objet détecté. Afin de limiter le nombre des hypothèses (les objets détectés) obtenues pour chaque détecteur, une stratégie de recherche du maximum local est appliquée. En ce qui concerne le traqueur, pour chaque objet cible, un ensemble des particules sont générés. La méthode du filtre de particules consiste à représenter l'objet cible par des particules pondérées. Au début, les particules doivent être comparées (en utilisant un modèle d'apparence comme l'histogramme de couleur) avec une région modèle. Ainsi, la particule la plus similaire peut être choisie comme la nouvelle prédiction de l'objet cible. Les particules sont générés suivant une distribution normale autour du pixel de centre de la particule sélectionnée x_t^i. Chaque particule est définie par un quadruplet : $\{x, y, u, v\}$ où (x, y) sont les coordonnées géométriques et (u, v) sont les composantes de vélocité. Le poids de la particule x_t^i à l'instant t est :

$$w_t^i = w_{t-1}^i . p(o_t | x_t^i) \tag{2.8}$$

Où p est la probabilité de similarité entre l'observation o_t et la particule x_t^i. Cette probabilité est calculée en fonction d'un classificateur et d'un détecteur d'objet. Dans chaque trame, les particules sont générées à partir des équations suivantes :

$$(x, y)_t = (x, y)_{t-1} + (u, v)_{t-1} . \Delta t + \varepsilon_{(x,y)} \tag{2.9}$$

$$(u, v)_t = (u, v)_{t-1} + \varepsilon_{(u,v)} \tag{2.10}$$

Les facteurs de bruits ε suivent une distribution normale.

Une autre approche similaire du suivi multi objets a été proposé dans Okuma et al. (2004), les auteurs utilisent un algorithme adaboost pour la détection afin de guider le traqueur Filtre de Particules. Dans Yang et al. (2009a), les auteurs développent un algorithme de suivi multi objets où le détecteur d'objet est utilisé afin de superviser un traqueur mono-objet. En fait, un algorithme de filtre bayésien est utilisé comme étant un algorithme de suivi. Le filtre bayésien

est appliqué pour chaque image afin de prédire la position courante de l'objet cible. D'un autre côté, un bon détecteur (un détecteur multi vue de tête humaine basé sur le réseau de neurones) de personne est exécuté. La mise en correspondance dépend de score de similarité entre l'objet du traqueur et l'objet du détecteur. Le score de similarité est calculé en combinant plusieurs caractéristiques (la couleur, la forme et la texture) pour construire les modèles d'observation. Dans le cas où le résultat du détecteur est mis en correspondance avec le résultat du traqueur bayésien, cette détection est utilisée pour mettre à jour la trajectoire de l'objet cible. Sinon, une nouvelle trajectoire sera initialisée (création d'un nouveau traqueur). Dans le même esprit, les auteurs dans Yan et al. (2012) exploitent une approche du suivi multi objets basée sur la combinaison d'un détecteur d'objet avec plusieurs traqueurs. Les sorties du détecteur et des traqueurs sont intégrées comme étant deux identités indépendantes dans la phase d'association de données. L'initialisation est faite en utilisant un détecteur de personne ainsi qu'un traqueur pour chaque personne cible. Le détecteur appliqué est un détecteur basé sur la modélisation de l'objet cible en différentes parties et un classificateur SVM. Pour le traqueur, un traqueur basé sur le Filtre de Particules est appliqué. Les particules sont sélectionnées en utilisant l'histogramme de couleur RGB comme modèle d'apparence.

Malgré que les méthodes de suivi basées sur un détecteur d'objets en combinaison avec un traqueur qui permet de guider le système de suivi ont montré un succès important, les résultats combinés obtenus génèrent des candidats redondants ce qui résulte à son tour des fausses assignations entre les objets cibles et les objets candidats. En plus, l'utilisation d'un détecteur d'objet plus un traqueur (soit mono-objets soit multi objets) reste coûteuse en termes de temps de calcul et rend l'algorithme plus complexe.

2.3 Modèle d'apparence

L'étape de modélisation de l'objet cible est une étape cruciale pour le système de suivi multi objets. En fait, la modélisation affecte d'une manière directe la performance d'un système de suivi multi objets. La modélisation est le processus d'extraction des caractéristiques discriminantes qui permet de décrire et de distinguer un objet d'intérêt dans la séquence vidéo. Dans le cas du suivi multi objets, cette représentation permet aussi de distinguer les objets entre eux. En fait, la région qui englobe l'objet cible est convertie en un descripteur qui peut être comparé avec d'autres descripteurs liés à d'autres objets cibles.

2.3.1 Région d'interêt

La région qui englobe l'objet cible peut prendre différents formats : des points, des formes géométriques ou des contours (voir figure 2.2).

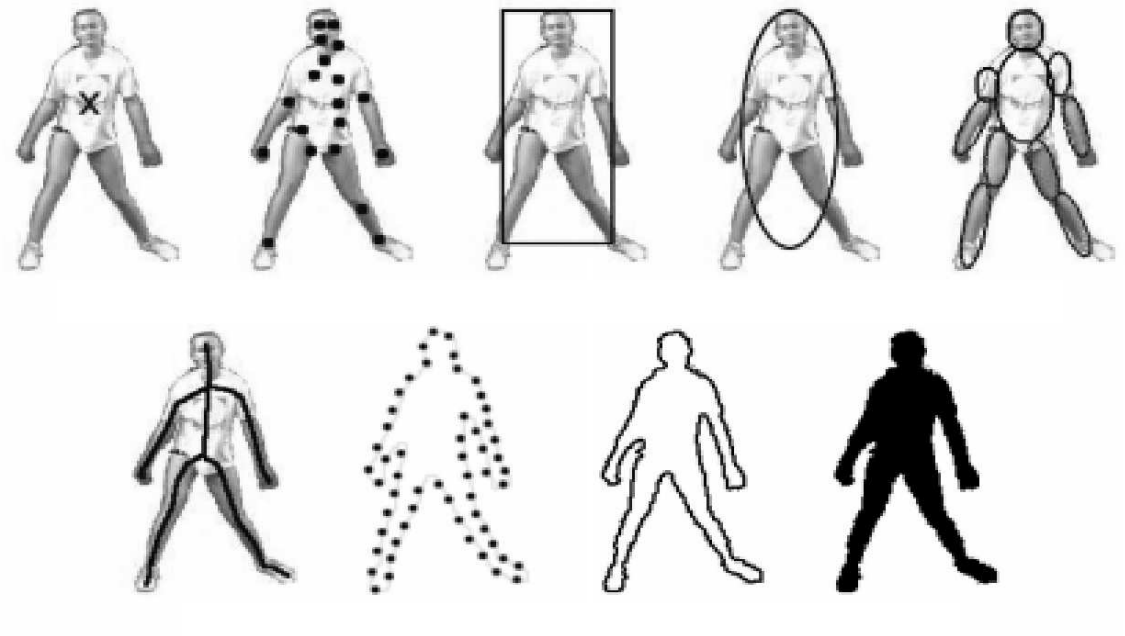


Figure 2.2 Types de détections [Yilmaz et al. (2006)].

— **Suivi par points**. L'objet cible est représenté par son centroïde ou par un ensemble de points caractéristiques. Le processus de suivi est considéré comme étant un processus d'appariement pixel par pixel entre deux trames successives. On peut distinguer deux approches pour cette catégorie : approches déterministes et approches probabilistes. L'approche déterministe consiste à minimiser le coût de correspondance, c.-à-d. minimiser la distance calculée sur certaines caractéristiques de l'objet cible. Cette méthode est basée sur un ensemble de contraintes : les contraintes de rigidité de l'objet, les contraintes de proximité, les contraintes de vélocité maximale, les contraintes de faibles changements et les contraintes de mouvements communs. Autres que l'apparence ou le mouvement d'un objet, les points caractéristiques peuvent être représentés par d'autres paramètres tels que la position, la vélocité et la taille. Le suivi d'objets peut s'effectuer à l'aide des méthodes probabilistes telles que le filtre de Kalman [Yilmaz et al. (2006)].

— **Suivi par fenêtres**. Les objets sont représentés par des formes géométriques telles qu'un rectangle, une ellipse ou un cercle (peut être une forme qui entoure ou qui est à l'intérieur de l'objet à suivre). Les méthodes de suivi par fenêtres sont baséês sur la conservation de l'apparence de l'objet pendant au moins deux trames consécutives. Dans cette approche, nous pouvons distinguer deux catégories : modèle d'apparence basé sur l'appariement de gabarits [Lin et al. (2007)] et modèle d'apparence multivue. La première catégorie est la méthode la plus simple. Elle est basée sur une mesure de similarité afin de déterminer la corrélation entre le modèle et les régions à suivre.
Parmi les mesures de similarité, on peut citer la mesure de corrélation ou la somme des différences au carré. Ces mesures sont basées sur les intensités et les couleurs des objets cibles. Ces dernières sont sensibles aux changements de luminosité. D'autres approches ont été étudiées telles que Mean-Shift [Comaniciu et al. (2003)]. L'approche Mean-Shift consiste à calculer l'histogramme pondéré de couleur pour une région qui englobe l'objet cible. Le principe de cette approche est basé sur la maximalisation de la similarité d'apparence itérativement. L'avantage d'un système de suivi Mean-Shift est d'optimiser la phase de recherche.
L'approche d'appariement de gabarits considère l'apparence de l'objet à partir d'un seul point de vue ce qui la rend appropriée seulement pour les objets dont les positions/orientations ne changent pas considérablement au cours du suivi. De ce fait, d'autres approches qui utilisent un modèle d'apparence multi vue ont été proposées [Black and Jepson (1998)]. Elles consistent à générer un espace d'apparence par apprentissage à partir des différents points de vue comme par exemple apprentissage via des classificateurs SVM (Support Vector Machine) [Papageorgiou and Poggio (2000)].

— **Suivi par silhouette**. La méthode de suivi par fenêtre n'est pas adaptée aux objets de petite taille ou de formes complexes (qui se déforment durant le processus de suivi). D'où la nécessité de définir les objets à suivre par des formes dynamiques (qui changent avec la forme de l'objet en mouvement). Le suivi par silhouette peut représenter les objets à suivre d'une façon dynamique tout dépendamment de leurs formes. Les méthodes de suivi par silhouette sont basés sur l'extraction de la silhouette ou le contour de la cible à chaque instant. Le principe est de modéliser l'objet de la trame courante tout en utilisant le modèle du même objet dans la trame précédente. Il existe deux classes pour le suivi par silhouette : suivi par appariement de contours et suivi direct du contour.
La méthode de suivi basée sur l'appariement de contours de l'objet est similaire au suivi par appariement de gabarits. En fait, elle consiste à calculer la similarité de l'objet dans la trame courante avec le modèle de l'objet dans la trame précédente. Le

modèle de l'objet est représenté sous forme des arêtes qui décrivent le contour de la cible. Le modèle est nécessairement réinitialisé afin d'éviter tous problèmes liés aux changements de points de vue et de luminosité. Dans Huttenlocher et al. (1993), les auteurs utilisent la distance de Hausdorff pour construire la surface de corrélation entre l'objet et son modèle. Cette distance sert à calculer la distance entre deux ensembles des points A et B de la façon suivante :

$$H(A, B) = max(h(A, B), h(B, A)) \tag{2.11}$$

Avec :

$$h(A, B) = sup_{a \in A} inf_{b \in B} \|a - b\| \tag{2.12}$$

D'autres méthodes sont basées sur l'utilisation du contour de l'objet cible. Elles consistent à construire itérativement un contour initial en l'évoluant à sa nouvelle position sur la trame courante. L'évolution du contour peut se faire en minimisant des fonctions d'énergie qui peuvent être définies en fonction de l'information temporelle (tel que le flux optique) ou de l'apparence statique déterminer en fonction de l'objet cible et l'arrière-plan.

— **Suivi par parties**. D'autres approches de MOT utilisent la représentation par parties afin d'améliorer le modèle d'apparence et de gérer les occultations entre les objets. Cette représentation consiste à diviser l'objet cible en différentes régions (qui peuvent être de mêmes tailles ou de tailles différentes). D'où avoir plusieurs modèles d'apparence pour un seul objet cible. Dans Führ and Jung (2014), un objet cible est représenté par un ensemble de parties de tailles égales et qui couvrent toute la région de l'objet cible. Pour obtenir une représentation plus précise de l'objet cible, une stratégie de soustraction de l'arrière-plan est appliquée pour chaque objet cible. Ceci sert à éliminer certains pixels de bruit (par exemple les pixels de l'arrière-plan) qui appartiennent à la détection. Les parties sont définies par rapport à un axe où ses points sont : le point le plus bas et le point le plus haut de l'objet cible. Les parties sont par la suite mises en correspondance séparément en utilisant la distance Bhattacharyya entre leurs histogrammes. Dans Dihl et al. (2011), les auteurs proposent une division en grille rectangulaire de la région de l'objet cible. Ainsi, les parties qui sont mises en correspondance seront considérés comme des parties correctes qui peuvent aider à construire le nouveau gabarit de l'objet cible.

Le choix d'un modèle d'apparence est fortement lié au type des objets cibles, la relation entre les objets cibles, la variabilité des objets cibles dans le temps. Cette modélisation est fidèle au modèle original de l'objet associé. Plus la modélisation de l'objet est robuste, plus

elle peut s'adapter afin de prendre en considération les changements par rapport au modèle d'apparence original (luminosité, échelle, rotation, etc.). Le modèle d'apparence d'un objet cible peut être décrit en utilisant une caractéristique discriminante telle que : le format de l'objet (ou contour), la couleur de l'objet (Tang et al. (2014)), les propriétés de mouvement (Yang and Nevatia (2014a)), les propriétés géométriques (Yao et al. (2010)). Par ailleurs, un objet peut être modélisé par plusieurs caractéristiques à la fois.

2.3.2 Un seul descripteur

Dans cette section, on va explorer une vue d'ensemble pour certains descripteurs les plus utilisés dans la littérature. Comme déjà vu, un descripteur peut décrire une région sélectionnée suivant une caractéristique bien déterminée à savoir les caractéristiques de couleur, les caractéristiques de mouvement, les caractéristiques de forme ou les caractéristiques de texture selon la structure de l'objet à modéliser. Parmi les descripteurs qui sont les plus utilisés dans la littérature, on peut citer :

— **SIFT(Scale Invariant Feature Transform) (Lowe (2004)) :** La modélisation d'un objet basé sur les points SIFT se fait sur trois étapes. Premièrement, il faut détecter les points caractéristiques en les représentant par rapport à leurs voisins. Un point est potentiellement intéressant s'il représente un extremum par rapport à ses 26 voisins (8 pixels voisins sur la même image et 9 pixels sur chacune des deux images voisines) dans des images de différences de Gaussiennes. Deuxièmement, il faut calculer l'orientation des points caractéristiques (en d'autres termes, calculer le gradient de chaque point). De ce fait, les points extrêmes de faible contraste et les points situés sur des arêtes de contour de faible courbure sont éliminés. Enfin, le descripteur SIFT se présente sous la forme d'un histogramme des orientations des gradients contenues dans son voisinage. Un histogramme de 128 intervalles (chaque histogramme comprend 4×4 composants qui représentent les 8 orientations principales entre 0 et 360 degrés) accumule les orientations des gradients du voisinage du pixel. Les points SIFT sont extraits des images et ils sont par la suite stockés dans une base de données. Afin d'identifier un objet dans une scène, il faut comparer ses points SIFT à ceux de bases de données. Ce descripteur est basé sur les gradients qui dépendent des niveaux de gris.

— **LSS(Local Self-Similarities) (Shechtman and Irani (2007)) :** Le descripteur LSS permet d'identifier une requête (sous forme d'une image réelle) dans une image ou une séquence vidéo. Différemment aux autres descripteurs, LSS capture la forme locale d'une région indépendamment de l'intensité, des couleurs, du gradient, etc. L'unité de mesure est une petite région d'image (partie) ce qui le rend plus significatif. Le principe

de fonctionnement de ce descripteur est le suivant : calculer la surface de corrélation pour chaque partie. Ensuite, cette surface est transformée en une représentation log polaire constitué par des intervalles. Enfin, les valeurs maximums de corrélation dans chaque intervalle forment le vecteur de descripteur LSS. Ce descripteur est robuste à plusieurs changements pouvant affecter la scène.

— **HOG (Histogramme de gradients orientés) (Dalal and Triggs (2005b)) :** Le descripteur HOG est un descripteur de forme de l'objet cible basé sur la distribution des gradients d'intensité. L'histogramme HOG consiste à calculer les variations d'intensité pour chaque classe de direction (les gradients ayant des directions différentes). Le principe est aussi simple : les gradients horizontaux et verticaux sont obtenus en appliquant un masque (par exemple, masque de Sobel) pour chaque pixel de la région qui englobe l'objet cible. Un histogramme de magnitude et d'orientation des gradients est formé. Les histogrammes sont par la suite normalisés et concaténés afin d'obtenir l'histogramme global du gradient.

— **Histogramme de couleur :** L'histogramme de couleur est un histogramme usuel souvent utilisé pour les applications de vision par ordinateur. Il est basé sur l'encodage de l'information visuelle de couleur afin de modéliser l'objet cible sans prendre en compte la localisation spatiale des pixels. Il consiste à estimer la distribution des valeurs d'intensité pour l'objet cible.

  Une autre version améliorée de l'histogramme de couleur : le spatiogramme. Le spatiogramme a été proposé par Birchfield and Rangarajan (2005). En plus, les informations de couleurs, il consiste à définir les informations spatiales sur les pixels de l'image. En plus de l'occurrence de l'intensité des pixels, la matrice de covariance, et le vecteur moyen seront définis. L'utilisation de spatiogramme porte plus d'avantages par rapport à l'histogramme classique (Conaire et al. (2008)). Récemment, plusieurs variétés de l'histogramme de couleur ont été développées. On peut citer l'histogramme LSH (sera détaillé dans le chapitre suivant).

— **Flux optique :** Le flux optique permet de décrire les mouvements des objets mobiles dans une séquence vidéo sans modélisation de l'arrière-plan. Il vise à caractériser chaque pixel de l'image par un vecteur de déplacement. Il est souvent utilisé pour la détection et le suivi des objets en mouvement. Par définition, le flux optique (Horn and Schunck (1981a)) est formulé comme un champ de vecteurs à deux dimensions qui caractérise les variations de mouvement d'un pixel d'une image à un instant t. Le flux optique a été proposé par Horn et Schunck et il repose sur l'hypothèse de

conservation de la luminance suivante :

$$p(x + dx, y + dy, t + dt) = p(x, y, t) \tag{2.13}$$

L'approche la plus connue pour l'estimation de flux optique est basée sur le calcul différentiel. L'estimation des vecteurs de champ du flux optique se fait par la résolution de l'équation différentielle suivante :

$$\partial E/(\partial x)dx/dt + \partial E/(\partial y)dy/dt + \partial E/(\partial t) = 0 \tag{2.14}$$

Néanmoins, il est difficile de résoudre cette équation. En effet, il faut trouver deux inconnues avec une seule équation. Il est nécessaire donc d'ajouter une autre équation ou en d'autres termes, une autre contrainte par exemple le fait que le mouvement autour du voisinage d'un pixel est quasi constant (Horn and Schunck (1981a)). Cette approche est moins coûteuse que d'autres (comme les méthodes fréquentielles qui sont basées sur la transformée de Fourier). Le flux optique n'est qu'une représentation des pixels en mouvement dans l'image. Par ailleurs, une étape supplémentaire doit être faite. Il faut donc comparer le flux optique entre deux trames successives afin de déterminer les similarités entre les objets en mouvement. En fait, il faut calculer l'histogramme du champ de mouvement (Histogram of Oriented Optical Flow) qui représente l'histogramme d'orientation pondéré par l'amplitude de champ de mouvement pour l'image. Vu l'importance des caractéristiques de mouvement, l'amélioration du flux optique a été à l'origine de plusieurs travaux qui sont dédiés à la détection et au suivi des objets en mouvement dans une séquence vidéo. Parmi ces travaux, nous pouvons mentionner :

— Calculer le flux optique seulement pour les points d'intérêts du type Harris en utilisant l'algorithme Lucas-Kanade (Salmane et al. (2011)). Le flux optique est ainsi propagé afin de l'estimer pour tous les autres pixels.
— Calculer le flux optique sur une image log-polaire (Zhang (2010)). En fait, le flux optique est estimé seulement sur les pixels en mouvements qui sont extraits en coordonnées log polaire.

L'inconvénient majeur des méthodes basées sur le flux optique est qu'elles sont coûteuses en temps de calcul surtout si le flux optique est estimé pour tous les pixels de l'image. De plus, l'estimation du vecteur de déplacement du flux optique est basée sur l'hypothèse que la différence entre deux images successives est expliquée comme étant une conséquence d'un mouvement, alors que ces changements peuvent être un changement de luminosité, une variation de l'arrière-plan, un bruit dans l'image, etc.

— **Représentation éparse (Sparse Representation) :** Récemment, les méthodes basées sur la modélisation via un modèle épars ont été appliquées avec grand succès au suivi mono-objet. La représentation éparse est basée sur une mesure de similarité entre un objet candidat et un objet cible. Son principe est le suivant : il consiste en la projection linéaire du modèle de l'objet cible sur un espace des gabarits (*Template*) appelé aussi dictionnaire. Ça revient à trouver la combinaison linéaire idéale de gabarits qui permet de représenter l'objet cible. Cette technique cherche à mettre en évidence les gabarits de l'image qui ressemblent le plus à l'objet candidat. En fait, chaque gabarit est défini par des caractéristiques locales ainsi qu'un poids qui dépend de sa corrélation avec le modèle de l'objet cible (sera vu en détail dans le chapitre suivant).

2.3.3 Descripteurs multiples

Parmi les différents descripteurs qu'on a déjà présentés, on peut conclure que chaque descripteur permet de décrire un objet selon un caractère particulier. D'un autre côté, les objets cibles peuvent posséder des caractéristiques similaires telles que les habits vestimentaires. Dans ce cas par exemple, l'histogramme de couleurs ne peut pas être un modèle discriminant. Donc le choix du descripteur dépend de plusieurs facteurs à savoir la nature de l'objet cible, la relation entre les objets à suivre, le facteur de luminosité et d'échelle, etc.

Afin d'obtenir la modélisation la plus robuste possible, certains travaux ont utilisé plusieurs descripteurs combinés pour décrire la région de l'objet cible. Prenons comme exemple, les auteurs dans Maggio et al. (2005), qui ont proposé un modèle d'apparence basé sur la combinaison de l'histogramme pondéré de couleurs et l'histogramme d'orientations du gradient. Un objet cible est approximé par une ellipse. Pour l'histogramme de couleurs, un noyau elliptique est appliqué afin de favoriser les pixels qui sont proches du centre. Par la suite, l'histogramme de couleurs est normalisé. Un vecteur de probabilité est ainsi obtenu pour chaque histogramme. Il est intégré dans un processus de suivi basé sur l'algorithme du Filtre de Particules.

Dans un autre article (Possegger et al. (2014a)), la modélisation de l'objet cible se fait en exploitant certaines caractéristiques géométriques. Pour chaque objet i à l'emplacement x, l'information sur l'occultation $c_{o,i}(x)$ (l'évolution spatio-temporelle des régions où il y a des occultations), la fiabilité du détecteur d'objet $c_{d,i}$ et la prédiction du mouvement de l'objet cible $c_{p,i}$ doivent être estimées. Le score de confiance est le produit de ces trois facteurs :

$$\varphi_i(x) = c_{o,i}(x)c_{p,i}(x)c_{d,i}(x) \tag{2.15}$$

Cette approche de suivi a prouvé que les propriétés géométriques peuvent aider à gérer l'occultation entre les objets cibles.

Les auteurs dans Yao et al. (2010) proposent une méthode du suivi pour des vidéos de sports. Vu la complexité du modèle de joueur, ce dernier est représenté par une combinaison de deux modèles d'apparence : un modèle statique et un modèle dynamique. Le modèle statique permet de coder l'information de couleur et de texture. Les auteurs ont utilisé un histogramme de couleur HSV et un descripteur LBP (Local Binary Patterns). Pour chaque particule pf_t^i, la mesure de similarité statique correspondant au descripteur f est le suivant :

$$V(pf_t^i, f) = 1 - BC(h_T^f, h^f(pf_t^i)) \tag{2.16}$$

Où : BC est le coefficient de Bhattacharyya, h_T^f est le descripteur de modèle de l'objet cible et h^f est le modèle de la particule (objet candidat). Le modèle dynamique est constitué de la position géométrique, la vélocité et le flux optique. Vu le mouvement des athlètes, le déplacement de la cible suit une transition gaussienne. La vélocité est une mixture pondérée entre le flux optique et la vélocité calculée dans la trame précédente. Ces descripteurs sont combinés à l'aide d'un vecteur de coefficient de poids.

Dans Possegger et al. (2014b), l'objet cible est décrit en utilisant : l'histogramme de couleur, les matrices de covariance et l'histogramme de gradients HOG. La combinaison de ces descripteurs est faite en utilisant l'algorithme Adaboost.

Les auteurs dans Breitenstein et al. (2011) exploitent aussi l'idée de combiner plusieurs descripteurs dans le cadre d'un traqueur Filtre de Particules. En fait, l'objet cible est représenté par deux caractéristiques : la caractéristique géométrique (la taille et la position) et la caractéristique de mouvement. Pour la taille, un objet cible possède une taille moyenne sur un nombre de trames (quatre dernières trames). Le modèle de mouvement est basé sur l'hypothèse que les objets cibles ont une vitesse de déplacement constante durant toute la séquence vidéo. De ce fait, la position courante de l'objet cible est égale à la position de l'objet plus la vitesse de déplacement estimée à la trame précédente. Après le calcul de mesure de similarité (en fonction des descripteurs mentionnés ci-dessous), il faut définir le poids de chaque particule. Ce dernier est estimé en fonction de trois termes de confiance (la somme) qui définissent eux-mêmes un modèle d'apparence pour la particule pf associée à l'objet cible tr. Premièrement, le terme de détection qui consiste à calculer la distance entre la particule et l'objet candidat associé. Ce terme est pris en considération seulement dans le cas de la mise en correspondance entre l'objet candidat et l'objet cible. Deuxièmement, le terme de confiance du détecteur d'objets (estimé en fonction de la densité de confiance du détecteur à la position de la particule en question). Troisièmement, le terme de classificateur (estimé en utilisant les

caractéristiques de couleur et de texture associées à la particule). Ces caractéristiques sont ainsi combinées en calculant la somme pondérée par des poids. Ces poids sont déterminés expérimentalement.

Une autre approche similaire a été développée par Yan et al. (2012), les auteurs ont proposé un modèle d'apparence en combinant plusieurs caractéristiques à savoir : l'histogramme de couleur, les caractéristiques de mouvement (incluant la vélocité de mouvement, l'échelle de l'objet et l'angle de déplacement de l'objet) et le flux optique (l'histogramme de mouvement basé sur la magnitude et l'angle du vecteur flux optique). Chaque score de similarité obtenu par une de ces caractéristiques est pondéré par un poids afin de calculer le score de similarité globale. Le poids est déterminé via un processus d'apprentissage discriminatif.

Dans Yang et al. (2009a), les auteurs utilisent un modèle d'apparence basé sur la combinaison de plusieurs caractéristiques dans le cadre du suivi multi objets, mais d'une manière différente. En fait, des modèles d'apparence différents sont utilisés pour représenter des parties particulières du corps humain. Un histogramme de couleur pondéré est calculé pour la partie supérieure du torse (incluant la tête). L'historique de chaque histogramme calculé est utilisé pour calculer le score de similarité (deux modèles d'histogramme moyen sont sauvegardés : à court terme et à long terme). La partie de la tête est représentée par une forme elliptique qui est modélisée par le vecteur d'intensité du gradient. Finalement un ensemble de caractéristiques locales (descripteur SIFT) est estimé pour la partie inférieure du torse. Les points du type SIFT sont estimés sur une grille carrée de pixels de taille 4. Un histogramme SPM (Spatial Pyramid Matching) est calculé pour les points. En gros, chaque partie est représentée par un modèle différent dépendamment de sa localisation et de son importance dans la modélisation globale de l'objet cible. Les détections sont obtenues en utilisant un détecteur de tête humaine basé sur le réseau de neurones CNN (Convolutional Neural Network) sur des séquences vidéo multi vues. Les scores de similarité sont linéairement combinés pour obtenir le score de similarité global.

Dans Kuo and Nevatia (2011), l'objet cible est représenté en utilisant un descripteur de la couleur (histogramme de couleur RGB), un descripteur de la forme (histogramme HOG) et un descripteur de la texture (matrice de covariance). Contrairement aux autres travaux qui utilisent plusieurs descripteurs, un seul descripteur sera sélectionné pour chaque trame. En fait, ces descripteurs sont appris en utilisant l'algorithme Adaboost afin de sélectionner séquentiellement le meilleur descripteur (c'est le descripteur qui donne la meilleure valeur de similarité). Selon Kuo and Nevatia (2011), l'histogramme de couleur est le plus souvent sélectionné comme le meilleur descripteur.

Une autre approche récente [Milan et al. (2014)] est basée sur une fonction de coût obtenue

à l'aide d'une représentation complète de l'objet cible. Outre les descripteurs usuels de la modélisation d'un objet, d'autres descripteurs sont estimés. La fonction de coût (la probabilité de similarité) est estimée en fonction : du terme de donnée qui sert à garder les trajectoires proches des objets candidats, du terme dynamique qui consiste à estimer la contrainte de mouvement basée sur une vitesse de déplacement constant, du terme de l'occultation mutuelle qui est une pénalité de continuité afin de gérer les cas d'occultation entre deux objets cibles, du terme de persistance de trajectoire qui permet d'éviter toutes les fragmentations ou les terminaisons abruptes de trajectoires et finalement du terme de régularisation qui sert à contrôler le nombre de trajectoires crées. En plus des termes définis ci-dessus, l'histogramme gaussien pondéré a été intégré pour créer le modèle d'apparence. Cette pondération a pour but de favoriser les pixels les plus importants (qui appartiennent à l'objet cible) et de défavoriser les pixels de bruit (qui appartiennent à l'arrière-plan). Pour le détecteur d'objets, un histogramme de gradients et un histogramme de flux optiques sont utilisés afin de construire le modèle d'apparence pour le processus de détection. Le modèle d'apparence basé sur le mouvement a été largement utilisé.

Dans Yoon et al. (2015a), le modèle d'apparence de mouvement est basé sur la relation dynamique entre les objets. Ainsi, un réseau de mouvements relatif RMN (*Relative Motion Network*) est construit. Ce dernier permet de sauvegarder les relations spatiales et dynamiques entre les objets en se basant sur la différence entre la vélocité et la position géométrique. Par la suite, pour chaque objet cible, un vecteur de mouvement est obtenu qui contient la relation du mouvement par rapport à chaque autre objet cible dans la séquence vidéo. Le score de similarité est ainsi estimé en comparant la similarité entre les modèles de mouvement relatif des objets cibles. Cette modélisation est intégrée au sein d'un processus de suivi basé sur le filtre bayésien. En plus du modèle de mouvement, l'association des données est réalisée en utilisant la taille de l'objet comme modèle d'apparence ainsi que l'apparence de couleur (histogramme de couleurs).

En récapitulation, il est clair que dans des séquences vidéo complexes (contenant plusieurs objets cibles similaires), il est difficile d'obtenir une représentation robuste en utilisant un seul descripteur. Donc, l'utilisation de plusieurs descripteurs est indispensable pour la construction du modèle d'apparence. En plus, les descripteurs du modèle d'apparence ont une contribution différente dans la création du modèle d'apparence. En fait, par exemple, pour représenter des objets qui ont des mouvements similaires le descripteur de mouvement n'offre pas un modèle discriminatif. De ce fait, les descripteurs doivent intervenir avec des poids différents dans la création du modèle d'apparence tout dépendamment de l'environnement de suivi.

2.4 Association des données

Dans cette thèse on est intéressé au suivi par détection. De ce fait, l'approche de suivi est basée sur l'association de données qui a pour but de relier les résultats d'un détecteur d'objets avec des trajectoires existantes. L'étape de l'association des données présente un défi supplémentaire pour le système de suivi multi objets. En fait, l'association de données est définie par la réponse sur la question suivante : quelle détection doit être reliée à quelle trajectoire ? Il y a trois réponses possibles pour chaque détection : une détection est assignée à une trajectoire existante, une détection n'est assignée à aucune trajectoire et ainsi considérée comme une fausse alarme ou bien une détection est assignée à une nouvelle trajectoire (un nouvel objet cible). Il y a généralement deux types d'association de données : une association type 1-1 (chaque objet cible est associé au plus à un objet candidat et vice versa), association 1-N (chaque objet cible peut être associée à plusieurs objets candidats par exemple dans le cas où les objets peuvent être fragmentés suite aux résultats de soustraction d'arrière-plan incomplets). Il existe plusieurs approches pour l'association des données à savoir les approches qui sont basées sur les algorithmes d'optimisation et les approches qui sont basées sur le principe de l'association locale : *Tracklet*.

2.4.1 Algorithmes d'optimisation

Dans cette section, on va présenter quelques algorithmes d'association de données couramment utilisés dans les systèmes de suivi multi objets. Parmi ces algorithmes, il y a les approches classiques telles que JPDAF (Joint Probabilistic Data Association Filter) [Fortmann et al. (1983)] et MHT (Multiple hypothesis Tracking) [Reid (1979)] et des algorithmes d'optimisation tels que l'algorithme glouton et l'algorithme hongrois [Kuhn (1955)].

— **JPDAF**. L'algorithme de filtrage probabiliste JPDAF prend en considération toutes les détections (ou objets candidats) qui sont présentés dans une fenêtre de validation (les détections qui sont aptes à être assignées à une trajectoire d'un objet cible). En fait, il est basé sur des filtres de mise à jour des objets cibles en prenant en compte d'autres objets cibles. C'est une approche de type association 1—N. Comme les algorithmes d'association des données, chaque observation est pondérée par une probabilité de similarité avec l'objet cible. Au début, un ensemble des hypothèses est généré à partir de toutes les observations (les objets candidats) et les trajectoires obtenues à l'instant courant. Ces hypothèses doivent respecter les deux contraintes : un objet candidat ne peut pas être associé à plusieurs objets cibles dans la même hypothèse et deux objets ne peuvent pas être associés à un objet cible dans la même hypothèse. Par la suite, la probabilité pour chaque association doit être calculée (en sommant, les scores des

objets cibles appartenant à l'hypothèse). Les hypothèses qui ont une faible probabilité sont supprimées. La mise à jour de l'état de l'objet cible se fait en considérant les états des autres objets cibles à l'instant courant. Le temps d'exécution de l'approche JPDAF dépend notamment du nombre d'hypothèses pour chaque objet cible. L'inconvénient majeur de cette approche est le fait qu'elle ne permet pas d'initialiser de nouvelles trajectoires ou de terminer des trajectoires qui existent déjà. Ainsi, cette approche peut être appliquée seulement dans le cas où le nombre des objets cibles est connu à l'avance.

— **MHT.** L'approche du filtre à hypothèses multiples est une association du type 1-1. Elle consiste à générer toutes les hypothèses possibles à chaque instant. Chaque hypothèse doit vérifier les conditions suivantes : il ne faut pas associer un objet cible à plus d'une trajectoire et vice versa. Contrairement au JPDAF, le filtre MHT peut être appliqué si le nombre de trajectoires est inconnu pour chaque instant. En fait, pour chaque trajectoire, un score est calculé qui permet d'ajouter ou d'annuler une trajectoire. Les hypothèses à faible probabilité (la probabilité d'une hypothèse est estimée en fonction de probabilités de trajectoires qui appartiennent à l'hypothèse) peuvent être aussi supprimées afin d'éviter la croissance exponentielle du nombre des hypothèses. Une hypothèse ne peut être validée ou annulée qu'après une période de temps pour considérer l'historique des hypothèses. Ainsi, cette approche ne peut pas être appliquée pour des applications de suivi temps réel.

— **Les algorithmes d'optimisation.** Parmi les algorithmes d'optimisation les plus utilisés dans la littérature, on peut citer l'algorithme glouton et l'algorithme hongrois. Ces deux algorithmes sont basés sur l'estimation d'une matrice de coût (où chaque case de la matrice reflète la relation de similarité entre un objet cible et un objet candidat). Un processus d'optimisation récursif est adapté afin d'avoir les meilleures associations possibles entre les trajectoires qui existent déjà et l'ensemble des objets candidats à chaque instant. Cette optimisation est basée sur le principe de maximiser ou de minimiser d'une façon globale la fonction de coût total pour toutes les assignations possibles (ça dépend de la valeur de similarité ou de dissemblance utilisé). La différence entre ces deux algorithmes d'optimisation est le fait que l'algorithme glouton considère seulement les assignations qui ont un score plus grand (ou plus petit) qu'un seuil, par contre, l'algorithme de hongrois considère toutes les assignations. Les deux algorithmes seront présentés en détail dans le chapitre suivant. Des approches récentes du suivi multi objets ont utilisé ces algorithmes d'optimisation afin d'effectuer l'association des données. Yan et al. (2012) ont proposé une approche du suivi multi objets qui est basée sur l'algorithme d'optimisation hongrois d'une façon hiérarchique. Dans une

autre approche, Breitenstein et al. (2011) utilisent l'algorithme glouton afin d'optimiser les assignations entre les objets cibles et les objets candidats. D'autres méthodes de suivi ont utilisé l'algorithme hongrois telle que Huang et al. (2008a), Yan et al. (2012) et l'algorithme glouton telle que Berclaz et al. (2011), Pirsiavash et al. (2011), Berclaz et al. (2009).

2.4.2 Stratégies avancées : les tracklets

De nouvelles stratégies ont été récemment proposées qui sont basées sur l'association de données locales. En fait, des trajectoires locales sont créées sur un intervalle de temps et puis une association de données globale est faite afin de relier les associations locales qui appartiennent à la même trajectoire appelée : *Tracklet* [(Zhang et al. (2015), Segal and Reid (2013a), Kuo and Nevatia (2011), Poiesi et al. (2013), Yoon et al. (2015b))]. Dans Possegger et al. (2014b), les auteurs ont proposé une approche du suivi basée sur le principe de tracklets. En fait, l'association de données est faite d'une façon hiérarchique sur plusieurs niveaux d'association. Tout d'abord, les tracklets sont créés en utilisant une stratégie de seuillage double. Soit S la matrice d'affinité (contient toutes les valeurs de similarité entre les objets cibles et les objets candidats) et deux objets candidates r_i et r_j, les deux objets sont associés seulement si : leur score de similarité est supérieur à un seuil et leur score de similarité est supérieur à tous les autres scores dans la ligne i et la colonne j avec un autre seuil. Un tracklet est fiable si et seulement s'il respecte les contraintes de relation spatiale et temporelle. En fait, si deux tracklets sont consécutifs (ou avec un petit écart de temps), mais il y a un déplacement spatial important entre les deux trajectoires, les tracklets sont considérés comme non fiables (les objets cibles ont une vitesse constante).

Dans Wang et al. (2015), les auteurs formulent l'association de données en utilisant un graphe cost-flow. Un ensemble de tracklets initiaux est généré. Ainsi, ces tracklets sont raffinés afin d'obtenir des tracklets plus fiables à l'aide d'une approche d'apprentissage. Chaque nœud de graphe représente l'objet candidat ainsi que les transitions entre les états sont estimées en fonction du score de similarité entre les objets candidats. Les tracklets initiaux sont courtes. En fait, les tracklets sont formés en reliant les détections obtenues (seulement celles qui sont obtenues dans des trames consécutives ayant un score de similarité supérieur à un seuil donné). Les tracklets initiaux seront par la suite appris afin d'éliminer ceux qui ne sont pas fiables. En fait, une fonction de distance est calculée pour chaque tracklet en estimant les vecteurs de distances positifs (la différence de similarité entre deux échantillons des détections pour le même objet cible) et les vecteurs de distances négatifs (la différence de similarité entre deux échantillons des détections pour différents objets cibles). Ainsi, la fonction de distance

finale est obtenue en utilisant la distance de Mahalanobis. Les échantillons seront utilisés pour l'apprentissage des tracklets. Deux contraintes seront prises en considération lors de l'apprentissage : un objet cible ne peut pas apparaître à différentes positions en même temps et si un objet quitte la séquence vidéo, il ne peut pas être localisé à l'intérieur de la scène. Finalement, les tracklets assignés au même objet cible seront fusionnés en se basant sur leurs modèles d'apparence.

Dans un autre travail fait par Zhang et al. (2015), l'association des données est effectuée en raccordant les tracklets dans le but de former de longues trajectoires. En fait, il faut trouver l'assignation entre les informations globales (raccordement des tracklets) et les informations locales (raccordement des détections). Les trajectoires sont par la suite mises à jour itérativement jusqu'à la convergence. Dans Yang and Nevatia (2014b), les auteurs ont proposé une approche d'association des données basée sur la liaison de tracklets courts d'une façon hiérarchique. En fait, l'association se fait sur plusieurs niveaux. Initialement, des tracklets sont construits en reliant les détections des trames voisines selon leurs scores de similarité (association de bas niveau). À chaque niveau d'association, les tracklets ayant une probabilité élevée seront fusionnés en utilisant un graphe CRF (Conditional Random Field). Chaque nœud de graphe contient les tracklets fiables (qui peuvent être fusionnés ensemble). Deux tracklets sont considérés comme fiables si et seulement si l'écart de temps entre la trame de début du premier tracklet et la trame de fin pour le deuxième tracklet est inférieur à un seuil donné. Pour la fusion des tracklets, il y a deux types de fusion pour chaque deux tracklets fiables : fusion par la tête ou fusion par la queue. Par la suite, une étape d'apprentissage est nécessaire afin d'améliorer la fusion des tracklets.

Une autre approche similaire a été développée dans Huang et al. (2008b). L'association des données est faite sur 3 niveaux : le niveau bas (des courts tracklets), le niveau moyen (tracklets fiables plus longs) et le niveau supérieur (tracklets affinés en prenant en considération les cas d'occultation, l'ajout ou la suppression des tracklets). Au niveau d'association bas, seules les détections obtenues dans des trames consécutives sont fusionnées afin d'obtenir des tracklets courts. Par la suite, pour chaque tracklet obtenu, la position et la vélocité sont estimées en plus du modèle d'apparence. L'algorithme hongrois est appliqué afin d'obtenir des associations optimales entre les tracklets courts. Du coup, des tracklets de longues trajectoires sont construits. Finalement, un modèle de la structure de la scène est intégré afin de raffiner les relations entre les tracklets et permet d'obtenir les tracklets finaux. Cette structure contient trois cartes (qui décrivent la structure de la séquence vidéo) : une carte pour les nouveaux tracklets (corresponds aux nouveaux objets qui entrent dans la séquence vidéo), une carte pour les tracklets à supprimer (corresponds aux objets qui quittent la séquence vidéo) et une carte pour les occultations entre les tracklets.

Une autre méthode a été introduite dans Kuo and Nevatia (2011) qui est différente des approches décrites ci-dessus où l'association des données est effectuée en suivant une stratégie de reconnaissance des personnes. En fait, les tracklets sont classifiés en deux catégories : les tracklets requêtes et les tracklets galeries. Initialement, les tracklets sont générés en reliant des trajectoires courtes (relier les détections qui ont un modèle d'apparence similaire entre deux trames consécutives). Par la suite, les tracklets doivent être regroupés : un tracklet galerie est un tracklet qui représente une trajectoire plus longue qu'un seuil donné et n'est pas en occultation avec d'autres tracklets. Donc, en d'autres termes, la longueur du tracklet reflète sa fiabilité. Par contre, un tracklet requête est un tracklet qui manque certaines caractéristiques de l'objet cible à qui il correspond. L'association des tracklets est basée sur trois différents descripteurs : le mouvement, l'écart de temps et l'apparence.

2.5 Discussion

D'une manière générale, un algorithme de suivi est effectué sur trois étapes : l'extraction des régions d'intérêts ou des objets candidats (soit dans la première trame seulement ou dans toutes les trames de la séquence vidéo), la construction du modèle d'apparence pour chaque objet (afin de différencier les objets) et la mise en correspondance entre les objets cibles et les objets candidats. Avant tout, il faut penser au type de suivi autrement dit à la forme des régions d'intérêts qu'il faut choisir. En fait, la forme des régions d'intérêts est fortement liée à l'application ou au type des régions d'intérêts. Si par exemple les objets occupent peu de pixels dans l'image, le plus adéquat dans ce cas est de les assimiler à un ensemble de points (le centroïde, les points caractéristiques, etc.). Dans le cas des objets simples et rigides, la meilleure façon est de les représenter en utilisant des formes géométriques (tel qu'un rectangle, une ellipse, etc.). Si les objets à suivre ont une structure complexe et non rigide, généralement il est recommandé d'utiliser le suivi par silhouette ou par contour.

Notre approche de suivi développée dans cette thèse est dédiée pour suivre des objets ayant un format plutôt rigide, plus précisément des personnes. Ceci explique le fait d'utiliser des formes géométriques pour représenter les objets cibles (des formes rectangulaires). De toute façon, les détecteurs produisent des détections de forme rectangulaire en général.

Le premier attribut de suivi est l'extraction des objets candidats. Elle peut être effectuée en utilisant les détecteurs d'objets ou bien les méthodes de soustraction d'arrière-plan. Ces dernières ont plusieurs inconvénients tels que la variation de l'arrière-plan, le bruit de mouvements, etc. Afin d'éviter ce genre de problèmes, les méthodes récentes de suivi multi objets sont basées sur l'utilisation d'un détecteur d'objets (*Tracking-by-Detection*). Dans ce travail, on soutient que les méthodes de suivi multi objets basés sur les détections offrent une amé-

lioration importante de résultats du suivi, car il n'y a pas de problèmes de fragmentation. En fait, un détecteur d'objet est généralement basé sur l'apprentissage ce qui permet d'avoir seulement des objets personnalisés. Par contre, une limitation de l'utilisation d'un détecteur d'objets est la qualité de détection. En fait, on peut trouver de fausses alarmes, des détections manquantes, etc.

Le deuxième attribut de suivi est le modèle d'apparence qui présente un grand enjeu pour le suivi multi objets. Le fait de suivre plusieurs objets de même type (soit des personnes, soit des véhicules, etc.), c.-à-d. qu'ils ont des caractéristiques communes telles que la texture, les habits extérieurs, la vitesse et le type de mouvement (marcher, courir, etc.), demande un modèle robuste. De ce fait, dans une scène complexe, utiliser un seul descripteur afin de mesurer la similarité reste peu fiable et ne permet pas d'obtenir de bonnes performances de suivi. Donc, l'intégration de plusieurs descripteurs à la fois est indispensable afin de représenter le modèle d'apparence. Selon la revue de la littérature, les approches de suivi basées sur l'interaction entre des descripteurs multiples donnent des résultats prometteurs. Il est clair que chaque descripteur a une contribution différente à la construction du modèle d'apparence. Ainsi, afin d'utiliser plusieurs descripteurs, il faut prendre en considération le poids de chaque descripteur. Par ailleurs, les interactions entre les objets cibles (les occultations partielles, les occultations totales, etc.) et la variation de modèle de l'objet (le changement d'échelle, le point de vue, la luminosité, etc.) présentent des problèmes difficiles dans le suivi multi objets. Ces défis peuvent atténuer la performance de suivi multi objets même en utilisant une représentation robuste des objets cibles.

Le troisième attribut est l'association des données où aussi appelé la mise en correspondance. Cette étape est primordiale dans le suivi basé sur la détection. En fait, elle consiste à identifier les objets afin de construire une trajectoire complète. La plupart des travaux existants sont basés sur la mise en correspondance entre la liste des détections et la liste des traqueurs (le résultat d'un algorithme de suivi mono-objet) telle qu'un Filtre de Particules. Ceci revient à suivre chaque objet séparément et puis relier les résultats afin d'obtenir une localisation continue. Ce qui rend le processus de suivi multi objets assez complexe. Autre point, il ne faut pas oublier les cas de coupures de trajectoires dues au problème déjà mentionnés ci-dessus. Ainsi, certaines approches utilisent des algorithmes d'optimisation usuels tels que l'algorithme de glouton ou l'algorithme hongrois. D'autres suivent une stratégie avancée basée sur la liaison des trajectoires locales (en termes de temps) appelés *tracklets* afin d'obtenir l'allure des positions d'un objet cible.

Dans ce travail, nous soutenons que la construction d'un modèle d'apparence robuste devrait être adressée en premier. En fait, pour l'étape d'association des données, le modèle d'appa-

rence sera introduit comme étant une donnée d'entrée afin de trouver la similarité entre les objets en question. Le modèle d'apparence doit être ainsi mis à jour pour prendre en mesure les changements qui peuvent affecter l'objet cible. De ce fait, cette mise à jour est indispensable, mais il faut prendre en note que cette étape élémentaire doit être appliquée seulement dans le cas de bon suivi (une bonne localisation de l'objet cible est obtenue à un moment donné de la séquence vidéo). Ainsi découle notre deuxième contribution majeure concernant l'association des données. Il faut concevoir une telle méthode qui permet d'optimiser les assignations entre les objets cibles et les objets candidats et de traiter les cas d'ambiguïtés entre les assignations.

En s'inspirant des travaux existants, on développe une méthode du suivi multi objets en améliorant les trois aspects décrits ci-dessus. Tout d'abord, notre méthode de suivi est une méthode de suivi par détection. Par la suite, on construit un modèle d'apparence robuste aux problèmes qui peuvent l'affecter. En fait, notre modèle d'apparence est basé sur l'intégration de plusieurs descripteurs où chaque descripteur fournit une valeur de probabilité qui contribue à la localisation de l'objet cible. L'ensemble de descripteurs représente à la fois les propriétés intrinsèques (l'histogramme de couleur et la représentation éparse) et les propriétés de mouvements (le flux optique et les coordonnées géométriques). L'étape de l'association des données repose sur l'utilisation d'un algorithme d'optimisation (l'algorithme hongrois), mais aussi elle intègre des post-traitements qui ont pour but la gestion des assignations. C'est une étape de filtration qui permet de supprimer ou d'ajouter des assignations selon le cas (si un nouvel objet cible apparaît dans la séquence vidéo ou si un objet existant quitte la scène). Dans le cas où le chemin de l'objet cible est perdu, une étape d'interpolation basée sur l'estimation de vélocité est effectuée.

CHAPITRE 3 MÉTHODOLOGIE

3.1 Vue d'ensemble et motivation

Les principales étapes de notre méthode sont décrites à la figure 3.1. Un objet à suivre est une ROI (région d'intérêt) qui est définie par une boîte rectangulaire située à l'intérieur d'une trame. L'ensemble des caractéristiques de l'objet cible est initialisé par sa première apparition dans une trame par des caractéristiques calculées pour chaque détection. Les détections sont obtenues dans chaque trame en utilisant un détecteur d'objets. Dans le but de diminuer le nombre de fausses détections, les détections sont filtrées en éliminant celles ayant une taille inappropriée ou une faible valeur de confiance de classification. Initialement, un ensemble d'un nombre connu de trajectoires est construit dans lequel chaque objet cible est défini par un état et un ensemble de caractéristiques. L'ensemble des objets cibles seront mise à jour dynamiquement pour refléter les changements de modèles d'apparence ainsi que pour traiter les problèmes connus d'un algorithme MOT. En plus d'un modèle de couleur et d'un modèle épars de l'objet cible, nous proposons également un modèle de mouvement qui inclut une caractéristique de flux optique et une caractéristique spatiale. Grâce à ce modèle du mouvement, on évite les fausses associations (appariements) entre l'ensemble des objets cibles et l'ensemble des objets candidats (un objet candidat est une détection). Pour chaque trame, une fonction d'affinité est calculée. Cette fonction reflète la similarité entre les objets cibles et l'ensemble des objets candidats en se basant sur leurs modèles d'apparence. Plus précisément, le modèle d'apparence de l'objet cible est défini par les quatre caractéristiques suivantes :

1. Un histogramme de couleur H_c est utilisé afin de coder les informations de couleur de l'objet cible. La similarité de couleur entre l'objet cible et un objet candidat est évaluée en calculant la distance euclidienne entre les histogrammes de couleur.
2. L'erreur de la représentation éparse p qui reflète l'erreur de projection linéaire de l'objet candidat dans l'espace des gabarits de l'objet cible. En effet, chaque objet candidat est projeté d'une façon linéaire et éparse dans l'espace des gabarits de l'objet cible qui sont générés linéairement à partir de la dernière position de la boîte englobante de l'objet cible.
3. Un histogramme orienté du flux optique H_m est utilisé pour coder les propriétés du mouvement de l'objet cible.
4. Un terme de consistance spatial $\vec{d}$ qui reflète la corrélation géométrique entre un objet cible ainsi que la liste des objets candidats en calculant la distance euclidienne entre

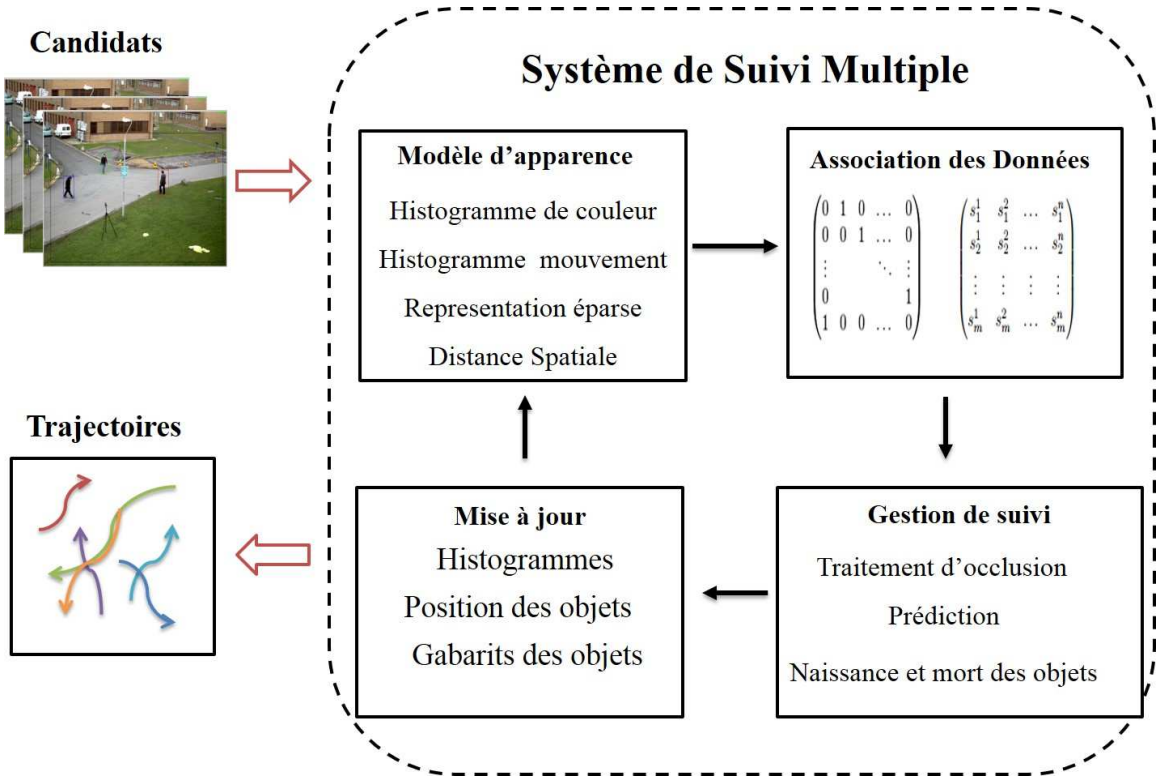


Figure 3.1 Aperçu général de la méthode

le point du centre de l'objet cible et le point du centre de chaque objet candidat.

Après que le modèle représentant l'objet cible est construit, l'association des données est une étape cruciale qui permet de mettre en correspondance la liste des objets cibles avec la liste des objets candidats tout en se basant sur la similarité entre les modèles d'apparence. L'association des données sera faite en deux niveaux : niveau actif et niveau caché. Au premier niveau, tous les objets cibles qui sont visibles (ou en état actif) sont mis en correspondance avec tous les objets candidats. Au deuxième niveau, seulement les objets cibles qui sont cachés (ou en état inactif) seront mis en correspondance avec seulement les candidats qui n'ont pas été appariés au premier niveau. Tous les appariements valides entre les objets cibles et les objets candidats obtenus aux deux niveaux d'association des données seront combinés afin d'avoir une association de données globale pour tous les objets cibles. Ainsi, une matrice d'association est obtenue dont les lignes représentent les objets cibles et les colonnes représentent les objets candidats. Un algorithme d'optimisation global doit être appliqué afin d'obtenir les meilleurs appariements possibles entre les deux listes d'objets. Pour la simplicité, on a appliqué un algorithme d'optimisation tel que l'algorithme de recherche vorace (*Greedy Search*) ou l'algorithme hongrois. Les appariements entre la liste des objets cibles et la liste des objets candidats sont obtenus en se basant sur le calcul de descripteurs

d'apparence pour chaque objet. Ces appariements doivent être filtrés dans le but d'éviter les faux appariements. Ceci est réalisé par la création d'un statut pour chaque objet cible. Le statut d'un objet représente son état actuel dans le cycle de vie d'un objet cible. En se basant sur le statut de l'objet cible ainsi que la valeur de score de similarité obtenue, un objet existant peut être supprimé ou un nouvel objet peut être ajouté à la liste des objets cibles.

3.2 Modèle d'un objet cible

Un objet cible est représenté par quatre descripteurs indépendants qui reflètent les propriétés intrinsèques (le modèle d'apparence de couleur et le modèle d'apparence éparse) et les propriétés du mouvement (le flux optique et la caractéristique spatiale). Ces descripteurs sont incorporés ensemble pour définir un modèle global de l'objet cible. Le choix de cet ensemble de descripteurs complémentaires et indépendants est justifié par le fait que l'utilisation de caractéristiques multiples fournit des informations complémentaires sur l'objet. Grâce à cet ensemble des caractéristiques, on obtient un modèle discriminant efficace pour tous les objets cibles. Un modèle d'apparence F^t est construit à chaque trame t.

$$F^t = [H_c, p, H_m, \vec{d}] \tag{3.1}$$

Où H_c est la concaténation des LSH (Locality Sensitive Histograms) au niveau de chaque pixel, p est la probabilité de l'erreur de la projection éparse dans l'espace des gabarits, H_m est l'histogramme d'orientation de flux optique et $\vec{d}$ est le vecteur de la distance euclidienne entre le point centre de l'objet cible et le point centre de l'objet candidat.

3.2.1 Modèle d'apparence de couleur

À cause de leurs simplicités, les histogrammes de couleurs sont souvent utilisés pour représenter un objet cible. Les histogrammes conventionnels reflètent le nombre d'occurrences d'une valeur particulière d'intensité dans l'image. La valeur de la classe b est incrémentée si la valeur d'intensité d'un pixel donné appartient à cette classe b. Ainsi, l'histogramme de couleur H correspondant à l'image I est un vecteur où chaque élément est défini par :

$$H_b = \sum_{q=1}^{W} Q(I_q, b), b = 1, \cdots, B, \tag{3.2}$$

Où W est le nombre de pixels dans l'image et $Q(I_q, b)$ est une fonction qui est égale à zéro sauf si la valeur d'intensité I_q appartient à la classe b. Cependant, la représentation de l'image par l'histogramme de couleur ne prend pas en considération la disposition spatiale de chaque

pixel dans l'image. Pour cette raison, on utilise une approche récente pour le calcul de l'histogramme de couleur appelée Locality Sensitive Histogram (LSH) (développée par He et al. (2013)). L'histogramme de couleur est construit à chaque pixel de l'objet cible. Le LSH est défini par un ensemble d'histogrammes locaux pour chaque pixel de la région englobant l'objet cible. Pour le suivi multi objets, les pixels de l'objet cible qui sont à l'intérieur d'une région de voisinage ne devraient pas avoir une contribution égale. En fait, plus les pixels sont proches du pixel du centre de la région de voisinage plus que leurs contributions devraient être élevées. Le LSH à un pixel donné est la somme des valeurs d'intensité pondérées dans une région de voisinage. Mathématiquement, soit H_{px}^{E} est le LSH au pixel px dans la région de voisinage E :

$$H_{px}^{E} = \sum_{q=1}^{W} \beta^{|px-q|}.Q(I_q, b), b = 1, \cdots, B, \tag{3.3}$$

Où $\beta \in [0, 1]$ est un paramètre qui contrôle le poids de chaque pixel en fonction de sa distance du pixel px et $Q(I_q, b)$ est une fonction qui est égale à zéro sauf si la valeur d'intensité I_q appartient à la classe b. Le LSH peut être calculé en se basant sur la contribution des pixels du côté gauche (les pixels situés à gauche du pixel px) et du côté droit (les pixels situés à droite du pixel px). Donc, le LSH est égal :

$$H_{px}^{E}(b) = H_{px}^{E,left}(b) + H_{px}^{E,right}(b) - Q(I_{px}, b), \tag{3.4}$$

où :

$$H_{px}^{E,left}(b) = Q(I_{px}, b) + \beta.H_{px-1}^{E,left}(b), \tag{3.5}$$

$$H_{px}^{E,right}(b) = Q(I_{px}, b) + \beta.H_{px+1}^{E,rigth}(b), \tag{3.6}$$

Les pixels du côté droit ne contribuent pas dans le calcul de $H_{px}^{E,left}$ et les pixels du côté gauche ne contribuent pas dans le calcul de $H_{px}^{E,right}$. Comme connu, les histogrammes doivent être normalisés. Pour la normalisation de LSH, il faut calculer le facteur de normalisation pour chaque histogramme, autrement dit, un facteur de normalisation en chaque pixel. Ce facteur est obtenu par la sommation des valeurs de chaque classe dans l'histogramme. Ainsi, on suppose que n_{px} est le facteur de normalisation au pixel px :

$$n_{px} = \sum_{b=1}^{B} H_{px}^{E}(b) = \sum_{b=1}^{B}\sum_{q=1}^{W} \beta^{|px-q|}.Q(I_q, b) = \sum_{q=1}^{W} \beta^{|px-q|}.(\sum_{b=1}^{B} Q(I_q, b) \tag{3.7}$$

Puisqu'un pixel dans l'image appartient à une seule classe, donc le facteur de normalisation au pixel px est égal :

$$n_{px} = \sum_{q=1}^{W} \beta^{|px-q|}. \tag{3.8}$$

On constate que le facteur de normalisation ne dépend pas du nombre de classes utilisé dans l'histogramme. Comme le LSH d'un pixel donné peut être calculé en fonction de pixels situés sur les côtés gauche et droit, le facteur de normalisation peut être aussi calculé de la façon suivante :

$$n_{px} = n_{px}^{left} + n_{px}^{right} - 1, \tag{3.9}$$

avec :

$$n_{px}^{left} = 1 + \beta.n_{px\text{-}1}^{left}, \tag{3.10}$$

$$n_{px}^{right} = 1 + \beta.n_{px\text{-}1}^{right}, \tag{3.11}$$

Une fois que le descripteur de couleur est calculé en fonction de l'histogramme LSH, une étape de comparaison de descripteurs doit être faite afin de trouver la similarité entre deux régions d'intérêts (une région qui englobe l'objet cible et une région qui englobe l'objet candidat). Afin de comparer deux LSH, on utilise la distance l_1 :

$$D(H_t, H_c) = \sum_{b=1}^{B}(|H_t(b) - H_c(b)|), \tag{3.12}$$

Avec H_t est l'histogramme LSH de la région qui englobe l'objet cible et H_c est l'histogramme LSH de la région qui englobe l'objet candidat.

3.2.2 Modèle de représentation éparse

Les modèles d'apparence basés sur la représentation éparse ont attiré beaucoup d'attention au cours des dernières années. La représentation éparse d'un objet cible sert à trouver une approximation éparse dans l'espace de gabarits (figure 3.2).

Les gabarits ont la même taille que la taille de l'objet cible. L'espace des gabarits est composé d'un ensemble de gabarits principaux (gabarits qui représentent l'objet cible) et des gabarits appelés gabarits triviaux (des gabarits qui contiennent des pixels provenant de l'arrière-plan) (figure 3.3). Les gabarits principaux dans la trame courante sont construits par la génération de translations successives autour de la région qui englobe l'objet cible dans la trame précédente. Un gabarit trivial est un vecteur avec seulement une entrée non nulle (différente

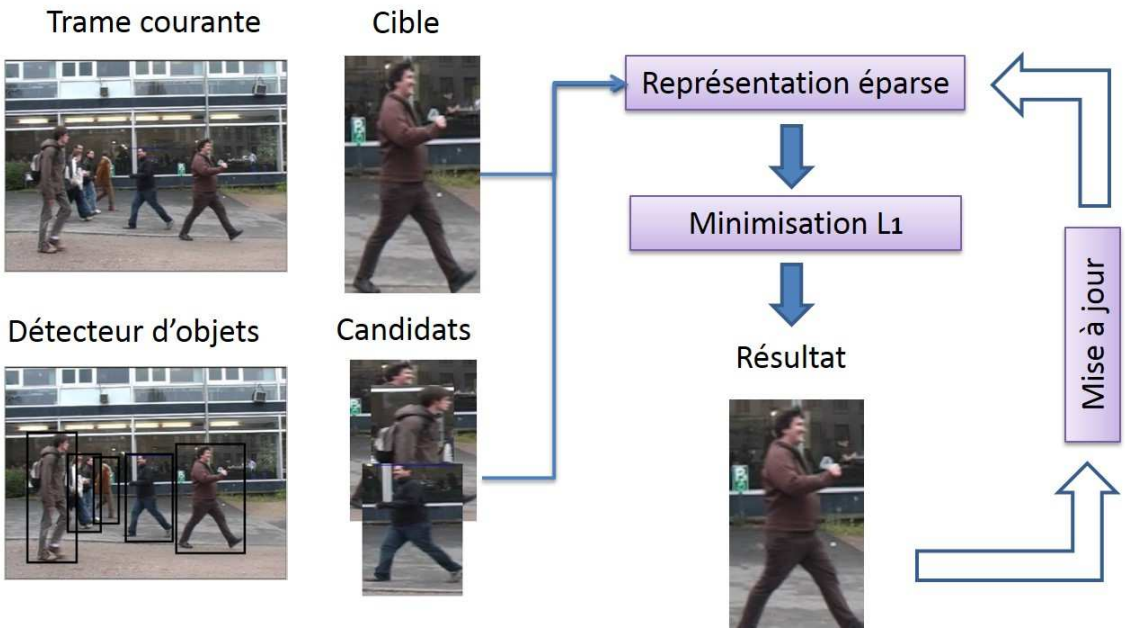


Figure 3.2 La représentation éparse d'un objet

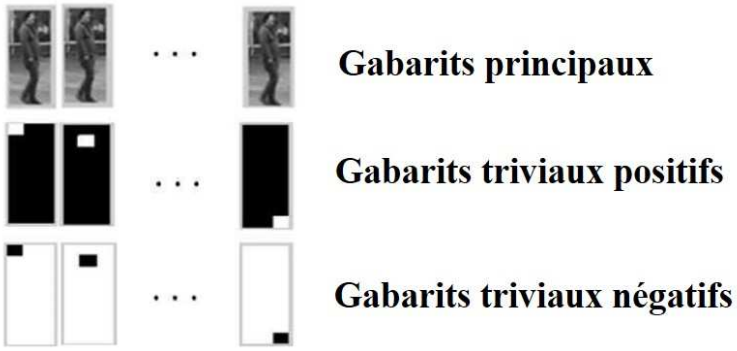


Figure 3.3 Dictionnaire des gabarits.

de zéro). Les gabarits triviaux servent à traiter les cas d'occultation ou de bruit. Le nombre de gabarits triviaux est beaucoup plus grand que le nombre de gabarits principaux. Ainsi, afin de trouver la similarité entre deux objets d'intérêt, un objet est représenté par une combinaison linéaire de l'ensemble des gabarits. Afin de trouver la similarité en fonction de la représentation éparse, il faut suivre les deux étapes suivantes :

— **Représentation éparse d'un objet cible**. Dans notre méthode de suivi multi objets (MOT), on adopte et modifie une technique récente de représentation éparse développée dans Bao et al. (2012). Dans Bao et al. (2012), les objets candidats sont générés à l'aide de l'approche du filtre des particules. En fait, pour chaque objet cible, un ensemble de particules ayant le même poids (initialement) est construit récursivement. Le poids des particules est proportionnel à la valeur de probabilité qui lui correspond. Ainsi, la probabilité postérieure qui reflète la similarité entre l'objet cible et l'ensemble des particules doit être estimée. Dans notre approche on propose une modification à la représentation éparse proposée dans Bao et al. (2012). Les objets candidats dans chaque trame sont utilisés comme étant des particules candidates. Ainsi, pour chaque objet cible, un score de similarité doit être calculé entre l'objet candidat ainsi que l'objet cible en se basant sur l'erreur de la représentation éparse. Le modèle de représentation éparse vise à calculer l'erreur de projection d'un objet candidat dans le dictionnaire de gabarits de l'objet cible. Un bon candidat est un candidat qui est représenté efficacement par seulement les gabarits principaux. En fait, les coefficients des gabarits triviaux tendent vers zéros sauf dans le cas d'occultation, où un nombre limité de coefficients de gabarits triviaux sont différents de zéro. Par contre, un mauvais candidat est un candidat qui est représenté par une représentation dense qui reflète la dissemblance par rapport aux gabarits de l'objet cible. En fait, les coefficients des gabarits principaux tendent vers zéros tandis que les coefficients des gabarits triviaux sont plus significatifs. Soit un ensemble de n gabarits de l'objet cible $T = \{t_1, t_2, ..., t_n\} \in \mathbb{R}^{d \times n}$. Soit x_t décrivant le modèle d'apparence de l'objet cible dans la trame t et soit $Y = \{y_1, y_2, ..., y_m\}$ un ensemble des objets candidats. Un candidat y_i est linéairement projeté dans l'espace des gabarits suivant l'équation ci-dessous :

$$y_i = \vec{a}T = a_1 t_1 + a_2 t_2 + ... + a_n t_n, \tag{3.13}$$

Avec : $\vec{a} = (a_1, a_2, ..., a_n)^{'} \in \mathbb{R}^n$ est le vecteur des coefficients pour chaque gabarit. Dans les scénarios réels, les objets cibles peuvent être en occultation ce qui résulte en un modèle de l'objet cible bruité. Afin de prendre en considération les erreurs de

projection, la représentation éparse de l'objet candidat y est :

$$y = \vec{a}T + \varepsilon \tag{3.14}$$

Où $\varepsilon = \vec{e}I$ est un vecteur qui reflète la projection de l'objet candidat dans l'espace des gabarits triviaux, $\vec{e} = (e_1, e_2, ..., e_d)$ est le vecteur des coefficients des gabarits triviaux et $I = \{i_1, i_2, ..., i_d\} \in \mathbb{R}^{d \times d}$ est l'ensemble des gabarits triviaux. Si l'objet cible est totalement visible (n'est pas occulté), ε est égale à zéro. Le nombre des gabarits triviaux est beaucoup plus grand que le nombre des gabarits principaux ($d >> n$). Les gabarits triviaux peuvent être classés en deux catégories : des gabarits triviaux qui sont plus similaires aux gabarits principaux et des gabarits triviaux qui sont moins similaires aux gabarits de l'objet cible. En fait, on peut trouver des gabarits triviaux qui sont similaires à l'objet cible, mais avec des intensités inversées (par exemple dans le cas d'ombre). Afin d'éviter un tel cas d'échec, des contraintes de non-négativité seront ajoutées dans l'approximation du modèle épars de l'objet candidat. Donc, on peut dire que si les coefficients des gabarits triviaux sont positifs alors ces gabarits sont appelés des gabarits positifs (des gabarits qui sont liés positivement au modèle de l'objet cible) et si les coefficients des gabarits triviaux sont négatifs alors ces gabarits sont appelés des gabarits négatifs (des gabarits qui sont liés négativement au modèle de l'objet cible). Ainsi deux types de gabarits triviaux doivent être inclus dans le modèle épars de l'objet candidat :

$$y = \vec{c}B, \tag{3.15}$$

Avec : $B = [T, I, -I] \in \mathbb{R}^{d \times (n+2d)}$ est l'ensemble de tous les gabarits incluant les gabarits triviaux positifs et négatifs et $\vec{c} = [a, e^+, e^-]' \in \mathbb{R}^{(n+2d)}$ est le vecteur de coefficients.

— **Probabilité d'erreur de la projection éparse**. Une fois que la projection linéaire de l'objet candidat est faite dans l'espace des gabarits de l'objet cible, un vecteur d'erreur d'approximation éparse sera calculé par la résolution de l'équation (3.15). En fait, un bon candidat doit avoir un vecteur de coefficient c dont les coefficients des gabarits triviaux ont un nombre limité des coefficients non nuls (dans le cas d'occultation). La similarité entre le modèle de l'objet cible x_t et le modèle de l'objet candidat y_i est calculée en utilisant la minimisation l_1 :

$$min \left\| Bc - y_i \right\|_2^2 + \lambda \left\| c \right\|_1 ; s.t.\ c \geqslant 0 \tag{3.16}$$

Avec : $||.||_2$ et $||.||_1$ sont les normes l_2 et l_1 utilisés pour résoudre le problème de

minimisation. La probabilité de vraisemblance $p(y_i|x_t)$ entre le modèle épars de l'objet candidat y_i et le modèle épars de l'objet cible x_t à la trame t est donc :

$$p(y_i|x_t) = \frac{1}{\tau} \exp[-\alpha \left\| y_i - cT \right\|_2^2], \tag{3.17}$$

où c est la solution de l'équation (3.16), α est une constante et τ est un facteur de normalisation. La probabilité de vraisemblance est inversement proportionnelle à l'erreur de reconstruction éparse c.-à-d. le candidat avec une erreur minimale de reconstruction éparse possède une probabilité de vraisemblance élevée. Un bon candidat doit être approximé par un faible vecteur de coefficients de gabarits triviaux tandis que les coefficients d'un mauvais candidat ont une allure dense pour tous les types de gabarits. L'objet candidat qui a une erreur de projection minimale aura une probabilité de similarité la plus élevée. Comme dans tous les modèles d'apparence, le modèle d'apparence éparse doit être mis à jour. La mise à jour des gabarits est effectuée seulement dans le cas de bon suivi afin de prendre en considération les changements du modèle d'apparence (si on trouve un candidat dont son approximation éparse est similaire au modèle d'apparence de l'objet cible). En fait, si le score de similarité entre un objet candidat et l'objet cible est supérieur à un seuil donné, l'espace des gabarits de l'objet cible sera mis à jour en fonction de la nouvelle localisation de l'objet en question (un nouveau dictionnaire des gabarits sera généré).

3.2.3 Modèle du mouvement

Dans notre travail, on propose de représenter le modèle d'un objet cible aussi par ses propriétés de mouvement. On utilise une approche globale afin d'estimer les propriétés de mouvements : c'est le flux optique. Le flux optique est un champ des vecteurs de mouvement 2D (deux dimensions) extrait à partir de deux trames consécutives. Le flux optique doit être calculé sur une région de l'image (pour prendre en considération une région de voisinage) et non pas sur un pixel de l'image. Pour le calcul du flux optique, on utilise l'approche de *Horn-Schunck* (Horn and Schunck (1981b)). Le calcul du flux optique se fait en résolvant une équation différentielle qui décrit la valeur de l'intensité différentielle en chaque pixel de la région cible. Mathématiquement, le flux optique d'une région de l'image est un vecteur $\vec{v} = [v_x, v_y]$ dont :

$$v_x = v_x^{avg} - f_x[f_x v_x^{avg} + f_y v_y^{avg} + f_t]/(\alpha^2 + f_x^2 + f_y^2), \tag{3.18}$$

$$v_y = v_y^{avg} - f_y[f_x v_x^{avg} + f_y v_y^{avg} + f_t]/(\alpha^2 + f_x^2 + f_y^2), \tag{3.19}$$

Avec v_x^{avg} est le déplacement moyen suivant l'axe des x, v_y^{avg} est le déplacement moyen suivant l'axe des y. La moyenne est calculée à l'intérieur d'une région de huit voisins où chaque point du voisinage à un coefficient selon le noyau dans la figure 3.4.

1/12	1/6	1/12
1/6	-1	1/6
1/12	1/6	1/12

Figure 3.4 Exemple pour l'histogramme de mouvement.

f_x est l'estimation de la dérivée partielle suivant l'axe des x. f_y est l'estimation de la dérivée partielle suivant l'axe des y. f_t est l'estimation de la dérivée partielle suivant le temps t. Après l'estimation de vecteur de champ de mouvement pour la région qui englobe l'objet cible, un descripteur de mouvement est obtenu par le calcul d'histogramme de flux optique orienté (HOOF) (Chaudhry et al. (2009)). Initialement, un vecteur de flux optique est estimé en chaque pixel de la région de l'objet cible. Ce vecteur sera représenté par sa force $\sqrt{v_x^2 + v_y^2}$ ainsi que sa direction $\theta = \tan^1(\frac{v_y}{v_x})$. Dans l'histogramme orienté de flux optique, chaque vecteur est classé selon son angle par rapport à l'axe horizontal.

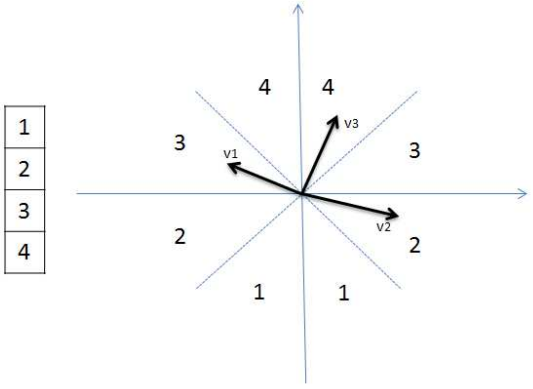


Figure 3.5 Noyau du voisinage pour l'histogramme de flux optique.

Dans l'exemple de la figure 3.5, le vecteur de flux optique $\vec{v_1}$ sera classé dans la classe 3 alors que $\vec{v_2}$ est classé dans la classe 2. En général, un vecteur de flux optique appartient à l'intervalle b si et seulement si son angle vérifie cette condition :

$$-\frac{\pi}{2} + \pi\frac{b-1}{B} \leqslant \theta < -\frac{\pi}{2} + \pi\frac{b}{B} \tag{3.20}$$

Où $1 \leqslant b < B$ et B est le nombre total de classes utilisées dans l'histogramme. Afin d'être robuste aux changements d'échelle, l'histogramme orienté du flux optique sera normalisé. Dans le but d'utiliser l'histogramme HOOF pour comparer les similarités entre les objets cibles et les objets candidats, on doit comparer les histogrammes en utilisant l'équation suivante :

$$D(H_t^m, H_c^m) = \sum_{b=1}^{B}(|H_t^m(b) - H_c^m(b)|), \tag{3.21}$$

Où H_t^m est le modèle de mouvement de l'objet cible et H_c^m est le modèle de mouvement de l'objet candidat.

3.2.4 Modèle spatial

La caractéristique spatiale est aussi intégrée dans la liste des caractéristiques pour représenter le modèle global d'un objet (un objet cible ou un objet candidat). Les caractéristiques de modèle épars, de modèle de couleur et de modèle de mouvement sont utilisées afin de décrire le modèle propre d'un tel objet tandis que la caractéristique spatiale est utilisée afin de définir la contrainte géométrique. En utilisant la contrainte géométrique, on peut rejeter les candidats qui sont improbables même si leurs modèles d'apparence sont les plus similaires au modèle d'apparence de l'objet cible. Cette caractéristique est basée sur l'hypothèse suivante : un objet cible ne se déplace pas d'une façon soudaine d'une trame à l'autre. En fait, les objets se déplacent avec une vitesse constante. Ainsi, un candidat qui est plus proche de l'objet cible en termes de la distance euclidienne est un candidat plus probable selon sa caractéristique spatiale. La caractéristique spatiale sert à étudier la corrélation spatiale entre les différentes positions de l'objet cible dans le temps. La contrainte spatiale est intégrée dans deux parties de notre algorithme de suivi : l'étape de la construction du modèle d'apparence d'un objet et l'étape d'association des données. Ceci nous permet d'explorer les relations spatiales entre les objets cibles et les objets candidats. L'information spatiale est utilisée dans le but d'éviter les correspondances non correctes entre la liste des candidats et la liste des objets cibles et pour observer la dynamique de mouvements des objets cibles (voir figure 3.6). Le modèle spatial du bonhomme en violet (figure 3.6 gauche) est un vecteur ou chaque élément est la distance euclidienne entre son point de centre et les points de centres de tous les bonshommes dans la figure 3.6 droite (les figures figure 3.6 gauche et droite présentent un exemple de détections entre deux trames successives).

La caractéristique spatiale est encodée à l'aide d'un vecteur de coordonnées géométriques (i_x, i_y, w, h) pour chaque objet à travers le temps où (i_x, i_y) sont les coordonnées géométriques (suivant l'axe des abscisses et l'axe des ordonnées), (w, h) est les dimensions de l'objet cible

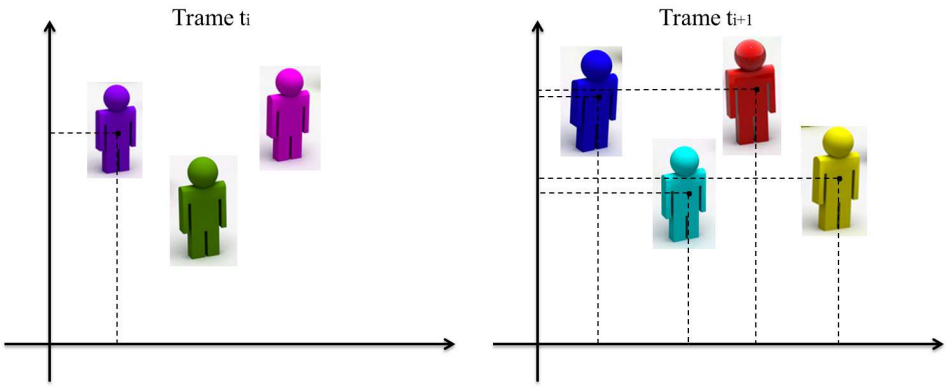


Figure 3.6 Modèle spatial d'un objet cible.

(largeur et longueur). La similarité de descripteur spatial entre un objet cible et un objet candidat est ainsi le vecteur de la distance euclidienne $\vec{d}$ entre les points centres de chaque objet :

$$d_i(j) = \sqrt{(i_x - j_x)^2 + (i_y - j_y)^2}, \tag{3.22}$$

Où (i_x, i_y) et (j_x, j_y) sont les coordonnées du point du centre de l'objet cible i et l'objet candidat j respectivement. La caractéristique spatiale doit être calculée entre deux trames successives. De ce fait, cette caractéristique est prise en considération dans l'estimation de similarité entre les objets candidats et les objets cibles seulement dans le cas de non-occultation (si l'objet cible est visible).

3.3 Association des données

Comme on l'a déjà introduit, le problème du suivi multi objets est reformulé en problème d'association des données. L'association des données est l'étape pour trouver la réponse à la question suivante : quel objet candidat doit être assigné à quel objet cible ? Une fois que le modèle d'apparence est estimé pour chaque objet cible en fonction de différents descripteurs (décrits dans la section précédente), une valeur de similarité pour chaque descripteur entre l'ensemble les objets cibles et les objets candidats doit être calculée. Les valeurs de similarité pour chaque descripteur sont fusionnées afin d'avoir une seule valeur de similarité globale en fonction des différents descripteurs. La valeur de similarité globale est obtenue à l'aide d'une fonction d'affinité. Cette dernière est la somme pondérée de chaque valeur de similarité dont chacun a un poids.

Tout d'abord, une étape d'initialisation est réalisée à la première trame (le début de la séquence vidéo). Un ensemble des objets cibles est construit à partir des objets détectés à

la première trame. À partir de la deuxième trame, un ensemble des objets candidats est construit à partir de l'ensemble des objets détectés dans chaque trame. Chaque objet cible est défini par un ensemble des caractéristiques à savoir : les coordonnées géométriques, les descripteurs du modèle d'apparence (couleur, éparse, mouvement et spatiale), un état, et un historique de la dernière position (où était l'objet cible). L'étape d'association des données est un processus de mise en correspondance entre l'ensemble des objets cibles et la liste courante des objets candidats dans le but de définir la nouvelle position pour chaque objet cible.

La mise en correspondance se fait en se basant sur la matrice d'affinité. La matrice d'affinité est une matrice des valeurs de similarité (voir section 3.3.2). Lors de la mise en correspondance, un objet cible doit être assigné à un et un seul objet candidat. En fait, un objet cible sera mis en correspondant avec l'objet candidat qui a le maximum de probabilité de vraisemblance. Afin de gérer les cas d'occultations, l'association des données est faite hiérarchiquement en deux niveaux :

- **Premier niveau** : la mise en correspondance est appliquée seulement avec la liste des objets candidats pour la liste des objets cibles qui sont dans l'état visibles.
- **Deuxième niveau** : la mise en correspondance est appliquée avec seulement les objets candidats qui n'ont pas été assignés au premier niveau pour la liste des objets qui sont en occultation.

Afin d'éviter les fausses assignations et gérer les cas d'occultation, un processus de gestion de suivi doit être appliqué. La gestion de suivi des objets multiples est réalisée en se basant sur : 1) l'interpolation des trajets des objets cibles perdus, et 2) la naissance et la mort des objets cibles.

3.3.1 Fonction d'affinité

La fonction d'affinité est une fonction qui permet de calculer la probabilité de vraisemblance entre deux objets (un objet cible et un objet candidat). La probabilité de vraisemblance reflète la similarité entre deux objets en se basant sur l'ensemble des caractéristiques déjà présentées. Cette fonction est une fusion de différentes valeurs de probabilité de vraisemblance obtenues pour chaque caractéristique afin d'avoir une probabilité de similarité globale. Dans cette thèse, on a procédé à la fusion de deux façons différentes :

1. **Règle de combinaison de Dempster-Shafer :** La règle de combinaison de Dempster-Shafer est souvent utilisée dans la fusion des classificateurs. On a intégré dans un premier temps cette règle dans le cadre de notre suivi multi objets. Comme son nom l'indique, cette théorie a été développée par Arthur P.Dempster et Glenn Shafer dans les années soixante et soixante-dix (Sergey et al. (2008)). Les premières utilisations de

cette théorie de combinaison ont été dans le domaine de l'intelligence artificielle, et par la suite elle a été adoptée afin de l'utiliser pour la fusion des capteurs et la combinaison des classificateurs. Cette théorie est appliquée dans un environnement dit environnement de discernement où chaque élément de l'environnement peut être interprété avec plus d'une réponse possible, mais seulement une et une seule réponse doit être acceptée. Chaque réponse est une masse d'évidence (probabilité de vraisemblance) pour un élément de l'environnement. Les masses d'évidence sont issues de différentes sources. Le principe de la théorie de combinaison est le suivant : à partir d'au moins deux masses d'évidence, une seule masse de combinaison est obtenue pour chaque élément de l'environnement. Pour appliquer la théorie de combinaison de Dempster-Shafer, il faut respecter les conditions suivantes :

— Les masses d'évidence doivent être indépendantes (provenir de différentes sources).
— Les valeurs des masses d'évidence doivent être dans l'intervalle $[0, 1]$ et $\sum m(A) = 1$ avec A appelé corps d'évidence et m est la masse d'évidence.

Pour des masses d'évidence indépendantes, cette théorie est énoncée comme suit : Soit m_1 et m_2 deux jeux de masses d'évidence provenant de deux sources différentes dans l'environnement ω. La masse jointe est définie par :

$$(m_1 \oplus m_2)(A) = \begin{cases} 0 & if A = \varnothing \\ \frac{1}{1-K} \sum_{B \cap C = A} m_1(B) m_2(C) & if A \neq \varnothing \end{cases} \tag{3.23}$$

avec :

$$K = \sum_{B \cap C = \varnothing} m_1(B) m_2(C) \tag{3.24}$$

et : A, B et C sont des éléments de l'environnement ω ou appelés aussi corps d'évidence. La masse jointe obtenue est normalisée en fonction du terme K qui est une mesure de conflit entre les masses d'évidence.

Dans notre modèle d'apparence, en premier lieu on a utilisé juste deux modèles d'apparence (modèle de couleur et modèle épars). En projetant cette théorie d'évidence dans le calcul de la fonction d'affinité afin de calculer une mesure de la similarité en fonction de différents descripteurs, l'environnement ω est formé par la liste des objets candidats, la masse d'évidence m_1 est le score de similarité pour le descripteur de modèle épars et la masse d'évidence m_2 est le score de similarité de descripteur de modèle de couleur basé sur LSH. On respecte bien les hypothèses pour appliquer la théorie de combinaison. En fait, les valeurs de similarité sont des probabilités de vraisemblance qui appartiennent à l'intervalle $[0, 1]$ en plus les valeurs de similarités sont calculées pour des descripteurs indépendants (modèle épars et modèle couleur). Pour chaque

objet cible, on doit calculer la masse jointe pour chaque élément de l'environnement (pour chaque candidat) en fonction de deux descripteurs : descripteur de modèle épars et descripteur de modèle de couleur. Un vecteur de masse jointe est ainsi obtenu pour chaque objet cible. Bien que cette théorie soit en mesure de généraliser une masse jointe globale en utilisant au moins deux masses d'évidence, la croyance que la masse jointe est correcte reste limitée. En fait, plus le conflit entre les masses d'évidence est grand, plus la valeur de masse jointe tend vers zéro. Le facteur de conflit dépend de la source d'où vient la masse d'évidence. En fait, plus les masses d'évidence sont incertaines, plus le conflit est élevé et en conséquence le résultat de la combinaison est moins certain. À noter que si une probabilité de vraisemblance d'un objet candidat par rapport un objet cible est nulle, cet objet candidat n'a aucune chance d'être mis en correspondance avec cet objet cible. Cette règle de combinaison ne peut pas être utilisée dans le cas d'un conflit élevé. En conséquence, l'idée est d'attribuer un poids pour chaque descripteur du modèle d'apparence. Les détails seront présentées dans la section suivante.

2. **Somme pondérée des descripteurs :** Puisqu'on utilise différents descripteurs afin de modéliser l'apparence d'un objet cible, chaque descripteur n'influe pas de la même façon sur la représentation de l'objet (que ce soit l'objet cible ou l'objet candidat). Par conséquent, chaque score de similarité associé à chaque descripteur doit avoir un poids qui permet de mettre en valeur son importance par rapport au modèle global. Afin d'obtenir une valeur de similarité globale entre deux objets (un objet cible et un objet candidat), les scores de similarité qui proviennent de différentes caractéristiques sont linéairement combinés selon leurs poids. Une carte de similarité globale est donc créée à chaque trame t afin de représenter la similarité en prenant en considération tous les descripteurs. Soit $X^t = \{x_1^t, x_2^t, ..., x_n^t\}$ l'ensemble des objets cibles à la trame t et $Y^t = \{y_1^t, y_2^t, ..., y_m^t\}$ l'ensemble des objets candidats à la trame t. L'ensemble de caractéristiques $S = [s_1, s_2, s_3, s_4]$ contient la liste des caractéristiques utilisées afin d'estimer le modèle global à savoir :
   — s_1 est la valeur de similarité pour la caractéristique de couleur ;
   — s_2 est la valeur de similarité pour la caractéristique de modèle éparse ;
   — s_3 est la valeur de similarité pour la caractéristique de mouvement ;
   — s_4 est la valeur de similarité pour la caractéristique spatiale ;
   La fonction d'affinité est énoncée sous la forme suivante :

$$f_t(x_i^t, y_j^t) = \sum_k \alpha_k s_k(x_i^t, y_j^t), \tag{3.25}$$

Avec α_k désignant le poids associé pour la caractéristique k et $s_k(x_i^t, y_j^t)$ est la probabilité de vraisemblance entre l'objet cible x_i^t et l'objet candidat y_j^t. Les poids associés pour chaque caractéristique sont fixés expérimentalement et ils sont pareils pour toutes les bases de données.

3.3.2 Mise en correspondance

Une fois que la fonction d'affinité est calculée en se basant sur la fusion de la similarité de différents descripteurs entre un objet cible et un objet candidat, une matrice de similarité globale est obtenue. Afin de trouver la position courante d'un objet cible, les objets cibles et les objets candidats doivent être mis en correspondance (figure 3.7). Dans la figure 3.7, les bonshommes qui sont sur le côté gauche représentent la liste des objets cibles et les bonshommes qui sont sur le côté droit représentent la liste des objets candidats. Par exemple, le bonhomme de couleur verte est mis en correspondance avec le bonhomme de couleur bleue foncée (les associations sont montrées par des traits continues dans la figure 3.7 sont à titre d'exemple). À noter que les couleurs sont utilisées pour illustrer des objets différents et ils ne reflètent pas la similarité entre eux. Pour ce faire, on doit utiliser un algorithme d'optimisation qui permet de résoudre le problème d'affectation en un temps polynomial. Dans cette thèse, on a utilisé deux algorithmes d'optimisation très connus et souvent utilisés dans la littérature : l'algorithme hongrois et l'algorithme glouton.

L'objectif de l'étape de mise en correspondance est de calculer la matrice d'assignation à partir de la matrice d'affinité (ou matrice de coût). La matrice de correspondance est une matrice qui contient seulement des 0 ou des 1 où 1 désigne une affectation faite et 0 aucune affectation faite.

— **Matrice d'affinité** La mappe de similarité globale obtenue à chaque instant t est stockée dans une matrice R_t de taille $n \times m$ (voir figure 3.8).
Chaque ligne de la matrice correspond à un objet cible et chaque colonne de la matrice correspond à un objet candidat. Chaque élément R_i^j de la matrice reflète la similarité entre l'objet cible x_i^t et l'objet candidat y_j^t défini par :

$$R_i^j = f_t(x_i^t, y_j^t) \tag{3.26}$$

où f est la fonction d'affinité définie par l'équation (2.25)

— **Algorithme glouton** L'algorithme glouton est un algorithme heuristique qui vise à chercher une solution optimale locale. Le principe est de trouver une solution locale qui peut mener à une solution optimale globale. Le but est de trouver la meilleure affectation qui maximise une fonction de coût. Initialement, une matrice de coût est

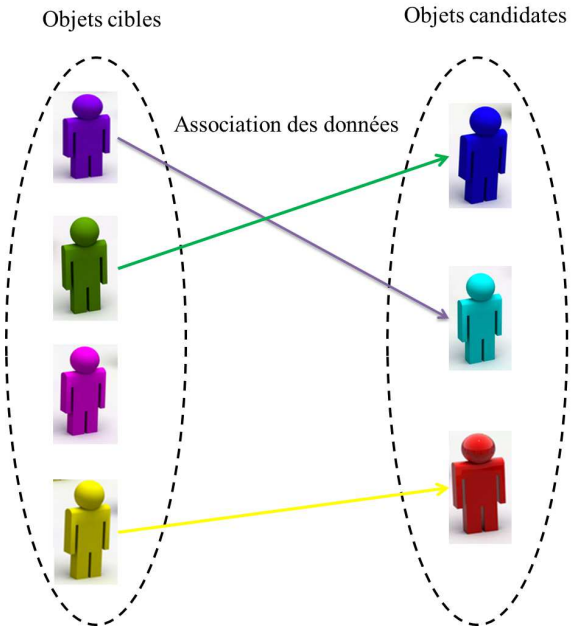


Figure 3.7 Association des données.

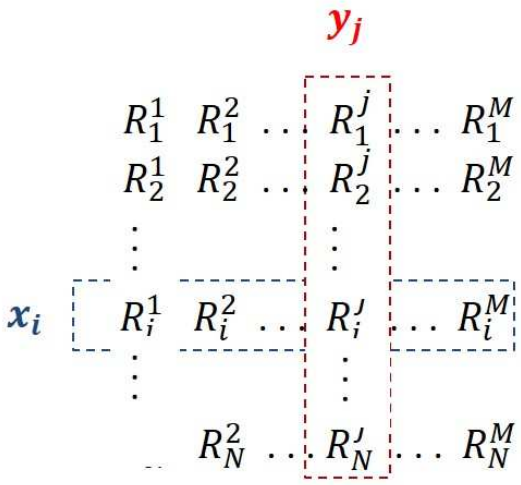


Figure 3.8 Matrice d'affinité.

calculée où chaque case reflète la valeur de la fonction d'affinité pour chaque couple (x, y). La paire pour laquelle la fonction d'affinité est la plus élevée est itérativement sélectionnée. Une fois qu'une paire est sélectionnée, la ligne et la colonne où se trouve la paire doivent être supprimées de la matrice d'affinité. Ce processus doit être répété jusqu'à ce qu'aucune paire ne soit disponible. À noter que seulement les coûts qui sont supérieures à un seuil Γ sont pris en considération. Une matrice d'affectation globale doit être obtenue à la sortie de l'algorithme d'optimisation glouton. Chaque ligne de la matrice peut contenir au plus une affectation correcte. En fait, c'est une association de type 1-1 : un objet cible doit être assigné à au plus un objet candidat et un objet candidat doit être assigné à au plus un objet cible. L'algorithme glouton est décrit dans l'algorithme 1.

Algorithm 1 Algorithme Gloton

Entrées :
X^t : ensemble de tous les objets cible à la trame t.
Y^t : ensemble de tous les objets candidats à la trame t.
R_t : matrice d'affinité à la trame t.
Γ est un seuil pour l'algorithme glouton.
Sortie :
A_t matrice finale d'affectation à la trame t.
Initialisation :
$A_t(x_i^t, y_j^t) = 0$
Contraintes :
$\forall x_i^t \in X^t : \sum_{j=1}^{m} A_t(x_i^t, y_j^t) \leqslant 1.$
$\forall y_j^t \in Y^t : \sum_{i=1}^{n} A_t(x_i^t, y_j^t) \leqslant 1.$
while $X^t \neq 0 et Y^t \neq 0$ **do**
  $(x_*^t, y_*^t) = \arg max R_t(i, j)$
  if $R_t(i^*, j^*) \geqslant \Gamma$ **then**
    $A_t(x_*^t, y_*^t) = 1.$
    $X^t = \{X^{t-1}, x_*^t, \}$
    $Y^t = \{Y^{t-1}, y_*^t, \}$
  end if
end while

— **Algorithme hongrois** L'algorithme hongrois est un algorithme d'optimisation combinatoire qui permet de résoudre le problème d'affectation en maximisant (ou minimisant) une fonction de coût. C'est une opération itérative qui permet de transformer une matrice de coût en une matrice d'affectation en minimisant une fonction de coût (ou maximiser une fonction de coût). L'algorithme hongrois est présenté dans l'algorithme 2.

Algorithm 2 Algorithme hongrois

Entrées :
X^t : ensemble de tous les objets cibles à la trame t.
Y^t : ensemble de tous les objets candidats à la trame t.
R_t : matrice d'affinité à la trame t.
Sortie : A_t matrice finale d'affectation à la trame t.
Initialisation :
$A = R$
Étape 1 :
for chaque ligne **do**
  $A_t(i) = A_t(i) - min(A_t(i))$
end for
for chaque colonne **do**
  $A_t(j) = A_t(j) - min(A_t(j))$
end for
Étape 2 :
for chaque ligne, chaque colonne **do**
  Encadrer les «0» qui sont soit unique sur ligne et colonne ou soit le plus haut (ou plus à gauche) sur une ligne (ou sur une colonne).
  Barrer tous les autres «0»
end for
Étape 3 :
(a) Marquer toutes les lignes ne contenant aucun zéro encadré.
(b) Marquer toutes les lignes ayant un zéro barré sur une ligne marquée.
(c) Marquer toutes les lignes ayant un zéro encadré sur une colonne marquée.
Répéter les étapes (b) et (c) jusqu'à ne reste aucune rangée à marquer.
Tracer toutes les lignes non marquées et toutes les colonnes marquées.
Étape 4 :
Les cases non tracées à l'étape 3 forment une nouvelle matrice $A_t^{'}$.
for chaque ligne, chaque colonne de $A_t^{'}$ **do**
  $A_t^{'}(i,j) = A_t^{'}(i,j) - min(A_t^{'})$
end for
for chaque ligne, chaque colonne de A_t **do**
  if $A_t(i,j)$ est tracé deux fois **then**
    $A_t(i,j) = A_t(i,j) + min(A_t^{'})$
  end if
end for
Répéter les étapes 1 à 3 jusqu'à avoir une matrice qui contient des zéros uniques par ligne et par colonne.

— **Mise en correspondance** Pour faire la mise en correspondance, on utilise un des algorithmes décrits ci-dessus. À chaque trame, on vise à trouver le meilleur couple d'appariement (objet cible, objet candidat). Les défis majeurs de cette étape sont :

1. Le nombre des objets cibles change au cours du temps.
2. Des occultations partielles et totales entre les objets cibles au cours du temps.
3. Objets candidats non fiables (peuvent être des fausses détections)

Ainsi, lors de la mise en correspondance, on peut trouver qu'un objet cible n'est pas assigné ou bien un objet candidat n'est pas mis en correspondance avec aucun objet cible. En conséquence, on exploite une étape de traitement supplémentaire afin de résoudre de tels problèmes de mise en correspondance. D'abord, on essaie de mettre en correspondance tous les objets (objets cibles et objets candidats) en même temps. Des conditions s'appliquent sur les affectations obtenues selon la valeur de similarité et aussi selon l'état de l'objet cible et aussi la fiabilité de l'objet candidat. Un sommaire de cet algorithme d'association des données est décrit dans l'algorithme 3 : selon le score de similarité obtenue pour chaque couple, un objet cible peut être marqué comme en état inactif ou en état actif.

Par ailleurs, dépendamment de la position d'un objet candidat par rapport à la région d'entrée/sortie (sélectionner manuellement à la première trame de la séquence vidéo), un objet candidat peut être ajouté comme étant un nouvel objet cible ou un objet cible peut être supprimé de la liste des objets cibles.

En utilisant l'algorithme glouton pour la mise en correspondance, on élimine toutes les affectations dont la valeur de similarité entre un objet cible et un objet candidat est inférieure à un seuil th. En conséquence, avec cette stratégie on peut dire qu'on a raté des possibilités d'affectation qui peuvent être correctes. En plus, si un objet cible entre à l'état inactif (en occultation avec d'autres objets dans la séquence vidéo), son modèle d'apparence change et du coup sa valeur de similarité devient plus basse à cause de cette variation dans le modèle d'apparence. De ce fait, on a proposé une stratégie hiérarchique plus sophistiquée pour l'association des données. Cette stratégie d'association de données permet d'améliorer la qualité du suivi multi objets et d'éviter l'ambiguïté lors de la mise en correspondance. Selon l'état de l'objet cible, un objet cible peut être en état actif ou bien en état inactif. Ainsi, le processus d'association est donc fait en deux sous-étapes :

1. **Niveau 1**. Seuls les objets cibles à l'état actif doivent être mis en correspondance avec tout l'ensemble des objets candidats.
2. **Niveau 2**. Seuls les objets cibles à l'état inactif doivent être mis en correspondance

Algorithm 3 Association des données de base

Entrées :
R_t : matrice d'affinité à la trame t.
th : seuil pour l'algorithme de glouton.
A_t = algorithmeGlouton (R_t, th)
for chaque objet cible x_i^t **do**
  if x_i^t n'est pas assigné **then**
    x_i^t est marqué comme un objet non actif ;
  end if
  if x_i^t est détecté dans la région d'entrée/sortie pour une durée du temps. **then**
    $X^t = \{X^{t-1}\} \setminus x_i^t$;
    $A_t(i) = 0$
  end if
end for
for pour chaque objet candidat y_i **do**
  if y_i n'a pas été assigné et il est dans la région d'entrée/sortie **then**
    $X^t = \{X^{t-1}, y_i\}$;
    $A_t(i) = 1$
  end if
end for

avec seulement l'ensemble des objets candidats qui n'ont pas été assignés au premier niveau.

L'algorithme de mise en correspondance global est détaillé dans l'algorithme 4.

3.3.3 Gestion du suivi

Certains défis du suivi multi objets ne peuvent pas être résolus seulement par un algorithme d'optimisation des assignations entre les objets cibles et les objets candidats par exemple lors d'une occulation (partielle ou totale), il est difficile de trouver la bonne assignation de l'objet cible. En fait, la tâche la plus difficile de l'association des données est lorsqu'un objet cible n'a pas été apparié ou un objet candidat n'a pas été étiqueté (voir figure 3.9).

De ce fait, on a proposé une solution post-traitement qui permet d'interpoler les trajectoires perdues et de mieux résoudre les problèmes d'occultations.

— **État d'un objet cible** : Un objet cible peut être défini par plusieurs caractéristiques dont : ses coordonnées géométriques (où se positionne l'objet pendant la dernière trame), un identifiant (afin de nommer un objet cible : personne 1, personne 2, etc.), un ensemble des caractéristiques pour présenter son modèle d'apparence (à savoir l'histogramme LSH, le modèle épars, l'histogramme HOOF et le modèle spatial). Afin

Algorithm 4 Algorithme d'association de données hiérarchique

Calculer la fonction d'affinité $f_t(x_i^t, y_j^t)$ pour tous les objets actifs ainsi que les objets candidats.
Calculer la matrice d'affinité en appliquant l'algorithme hongrois.
for all Affectation valide **do**
  if $f_t(x_i^t, y_j^t) > seuil$ **then**
    Supprimer l'affectation.
  end if
end for
Calculer la fonction d'affinité $f_t(x_i^t, y_j^t)$ pour tous les objets inactifs ainsi que les objets candidats non mis en correspondance à l'étape précédente.
Calculer la nouvelle matrice d'affinité en appliquant l'algorithme hongrois.
for all Affectation valide **do**
  if $f_t(x_i^t, y_j^t) > seuil$ **then**
    Supprimer l'affectation.
  end if
end for
for all objets cibles **do**
  for all objets candidats **do**
    if objet cible actif n'a pas été affecté **then**
      Objet cible sera défini comme inactif (caché)
    end if
    if objet cible caché est affecté **then**
      Objet cible sera défini comme actif
    end if
    if objet candidat n'a pas été affecté et qu'il n'est pas dans la région d'entrée/sortie **then**
      L'objet candidat sera défini comme hypothèse.
    end if
    if objet candidat n'a pas été affecté et il est dans la région d'entrée/sortie **then**
      L'objet candidat sera défini comme nouvel objet cible.
    end if
    if objet cible n'a pas été affecté et il est dans la région d'entrée/sortie pour plus que $\varphi 1$ trames **then**
      L'objet cible sera défini comme un objet cible sortant.
    end if
    if objet cible est à l'état hypothèse et il est affecté pour plus que $\varphi 2$ trames **then**
      L'objet cible sera défini comme un objet cible actif.
    else
      L'objet cible sera supprimé de la liste des trajectoires.
    end if
  end for
end for

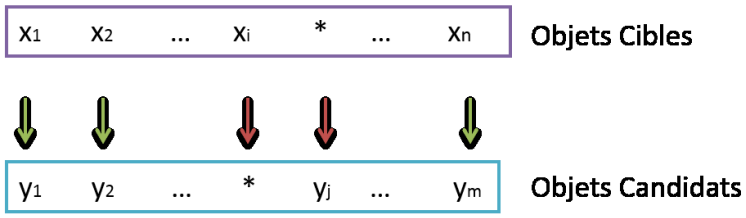


Figure 3.9 Appariement des objets cibles.

de remédier aux problèmes du suivi multi objets, une autre propriété sera ajoutée à la liste des caractéristiques d'un objet : son état (voir graphe d'état à la figure 3.10). L'état d'un objet est utilisé afin de différencier les objets qui sont visibles ou actifs, et aussi les nouveaux objets cibles ou les objets qui quittent la scène. Au total, il y a cinq états différents :

1. *Actif*. Un objet cible est en état actif s'il est détecté (il n'est pas occulté par d'autres objets).
2. *Occulté* . Un objet cible est en état occulté si il n'est pas mis en correspondance avec aucun détection. En fait, sa position a été perdue à cause d'une occultation (totale ou partielle) ou bien à cause de non-fiabilité des objets candidats (fausses alarmes).
3. *Sortant*. Un objet cible est en état sortant s'il quitte la séquence vidéo ou s'il est temporairement hors du champ de vue de la caméra.
4. *Entrant*. Un objet candidat est en état entrant s'il est apparu récemment dans la séquence vidéo (dans la région d'entrée/sortie). Un objet entrant est un nouvel objet cible qui sera ajouté à la liste des objets cibles.
5. *Hypothèse*. Un objet candidat est en état hypothèse s'il n'est pas affecté à un objet cible et il n'est pas dans la région d'entrée/sortie (au milieu de la scène). Un objet hypothèse peut être un nouvel objet cible qui apparaît, une fausse alarme ou bien un objet cible qui existe déjà, mais qui a quitté la scène temporairement.

L'entrée et la sortie des objets cibles sont déterminées en se basant sur une région de la trame qu'on appelle : région d'entrée/sortie. La région d'entrée/sortie est sélectionnée manuellement pour chaque séquence vidéo à la première trame (voir la zone hachurée en rouge de la figure 3.11). Il faut noter que cette zone est utilisée seulement afin de définir si un objet détecté est considéré comme un nouvel objet ou non. C'est une région qui encadre les bordures de la trame. En effet, on suppose que les objets en mouvements lors de l'entrée ou de sortie traversent nécessairement les bordures de la

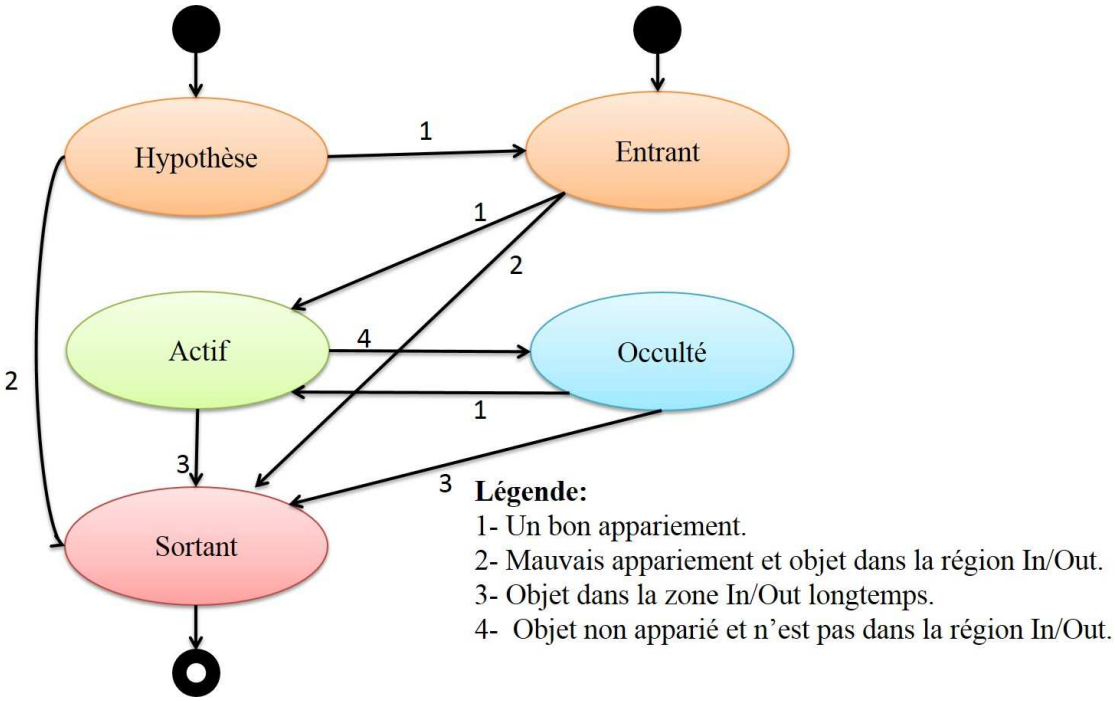


Figure 3.10 Graphe d'état d'un objet cible.

Figure 3.11 Région d'entrée/sortie.

trame (les bordures d'une trame définissent la limite du champ de vue de la camera). En conséquence le nombre d'objets cibles varie souvent au cours de temps à cause de ce qu'on appelle le processus de la naissance (un nouvel objet est ajouté à la liste des objets cibles) et la mort (un objet qui existe déjà est supprimé de la liste des objets cibles) des objets. En outre, afin de gérer les cas d'occultation entre les objets, un objet cible peut être en état occulté ou en état actif. Afin de définir l'état courant de l'objet cible, on a plusieurs conditions à vérifier :

1. Si un objet candidat a été détecté à l'intérieur de la région d'entrée/sortie alors cet objet doit être ajouté à l'ensemble courant des objets cibles comme étant une nouvelle entrée dont son état est entrant.
2. Si un objet cible reste à l'intérieur de la région d'entrée/sortie plus qu'un nombre prédéfini de trames alors cet objet doit être supprimé de l'ensemble courant de l'objet cible et il est marqué comme en état sortant.
3. Si un objet cible n'a pas été assigné alors on suppose que cet objet est en état d'occultation : état occulté.
4. Si un objet cible est en état occulté et qu'il est affecté à un objet candidat avec un score de similarité élevé alors cet objet sera remis à l'état actif.
5. Si un objet candidat n'a pas été affecté, il peut être marqué comme étant une hypothèse, car il faut suivre sa trajectoire dans les prochaines trames pour définir son état final (soit actif soit sortant) : état hypothèse.

— **Interpolation des trajectoires perdues** : L'étape de l'association des données est la mise en correspondance qui se fait entre la liste des objets cibles et la liste des objets candidats. Cela signifie que si dans une trame donnée l'objet cible n'a pas été détecté, par conséquent, l'objet cible ne sera pas apparié et du coup il sera mis comme objet occulté. On appelle les trous d'une trajectoire les positions où l'objet cible n'a pas été localisé (un exemple est donné dans la figure 3.12). Dans la figure 3.12, les trois trajectoires présentent un trou du suivi (car les objets cibles associés n'ont pas été mis en correspondance avec aucun candidat).
Afin de remplir les trous des trajectoires durant l'occultation, on propose une étape de post-traitement qui permet de remplir les trous par prédiction des objets cibles. Cette étape est appelée une étape d'interpolation de trajectoires perdues. L'interpolation des trous de trajectoires est obtenue en se basant sur l'historique du mouvement de l'objet cible entre deux états : objet cible en état occulté et objet cible en état actif. L'interpolation est basée sur l'hypothèse suivante : les objets cibles dans la séquence vidéo font des mouvements de type linéaire constant (les objets cibles sont en général

Figure 3.12 Interpolation des objets cibles.

des piétons). Tout d'abord, un vecteur de mouvement pour l'objet cible doit être estimé en utilisant les positions où l'objet a déjà été observé à travers le temps. Chaque élément de ce vecteur contient la différence entre deux positions successives de l'objet cible. Mathématiquement, on suppose un objet cible donné x_i^t dans la trame t. L'objet cible x_i^t était en état cocculté depuis la trame t_{occ} et il est remis à l'état active à la trame courante t_{cur}. Le vecteur de mouvement à un temps t_j est donc :

$$\vec{dep}(t_j, t_k) = \left|(\vec{v}(t_j) - \vec{v}(t_k))/(t_j - t_k)\right|, \tag{3.27}$$

Avec $\vec{v}(t)$ est le vecteur qui contient les coordonnées géométriques (abscisse et ordonnée) de la position de l'objet cible à la trame t. Et $t_j, t_k \in [t_{seuil}, t_{occ} - 1]$ sont deux temps successifs où la position de l'objet cible est suivie et t_{seuil} est le numéro de la trame à partir de laquelle on peut commencer à calculer l'historique de déplacement (*seuil* est un seuil choisi aléatoirement). La position perdue (durant la période d'occultation) est calculée à partir de la valeur moyenne de déplacement de l'objet cible. La position interpolée de l'objet cible x_i^t à la trame t est donc :

$$pos_t(x_i) = pos_{t-1}(x_i) + \mu_{dep} \tag{3.28}$$

Avec : μ_{dep} est la valeur moyenne du vecteur de déplacement $\vec{dep}$.

3.3.4 Mise à jour du modèle d'apparence

Le modèle d'apparence de l'objet cible change durant le temps à cause de plusieurs facteurs à savoir : l'occultation, le changement d'échelle, le changement de l'orientation (à cause d'une rotation), la variation d'illumination, etc. Ceci rend l'étape de la mise à jour du modèle d'apparence une étape nécessaire dans le cadre du suivi multi objets. La mise à jour du modèle d'apparence d'un objet cible est faite seulement si :

— L'objet cible n'est pas en occultation. En fait, si l'objet est en occultation, son apparence contient des pixels de l'objet occultant et seulement une partie de l'objet cible

sera modélisé.

— Un bon suivi est réalisé. À une trame t, on peut dire qu'on a obtenu un bon suivi si et seulement si le score de la similarité entre l'objet cible et l'objet candidat sélectionné est supérieur à un seuil τ_{maj}.

La mise à jour peut se faire pour n'importe quel objet cible (pour tous les objets cibles, pour seulement un objet cible ou pour aucun objet cible). Pour la mise à jour du modèle d'apparence d'un objet cible, toutes les propriétés associées doivent être mises à jour à savoir : les coordonnées spatiales, l'histogramme de couleur LSH, l'histogramme de mouvement HOOF et le modèle épars de l'objet cible. Les nouvelles caractéristiques seront calculées selon la nouvelle prédiction de l'objet cible. Ainsi, les caractéristiques des objets cibles seront remplacées par les nouvelles. Seulement les objets en état actif seront mis à jour. En fait, si un objet est en état occulté, son modèle d'apparence est inconnu durant la période d'occultation. En outre, la liste courante des objets cibles est mise à jour aussi après chaque trame en ajoutant des nouveaux objets cibles ou en supprimant des objets cibles existants.

CHAPITRE 4 RÉSULTATS

Dans cette section, nous allons présenter et évaluer les résultats de notre approche de suivi multi objet.

Un algorithme de suivi multi objets est dit idéal s'il est capable de :

— Estimer la position de chaque objet cible d'une façon précise.

— Garder la même identité pour un objet cible au cours du temps.

En conséquence, un algorithme de suivi doit être évalué en fonction de ces deux propriétés.

4.1 Méthode expérimentale

4.1.1 Séquence vidéo

Afin de démontrer la généralité de notre algorithme de suivi, ce dernier est validé sur une variété de séquences vidéo publiques : TUD Campus, TUD Crossing, TUD Stadtmitte et PETS2009 S2-L1 (Milan et al. (2013)). Le choix de ces séquences vidéo est justifié par le fait que chaque séquence vidéo a ses propres défis. Ces séquences vidéo montrent des piétons dans des environnements extérieurs. Ces piétons ont des caractéristiques communes telles que : la forme, les vêtements, l'allure du mouvement, la taille, etc. Toutes ces caractéristiques présentent des défis majeurs pour le suivi multi objet.

De plus, puisque les séquences vidéo sont captées dans des environnements extérieurs, les conditions d'illumination ne sont pas contrôlées. Par ailleurs, due aux changements du champ de vision de la caméra, la taille des piétons varie souvent. En fait, les piétons deviennent très petits quand ils sont loin de la caméra et deviennent grands quand ils sont près de la caméra. Donc, il faut prendre en considération le changement de l'échelle lors du suivi. Les piétons qui apparaissent dans les séquences vidéo sont souvent en occultation entre eux (occultation partielle ou occultation totale) ou en occultation avec des objets statiques (des objets appartenant à l'arrière-plan). Ces occultations provoquent des interruptions dans les trajectoires des objets à suivre. En outre, le nombres des piétons dans les séquences vidéo changent souvent (plusieurs piétons entrent et/ou quittent la scène). Les objets candidats (résultats du détecteur d'objets) sont fournis avec la base de données des séquences vidéo. Plus des détails sur les séquences vidéo sont donnés dans le tableau 4.1.

Tableau 4.1 Séquences vidéo

Séquences vidéo	# Trames	# Personnes	Résolution
TUD-CAMPUS	71	6	640x480
TUD-CROSSING	201	8	640x480
TUD-STADTMITTE	179	8	640x480
PETS2009-S2-L1	795	10	768x576

4.1.2 Implémentation et paramètres

Pour toutes les expériences, on utilise les détections et les (vérités de terrain) fournies avec les séquences vidéo. Pour les objets candidats (les détections fournies), on applique un processus de filtrage qui permet d'éliminer les fausses détections. Une fausse détection est une détection où la taille (la largeur et la longueur de la région qui l'englobe) est inappropriée par rapport au type de l'objet cible (par exemple la taille et le ratio d'aspect d'une voiture ne sont pas les mêmes que ceux d'un piéton), ou bien qui ne contient pas d'objet d'intérêt (la valeur de confiance du classificateur est en dessous d'un seuil choisi). Dans la figure 4.1, on montre quelques exemples des fausses détections qui doivent être supprimées. Dans l'exemple, on peut trouver des détections dont les tailles sont plus grandes ou plus petites que la taille normale (les fausses détections sont pointées par des flèches rouges).

Figure 4.1 Exemples de fausse détection.

Tous les paramètres utilisés dans l'algorithme sont fixés expérimentalement. Pour l'histogramme de couleur LSH, on a choisi les valeurs suivantes $bin = 32$. Pour la représentation éparse, le nombre des gabarits principaux est fixé à la valeur 10 pour toutes les séquences vidéo. Pour l'histogramme de mouvement, le nombre de classes dans l'histogramme est $nbBin = 4$. Les seuils utilisés pour l'association des données sont ajustés pour chaque séquence vidéo (en fonction de la taille des objets cibles et leurs positions par rapport à la région d'entrée/sortie). Finalement, la région d'entrée/sortie est sélectionnée manuellement pour chaque séquence vi-

déo.

4.1.3 Métriques d'évaluation

L'évaluation des performances d'un algorithme de suivi n'est pas une tâche triviale. En fait, plusieurs facteurs peuvent limiter la validité des résultats, à savoir : les métriques de performance choisies, la qualité des détections utilisées, le choix des seuils d'évaluation. Par ailleurs, afin de comparer un algorithme de suivi avec d'autres approches de la littérature, il faut utiliser les mêmes métriques, les mêmes détections et les mêmes paramètres de mesure. Dans le cadre de cette thèse, on a choisi d'utiliser les métriques appelées CLEAR MOT (Keni and Rainer (2008)). Ces métriques sont les plus utilisées pour évaluer un algorithme de suivi multi objet. CLEAR MOT comprend :

— **MOTP** : une mesure qui indique la précision des résultats de suivi par rapport à la vérité de terrain. Autrement dit, à quel point la position estimée de l'objet cible est exacte. Mathématiquement, on écrit :

$$MOTP = \frac{\sum_{i=1}^{matches} \sum_{t=1}^{trames} overlap(g_i^t, x_i^t)}{\sum_{t=1}^{trames} matches_t} \tag{4.1}$$

avec : *overlap* est le recouvrement entre la position prédite de l'objet cible x_i et la position de la vérité de terrain de l'objet cible g_i. Il est calculé en fonction du rapport de l'intersection sur l'union des objets suivis. Et : $matches_t$ est le nombre d'appariements dans la trame t, *trames* est le nombre de trames dans la séquence vidéo.

— **MOTA** : une mesure qui reflète le nombre d'erreurs dans la prédiction des objets cibles. Cette mesure est calculée en fonction des trois mesures suivantes : le taux de faux négatif, le taux de faux positifs et le nombre de changements d'identifiant. Mathématiquement, MOTA est égale :

$$MOTA = 1 - \frac{\sum_t (fn_t + fp_t + ids_t)}{\sum_t g_t} \tag{4.2}$$

où : fn_t est le nombre d'objets de la vérité de terrain qui ne sont pas prédits (ou manquants) dans la trame t. fp_t le nombre des prédictions qui ne sont pas appariées avec les objets de la vérité de terrain dans la trame t. ids_t est le nombre de fois où le même objet cible change d'identité à la trame t. g_t est le nombre total des objets dans la vérité de terrain à la trame t.

Plus les valeurs de $MOTA$ et de $MOTP$ sont élevées, meilleure est la performance du système de suivi. Outre les métriques CLEAR MOT, on a utilisé deux autres métriques très connues :

— **Précision** est une métrique qui mesure le pourcentage du nombre d'objets cibles

appariés par rapport au nombre total d'objets cibles dans le résultat de suivi. La précision est calculée par :

$$Precision = \frac{TP}{TP + FP} \tag{4.3}$$

— **Rappel** est une métrique qui mesure le pourcentage du nombre des objets cibles appariés par rapport au nombre total d'objets cibles dans la vérité de terrain.

$$Rappel = \frac{TP}{TP + FN} \tag{4.4}$$

avec : TP (vrai positif) est le nombre des objets correctement suivis. FP (faux positifs) est le nombre des objets non cibles suivies. FN (Faux négatifs) est le nombre des objets incorrectement suivis.

4.1.4 Temps d'exécution

L'algorithme proposé a été implémenté en utilisant le logiciel Matlab sur un PC Intel Core i7 PC à 3 GHz et d'une mémoire de 16 Go. Aucune étape d'optimisation n'a pas été effectué pour le code. Le temps d'exécution dépend de deux grands facteurs : le nombre et la taille des objets cibles. Une comparaison de temps d'exécution de notre approche par rapport aux autres méthodes de l'état de l'art est illustrée dans le tableau 4.2. Notez que les résultats donnés dans le tableau 4.2 représentent le temps d'exécution moyen pour les différentes bases de données. Pour la séquence vidéo où le nombre d'objets est limité comme TUD-Campus, le temps d'exécution de notre approche est d'environ $5,5$ (sec/trame). En fait, les objets cibles sont près de la caméra de sorte que nous avons des détections avec de grand taille. Pour la séquence vidéo PETS2009 - S2L1, le temps d'exécution est d'environ $7,45$ (sec/trame). La partie la plus coûteuse de notre approche est la construction du modèle d'apparence, en particulier l'histogramme de couleur LSH.

4.1.5 Les algorithmes de comparaison MOT

Notre approche de MOT est évaluée en utilisant les paramètres décrits ci-dessus et aussi par comparaison avec d'autres approches récentes de MOT. On classifie les approches de MOT

Tableau 4.2 Comparaison de temps d'exécution .

Méthode	Proposée	[Breintenstein, 2011]	[Milan, 2014]	[Yoon, 2015]	[Poiesi, 2013]	[Kuo, 2010]
Temps(s/f)	6.47	0.5	1	0.2	3	0.25

utilisées pour la comparaison en trois catégories :

— Des approches qui utilisent un traqueur afin d'améliorer les réponses de détections (exemple : Breitenstein et al. (2011) et Milan et al. (2013)).

— Des approches qui utilisent des méthodes avancées d'association des données (exemple : Andriyenko and Schindler (2011), Segal and Reid (2013b) et Berclaz et al. (2006)).

— Des approches qui améliorent la construction du modèle d'apparence des objets cibles (exemple : Yang et al. (2009b) et Führ and Jung (2014)).

Les résultats, lorsqu'ils sont disponibles, sont obtenus à partir des articles.

4.2 Résultats expérimentaux

4.2.1 Résultats globaux

Les résultats sont présentés dans le tableau 4.3.

En général, pour tous les indicateurs de performance, notre approche proposée surpasse les autres algorithmes de suivi multi objets en réalisant jusqu'a 84% d'exactitude (MOTA). Notre valeur de métrique MOTA est souvent plus élevée que dans les résultats des travaux antérieurs. Pour les séquences vidéo PETS2009-S2L1, TUD-Campus et TUD-Crossing, notre algorithme surpasse le suivi par la méthode de détection (tracking by detection) de Breitenstein et al. (2011). Cette méthode utilise les résultats de l'algorithme de suivi *filtre des particules* combinées avec ceux qui proviennent du détecteur d'objets *HOG*. Ceci montre que l'utilisation d'un modèle d'apparence robuste permet d'atteindre des résultats meilleurs que l'utilisation d'un traqueur combiné avec un détecteur d'objets. En outre, pour les séquences vidéo TUD-Stadtmitte et PETS2009-S2L1, on obtient des résultats meilleurs que la méthode de suivi développée par Segal et al (Segal and Reid (2013b)) qui utilise une technique avancée d'association des données. On peut observer aussi que la valeur de la précision MOTA obtenue pour la séquence vidéo PETS2009-S2L1 est plus élevée que celle obtenue par l'approche de Gustavo et al. (Führ and Jung (2014)) avec une différence de 14%. Donc, malgré que dans cette méthode les objets cibles soient représentés par un ensemble des particules *patchs*, notre modèle d'apparence globale montre qu'il est plus robuste qu'un modèle par particules. Malgré que la méthode de suivi développée par Yang et al (Yang et al. (2009b)) utilise une approche basée sur la soustraction d'arrière-plan afin de gérer les cas d'occultation entre les objets cibles, notre approche obtient des résultats plus performant (84% vs 76% pour la valeur de MOTA).

Les résultats présentés dans le tableau 4.3 démontrent le fait que l'utilisation d'un modèle d'apparence robuste avec une technique simple de détection ou de l'association de données

Tableau 4.3 Comparaison des performances pour les séquences vidéo TUD et PETS2009. Les meilleures performances sont en **rouge** et les deuxièmes meilleures en *bleu*

Séquences Vidéo	Méthode	MOTA	MOTP	FN	FP	IDS
TUD-CAMPUS	Proposée	**78,18%**	*69%*	0%	13%	0
	[Riahi, 2014]	72%	**74%**	25 %	2%	1
	[Breitenstein, 2011]	*73%*	67%	26%	0.1%	2
TUD-CROSSING	Proposée	*78%*	66%	1%	8%	7
	[Riahi, 2014]	72%	**76%**	26%	1%	7
	[Breitenstein, 2011]	**84%**	*71%*	14%	1%	2
	[Andriyenko, 2011]	63%	75,5%	—	—	—
	[Pirsiavash, 2011]	63,3%	76,3%	—	—	—
	[Tang, 2014]	70,7%	77,1%	—	—	—
	[Segal, 2013]	74%	76%	—	—	—
TUD-STADTMITTE	Proposée	*67%*	57,26%	26%	6%	22
	[Andriyenko, 2011]	60,5%	*66%*	—	-	7
	[Milan, 2013]	56,2%	62%	—	-	15
	[Segal, 2013]	63%	**73%**	—	-	-
	[Milan, 2014]	**71%**	65,5%	—	-	4
	[Andriyenko, 2012]	61,8%	63,2%	—	-	4
PETS2009-S2-L1	Proposée	*84%*	66%	13%	2%	35
	[Yang, 2009]	76%	54%	—	—	—
	[Breitenstein, 2011]	80%	56%	—	—	—
	[Andriyenko, 2011]	80%	*76%*	—	—	15
	[Barclaz, 2006]	60%	66%	—	—	—
	[Fuhr, 2014]	70%	—	—	—	—
	[Milan, 2014]	**90%**	**80%**	—	—	11
	[Sherrah, 2013]	81,3%	74,4%	—	—	—
	[Bae, 2014]	80,34%	69,72%	—	—	3
	[Bae, 2014]	83%	69,59%	—	—	4

peut résulter en un système de suivi multi objets avec de meilleures performances. La robustesse de notre modèle d'apparence vient de l'utilisation du modèle de représentation éparse en plus des autres propriétés indépendantes des objets cibles.

4.2.2 Robustesse du modèle d'apparence

La contribution la plus importante de ce travail réside dans le fait qu'utiliser des descripteurs indépendants (afin de représenter le modèle d'apparence de l'objet cible) permet d'améliorer les performances d'un système de suivi multi objets. Afin d'évaluer la robustesse de la combinaison des descripteurs, on a évalué la performance de l'algorithme proposé pour chaque descripteur utilisé. Pour ce faire, on a évalué la performance de notre système pour deux séquences vidéo (PETS2009-S2-L1 et TUD-CROSSING) et pour toutes les combinaisons possibles des descripteurs : tous les descripteurs, descripteur de couleur, descripteur de représentation éparse, descripteur de mouvement, descripteurs de couleur et de mouvement, descripteurs de couleurs et de représentation éparse et finalement descripteurs de représentation éparse et de mouvements. Pour toutes les combinaisons, la contrainte géométrique est toujours appliquée.

Les tableaux 4.4 et 4.5 présente l'évaluation pour chaque modèle d'apparence utilisé. Pour les deux séquences vidéo utilisées pour cette évaluation, en utilisant la combinaison de tous les descripteurs (l'approche proposée), la valeur de MOTA est la plus élevée en comparant avec le reste des combinaisons. Par exemple, pour la séquence vidéo PETS2009-S2L1, on obtient une valeur de MOTA égale à 84% si on utilise tous les descripteurs pour construire le modèle d'apparence de l'objet cible tandis qu'on obtient une valeur de 62% de MOTA en utilisant seulement la combinaison de deux descripteurs de représentation éparse et de mouvement. En termes de précision, la valeur de MOTP reste à peu près la même pour toutes les combinaisons.

Tableau 4.4 Résultats d'évaluation pour chaque combinaison de descripteurs pour PETS2009-S2L1. Meilleurs résultats en ***rouge***

Descripteurs	MOTA	MOTP	FN	FP	IDS	Recall	Précision
Tous les descripteurs	***84%***	***66%***	***13%***	2%	***34***	***87%***	98%
Descripteur de couleur	76%	66%	21%	3%	34	78%	97%
Descripteur épars	45%	66%	40%	12%	130	57%	83%
Descripteur de mouvement	0%	65%	38%	46%	1178	37%	45%
Couleur et Mouvement	76%	66%	18%	5%	48	81%	94%
Couleur et modèle épars	79%	***66%***	20%	***1%***	39	80%	***99%***
Modèle éparse + Mouvement	62%	66%	17%	17%	166	79%	82%

Tableau 4.5 Résultats d'évaluation pour chaque combinaison de descripteurs pour TUD-CROSSING. Meilleurs résultats en ***rouge***.

Descripteurs	MOTA	MOTP	FN	FP	IDS	Recall	Precision
Tous les descripteurs	***78%***	66%	15%	***2%***	45	81%	***97%***
Descripteur de couleur	73%	66%	***13%***	12%	22	***85%***	88%
Descripteur éparse	43%	66%	50%	5%	24	75%	91%
Descripteur de mouvement	1%	***66%***	35%	42%	214	43 %	50 %
Couleur + Mouvement	68 %	66%	17%	12%	29	80%	87%
Couleur + Éparse	76%	66%	17%	5,98%	***11***	82%	***93%***
Éparse + Mouvement	68%	66%	23%	7%	20	75%	91%

En interprétant la performance du système de suivi multi objets en utilisant les descripteurs séparément, on peut conclure que le descripteur de couleur est le plus performant (pour les séquences vidéo utilisées) suivi par le descripteur de la représentation éparse. À titre d'exemple, la valeur de MOTA pour le descripteur de couleur est de 76% contre 45% pour le descripteur de représentation éparse. Lorsqu'on se fie uniquement sur l'utilisation de descripteur de mouvement, la performance du système MOT échoue régulièrement (pour la séquence vidéo TUD-CROSSING, la valeur de MOTA obtenue est très faible 1%) surtout dans le cas où il y a des occultations multiples (pour la séquence vidéo PETS2009-S2L1, la valeur de MOTA obtenue est presque 0%)). Ceci est justifié par le fait que les objets cibles dans les séquences vidéo utilisées ont la même allure de mouvements sauf qu'ils suivent des directions différentes. Cela restreint le rôle du descripteur de mouvement à la distinction entre les directions de mouvement des objets cibles et non pas à la similarité des objets cibles. Cependant, en combinaison avec d'autres descripteurs, le descripteur de mouvement est souvent utile pour éliminer les ambiguïtés d'affectation. Par exemple, la valeur de MOTA est 45% en utilisant uniquement le descripteur de représentation éparse tandis qu'il est de 62% si on combine le descripteur de représentation éparse avec le descripteur de mouvement. On peut constater aussi que la valeur de taux de faux négatifs obtenus dans le cas de la combinaison de tous les descripteurs pour la séquence vidéo PETS2009-S2L1 est le plus petit en comparant avec les autres descripteurs. Par contre, pour la séquence vidéo TUD-CROSSING, l'utilisation de descripteur de couleur est plus performante en termes de taux de faux négatifs que l'utilisation de tous les descripteurs combinés. Ceci est expliqué par le fait que le descripteur de couleur peut réaliser des résultats prometteurs dépendamment des difficultés de la séquence vidéo (nombre d'objets cibles, niveau de difficulté des occultations).

En interprétant chaque descripteur séparément, on remarque qu'il y a un ordre d'importance pour chaque descripteur. En effet, le descripteur de couleur est toujours plus performant que

le descripteur de représentation éparse qui est à son tour plus performant que le descripteur de mouvement. Ceci explique le fait qu'on a combiné tous les descripteurs selon leurs importances (leur poids sera choisi expérimentalement). Grâce aux poids, on donne plus d'importance pour un descripteur qu'un autre. Le descripteur de couleur a un poids plus grand que le descripteur de représentation éparse qui à son tour a un poids plus grand que le descripteur de mouvement.

4.2.3 Performance qualitative

Qualitativement, notre approche réalise des résultats prometteurs. Les figures 4.2, 4.3, 4.4 et 4.5 présentent quelques exemples de résultats pour quatre séquences vidéo : PETS2009-S2L1, TUD-Stadtmitte, TUD-Crossing, TUD-Campus. On peut constater que notre algorithme de suivi multi objets permet de gérer certains problèmes de MOT tel que les cas d'occultation à long terme et multiples.

Figure 4.2 Résultats pour PETS2009-S2L1. Première ligne : Trames 5, 30, 60 et 90, Deuxième ligne : Trames 130, 140, 150 et 160

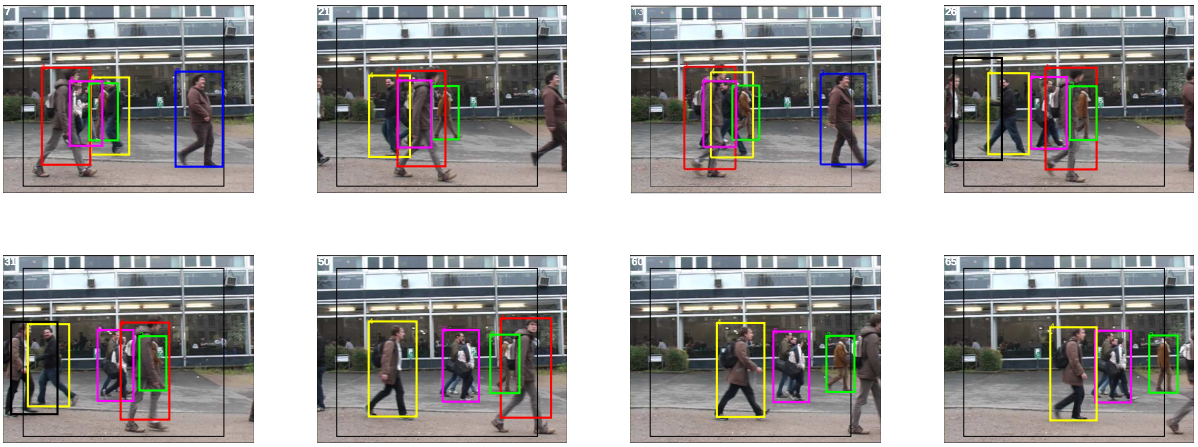


Figure 4.3 Résultats pour TUD CAMPUS. Première ligne : Trames 7, 13, 21 et 26, Deuxième ligne : Trames 31, 50, 60 et 65

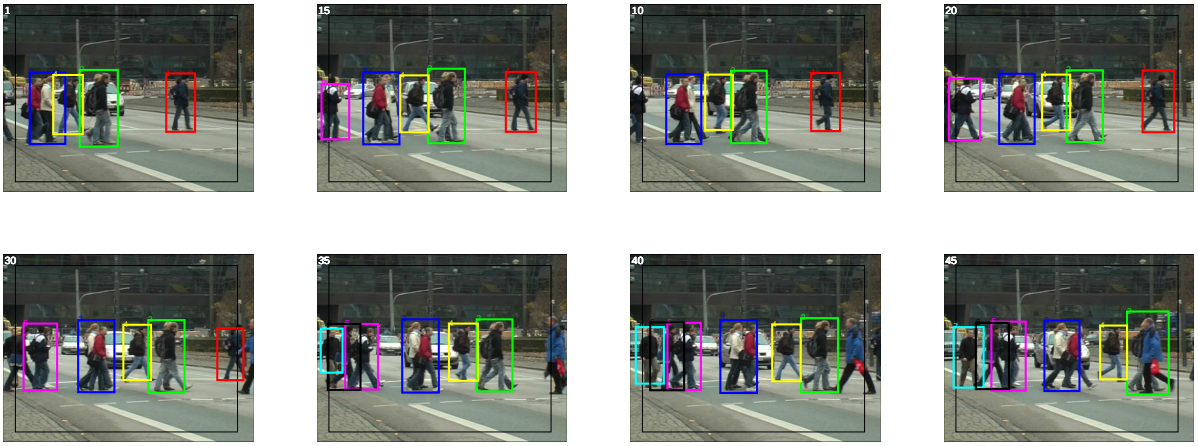


Figure 4.4 Résultats pour TUD CROSSING. Première ligne : Trames 1, 10, 15 et 20, Deuxième ligne : Trames 30, 35, 40 et 45

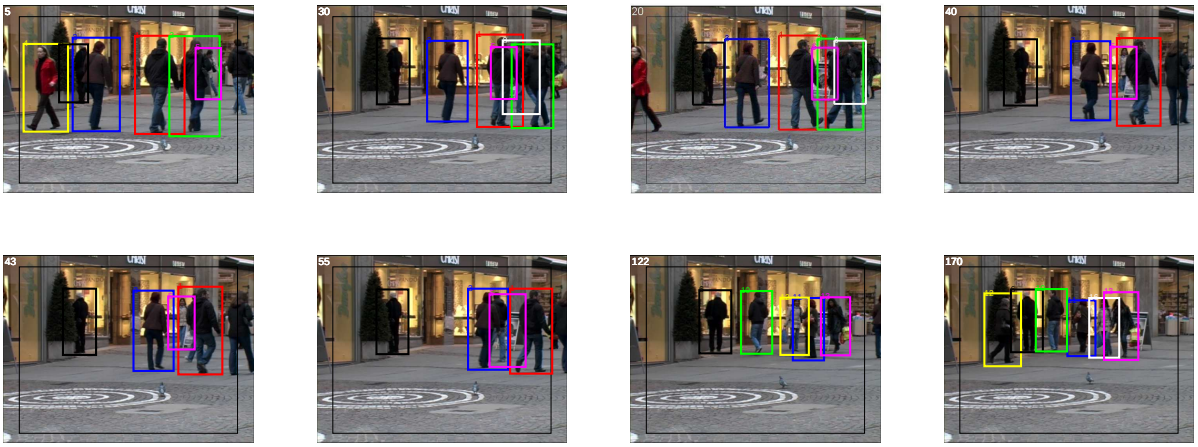


Figure 4.5 Résultats pour TUD STADMITTE. Première ligne : Trames 5, 20, 30 et 40, Deuxième ligne : Trames 43, 55, 122 et 170

Chaque séquence vidéo utilisée contient des difficultés :

— *PETS*2009-*S*2*L*1. Cette séquence vidéo contient des problèmes particulièrement difficiles. Tout d'abord, les objets cibles sont totalement occultés par le signe de circulation (voir la figure 4.2, première rangée) qui peut altérer leur modèle d'apparence. Deuxièmement, certains objets cibles arrêtent soudainement leurs mouvements pour une longue période de temps. Par exemple, la personne dont l'identifiant est 1 (voir la figure 4.2) arrête pendant plus que 100 trames. De plus, les objets font des mouvements circulaires et dans différentes directions. Ceci produit une modification significative sur l'apparence du modèle de l'objet cible puisqu'il est observé de différents points de vue (voir figure 4.6).

Figure 4.6 Variation du modèle d'apparence due aux changements de la pose

Une autre difficulté soulevée dans cette séquence vidéo, c'est que le nombre des objets cibles est grand. Ceci augmente le nombre des faux assignations lors de processus d'association des données. Par ailleurs, le point de vue de la caméra est loin par rapport à la scène ce qui se reflète sur la taille des objets en mouvements (les objets sont très petits). Un autre défi pour cette séquence vidéo est le choix de la zone d'entrée et de sortie (choisie à la première trame de la séquence vidéo). En fait, les objets cibles quittent et entrent souvent dans la scène. Notre algorithme gère avec robustesse les difficultés de cette séquence vidéo par l'utilisation d'un modèle d'apparence robuste et puissant (en utilisant un modèle d'apparence basé sur la fusion des différents descripteurs) et de notre stratégie de mise à jour pour le modèle d'apparence.

— *TUD*. Pour les trois séquences vidéo (CAMPUS, CROSSING, STADMITTE) de cette base de données, la plupart des objets cibles ont la même taille, des vêtements similaires et ont des mouvements similaires (même direction et même vitesse) et parallèles. Même si le nombre des objets cibles est limité (7 à 9 objets), il y a plusieurs occultations multiples (occultations entre plusieurs objets en même temps) et totales (un objet cible est totalement caché par d'autres objets). Aussi les occultations entre les objets sont longues (en termes de nombre de trames). Bien que les objets cibles soient clairement visibles (à cause du point de vue de la camera, la taille des objets est grande), les détection des objets ne couvrent pas tout le contour de l'objet (voir figure 4.7). Dans l'exemple de la figure 4.3, on constate que le suivi a été perdu pour la personne en brun. Ceci est causé par le fait que la personne n'a pas été détecté et elle quitte définitivement la séquence vidéo.

Figure 4.7 Exemples de résultats de détections

En outre, dans la séquence vidéo TUD-STADTMITTE, il y a une personne qui apparaît au milieu de la scène et reste fixe pour une longue période de temps (voir figure 4.8).

Figure 4.8 Un exemple d'un objet avec mouvement statique

Afin de bien gérer les défis de cette base de données TUD, notre algorithme traite les ambiguïtés lors de l'assignation des objets cibles par l'étape de post-traitement : la gestion de l'association des données (qui contient : l'interpolation des trajectoires manquantes, gestion des occultations et mettre à jour la liste des objets à suivre). En fait, les fausses assignations entre la liste des candidats et la liste des objets cibles seront détectées et supprimées dépendamment du score de similarité obtenu pour chaque couple (objet cible et objet candidat). Aussi, les objets qui traversent la région d'entrée/sortie, doivent être classés en nouveaux objets ou objets sortants (voir figure 4.9)

Figure 4.9 Exemples des objets entrants/sortants

Par la suite, on présente différents scénarios de résultats afin de montrer comment notre algorithme de suivi se comporte lors des cas difficiles.

— **Détections manquantes**. La qualité des résultats du détecteur d'objets présente un défi pour le suivi multi objets. Afin de gérer ce problème, on suit une approche d'interpolation qui permet de reconstruire les points manquants des trajectoires d'un objet

cible. L'interpolation est le fait d'estimer la position courante de l'objet cible même s'il n'est pas assigné à aucun objet candidat. Par exemple, dans la figure 4.10, l'objet cible (le rectangle englobant de couleur verte) il n'a pas été assigné en appliquant seulement l'association des données. Mais, après l'étape d'interpolation, on peut constater que la position courant de l'objet de couleur vert a été prédite avec succès.

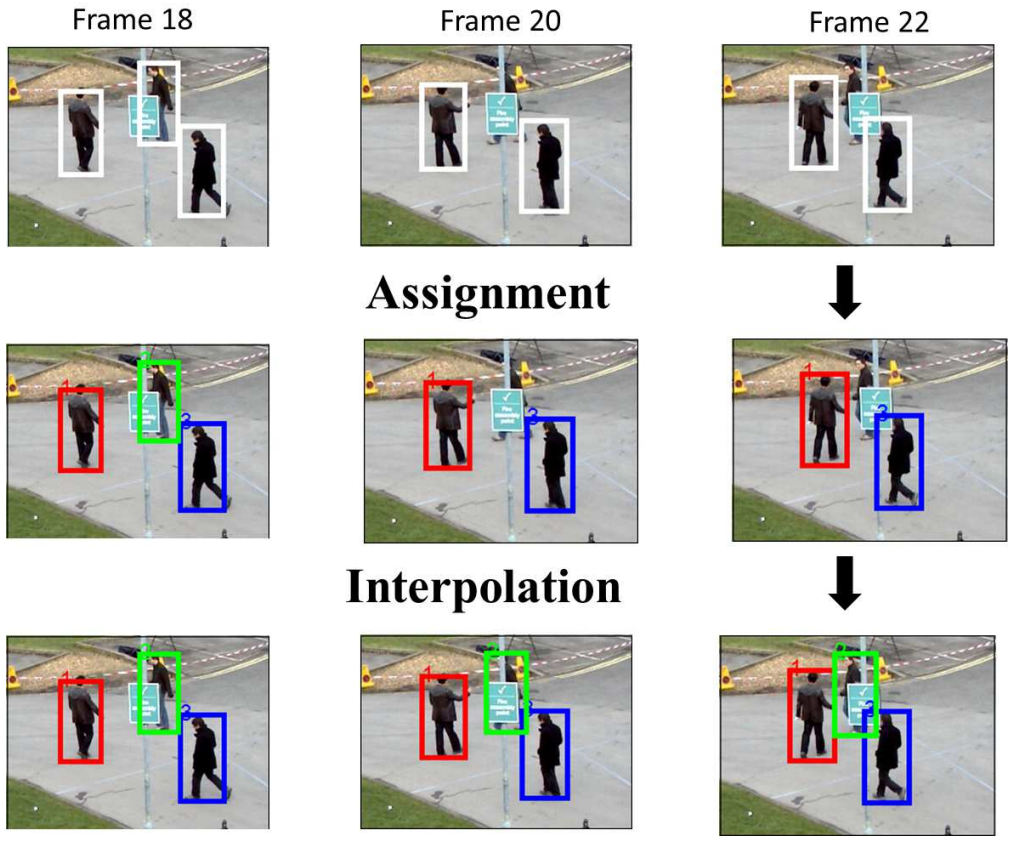


Figure 4.10 Interpolation des objets cibles

Grâce à l'étape d'interpolation, on a réussi à prédire la position de l'objet cible même s'il n'a pas été détecté ou s'il est invisible.

— **Occultations**. Suivre plusieurs objets cibles augmente les cas d'occultations entre les objets. La figure 4.11 montre un cas d'occultation multiple (plusieurs objets entrent en occultations en même temps) et d'occultation des objets cibles avec un objet statique (un panneau).

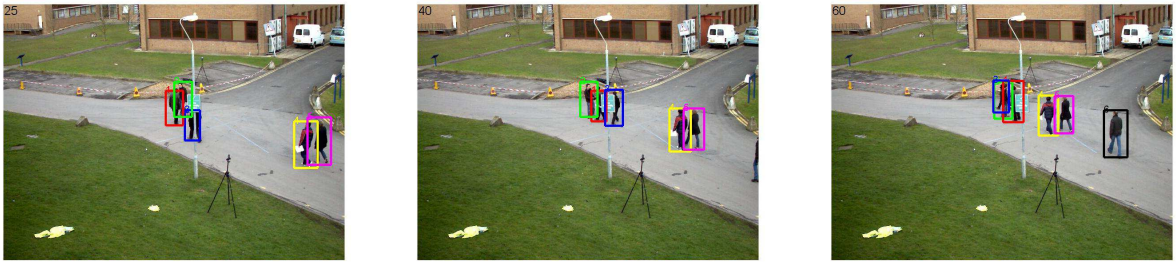


Figure 4.11 Maintient des identités des objets dans le cas d'occultations multiples. Résultats de suivi dans les trames 25, 40 et 60.

En outre, le scénario présenté dans la figure 4.12 montre un cas d'occultation longue (la durée de l'occultation dépasse 100 trames).

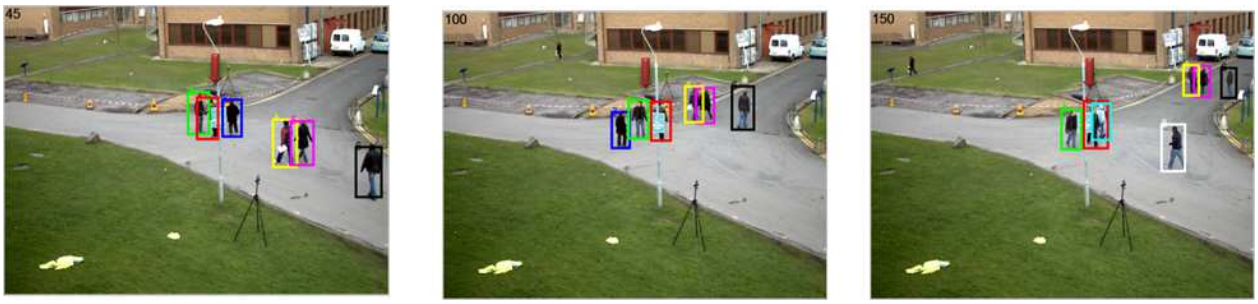


Figure 4.12 Maintient des identités des objets dans le cas d'occultation avec un objet statique. Résultats de suivi dans les trames 45, 100 et 150.

Notre approche de suivi est capable de garder les identités des objets cibles durant tous les types d'occultation.

— **Variations d'échelle**. La distance entre les objets cibles et la position de la caméra varie (à cause des mouvements des objets cibles), ce qui entraîne un changement d'échelle (la taille d'un objet cible change en s'éloignant ou en s'approchant de la camera). Alors, le modèle d'apparence varie si l'échelle change. La figure 4.13 montre un exemple de changement d'échelle pour les objets cibles de couleurs jaune et rose. L'identité de ses derniers reste inchangée pendant ce changement.

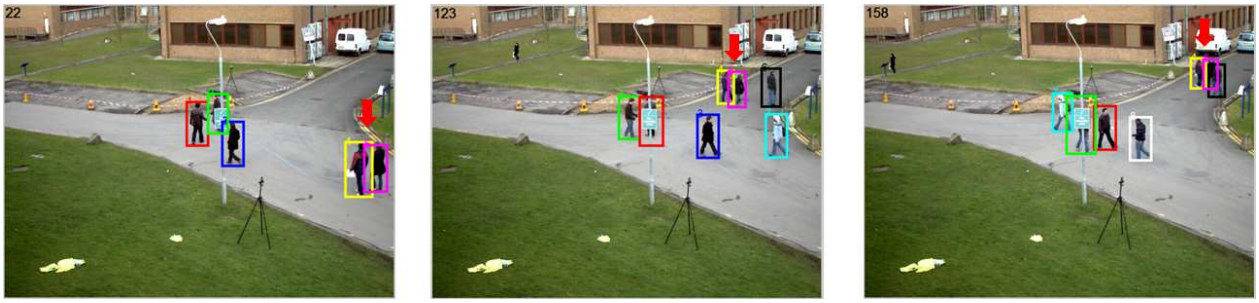


Figure 4.13 Maintient des identités des objets dans le cas de changements d'échelle. Résultats de suivi dans les trames 22, 123 et 158.

— **Variation de pose**. À cause du mouvement arbitraire des objets cibles (mouvements dans toutes les directions), la pose d'un objet cible peut changer (objet en vue de face ou vue de droite ou vue de gauche, etc.). Cette variation influe beaucoup sur le modèle d'apparence. La figure 4.14 présente la robustesse de notre modèle d'apparence aux variations de pose.

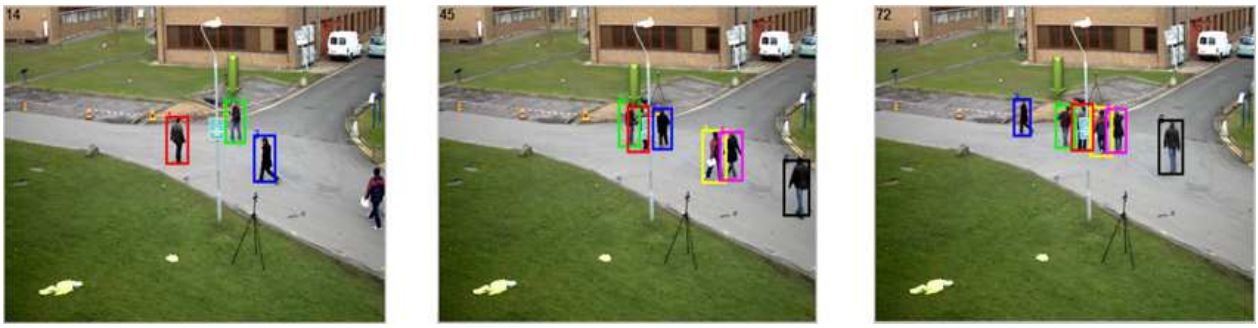


Figure 4.14 Maintient des identités des objets dans le cas de changements de pose. Résultats de suivi dans les trames 14, 45 et 72.

4.2.4 Évolution de métriques d'évaluation pour chaque trame

Dans cette section, on va étudier la variation des valeurs de métriques en fonction de temps (pour chaque trame). Les figures 4.15, 4.16 et 4.17 montrent les courbes de variations de MOTA et de la précision en fonction du numéro de trame pour les trois séquences vidéo : PETS2009-S2L1, TUD-CROSSING et TUD-STADMITTE.

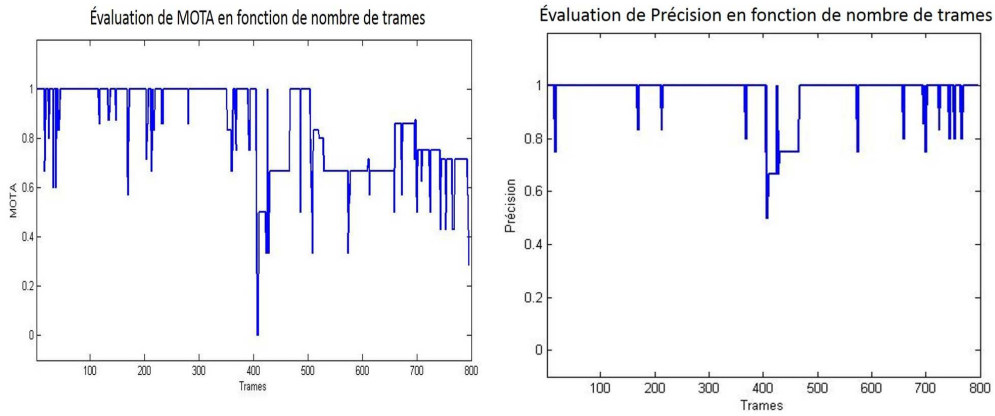


Figure 4.15 Évaluation de MOTA et de la Précision en fonction des trames pour la séquence vidéo PETS2009-S2L1

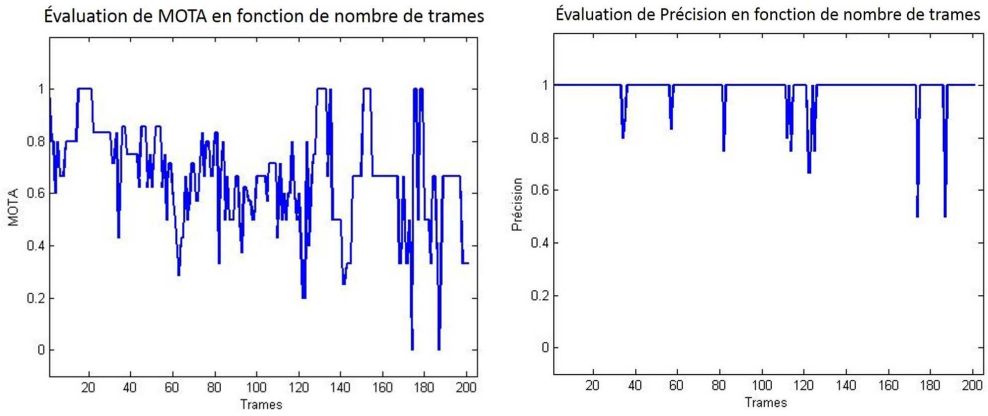
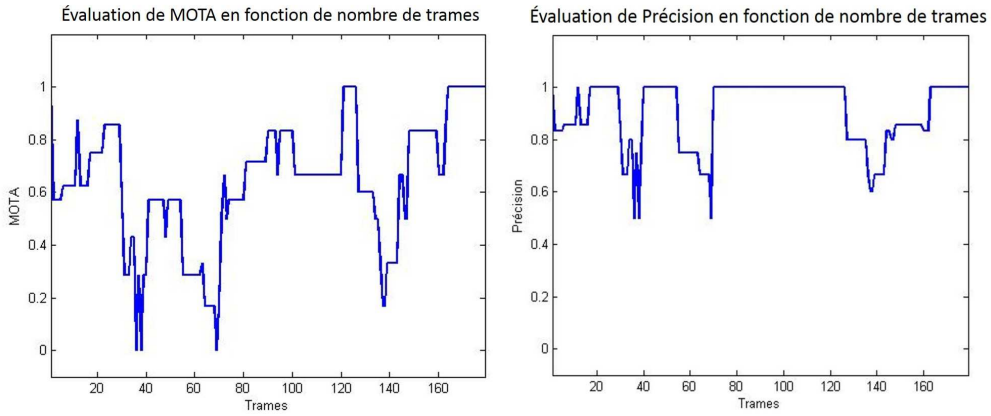


Figure 4.16 Évaluation de MOTA et de la Précision en fonction des trames pour la séquence vidéo TUD-CROSSING



(a) (b)

Figure 4.17 Évaluation de MOTA et de la Précision en fonction des trames pour la séquence vidéo TUD-STADMITTE

On peut constater que l'allure de la courbe de MOTA est presque la même pour toutes les séquences vidéo : la valeur de MOTA varie beaucoup au cours de temps. En fait, la valeur de MOTA dépend du niveau de la difficulté courante (nombre d'objets, type d'occultation, qualité des détections, etc.). À titre d'exemple, on va interpréter trois cas où la valeur de MOTA est très faible :

— **PETS**2009-**S**2**L**1 **(Trame** 407**)**. On peut constater que la trame 407 est un minimum global pour la courbe de MOTA (figure 4.15). Si on étudie ce qui se passe au niveau de cette trame, on trouve qu'il y a deux objets cibles, mais par contre seulement un objet a été détecté. Au niveau du résultat, un objet a été bien suivi, mais l'autre a été interpolé avec une légère erreur (voir figure 4.18).

(a) (b) (c)

Figure 4.18 Trame 407 (PETS2009-S2L1) : Vérités de terrain, Détections et Résultats.

— **PETS**2009-**S**2**L**1 **(Trame** 16**)**. Un autre minimum local pour la valeur de MOTA est à la trame 16 (voir figure 4.19) dans cette trame, il y a quatre objets à suivre, mais par contre dans la vérité de terrain il y a seulement trois objets. Donc, en réalité notre résultat est correct, mais en le comparant avec la vérité de terrain fourni, on aura un faux positif.

— **TUD-STADMITTE (Trame** 147**)**. Dans la trame 147, il y a deux objets à suivre (voir figure 4.20). En comparant notre résultat avec la vérité de terrain, on trouve que dans notre résultat le rectangle englobant des objets cibles est plus large (à cause des détections). Du coup, les rectangles englobants contiennent plus de pixels inutiles (pixels d'arrière-plan) ce qui cause une grande erreur de recouvrement tel qu'utilisée dans les métriques. Notre système de suivi maintient une bonne performance en termes de la précision.

(a) (b) (c)

Figure 4.19 Trame 16 (PETS2009-S2L1) : Vérités de terrain, Détections et Résultats.

Figure 4.20 Trame 174 (TUD-CROSSING) : Vérités de terrain, Détections et Résultats.

4.2.5 Sensibilité au nombre de fausses détections

Dans cette section, nous allons étudier la sensibilité de notre système de suivi multi objets par rapport à la qualité du détecteur d'objets. Premièrement, dans le tableau 4.6, on présente les performances de notre système de suivi en utilisant un détecteur d'objet parfait (0% de fausses détections). En utilisant les détections de *la vérité de terrain*, notre approche atteint une performance optimale : 100% de MOTA, 100% de MOTP, 0% de faux positifs et 0% de faux négatifs pour deux séquences vidéo. Les assignations sont donc toutes correctes. Obtenir

Tableau 4.6 Évaluation des performances en utilisant les vérités de terrain.

Séquences vidéo	MOTA	MOTP	FN	FP	IDS	Recall	Précision
TUD-CAMP	100%	100%	0%	0%	0	100%	100%
TUD-CROSS	97%	100%	3%	0%	1	97%	100%
TUD-STADM	100%	100%	0%	0%	0	100%	100%
*PETS*2009-*S*2*L*1	99,65%	97,27%	0%	0%	5	99,6%	100%

une performance presque idéale pour toutes les séquences vidéo utilisées montre le fait qu'on a réussi à concevoir un système de suivi qui est robuste aux défis majeurs connus à savoir la similarité entre les objets cibles, etc.

La question qu'on peut se poser : à quel point notre approche est sensible aux variations de pourcentage de fausses détections ? Afin de répondre à cette question, on a également étudié l'impact de pourcentage différent de fausses détections sur les valeurs de MOTA, la précision et le rappel. On crée trois types de fausses détections : fausses détections négatives (des rectangles englobants sur des régions de l'arrière-plan), fausses détections positives (des rectangles englobants sur des régions d'intérêt, mais pas sur des objets cibles) et des détections incorrectes (éliminer certains rectangles englobants autour des objets cibles). Tous les types de fausses détections sont additionnés d'une façon aléatoire avec différents pourcentages : 0%, 5%, 10%, 15%, 20%, 25% et 30%. On compare le résultat d'évaluation de performance avec les approches de base suivante :

- *Baseline* 1. On a implémenté une version de notre approche sans l'utilisation de l'interpolation des trajectoires afin de montrer l'intérêt de ce processus et l'évolution de performance du système de suivi en fonction du pourcentage de fausses détections. Pour le modèle d'apparence, on utilise le modèle d'apparence basé sur la fusion des quatre descripteurs.
- *Baseline* 2. On a implémenté une autre version de notre approche où on utilise un modèle d'apparence usuel basé sur le descripteur de couleur. Avec cette approche, on évalue l'influence de la fusion de descripteurs indépendants sur la performance du système de suivi.
- *Baseline* 3. On a implémenté une autre version de notre approche où on utilise un modèle d'apparence basé sur le descripteur de représentation éparse. Avec cette approche, on évalue l'influence de la fusion de descripteurs indépendants sur la performance de système de suivi.

Les graphiques de la figure 4.21 montrent que notre algorithme proposé est plus robuste que les approches de *Baseline*.

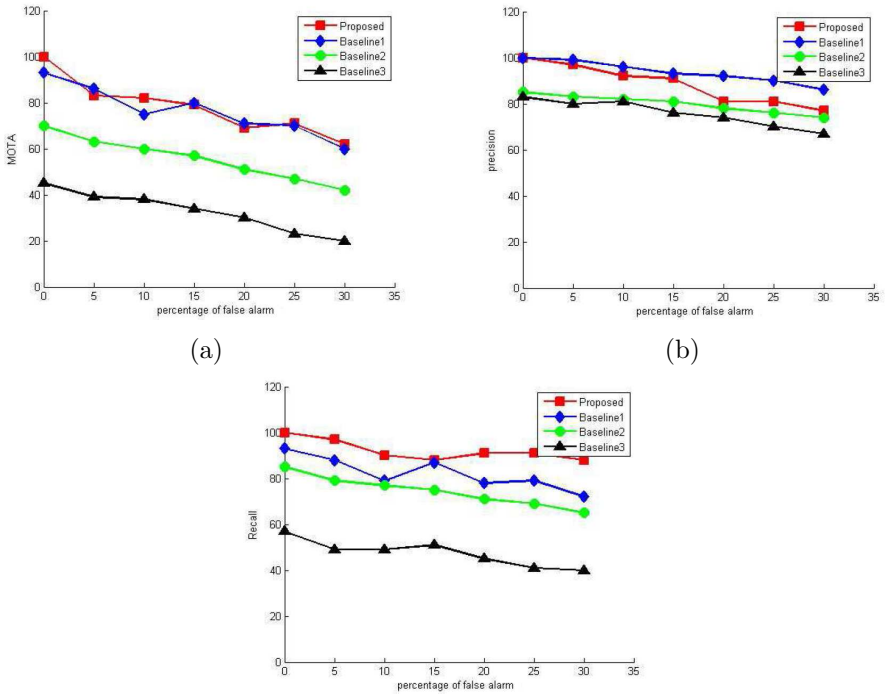


Figure 4.21 Résultats de l'évaluation de MOTA, précision et de rappel en fonction de pourcentage de fausses détections pour la séquence vidéo PETS2009

En fait, notre approche maintient la meilleure performance, lorsque le pourcentage des fausses détections augmente. En évaluant la valeur de MOTA, on obtient des résultats entre 100% et 62% avec un pourcentage de fausse détection entre 0% et 30% alors que si on utilise uniquement le descripteur de la couleur afin de construire le modèle d'apparence, la valeur de MOTA est moins de 70% et il baisse à 40% pour un pourcentage élevé de pourcentage de fausses détections (30%). En ce qui concerne *Baseline* 1 (courbe de couleur bleue), le système de suivi multi objets est plus performant que les autres *Baseline*. Avec 0% de pourcentage de fausses détections, la valeur de MOTA est de plus que 80%, la précision est de 90% et le rappel est de 80%. On peut constater qu'en utilisant des résultats de détections parfaits, l'approche sans interpolation reste aussi performante. Ceci est expliqué par le fait que l'étape d'interpolation est intégrée dans notre système de suivi multiple afin de réduire les défauts de résultats de détections. En comparant la performance de notre algorithme avec celle de *Baseline*1, on peut constater qu'il y a une légère différence pour MOTA, précision et rappel. En fait, l'importance de l'étape de l'interpolation dépend de la qualité du détecteur d'objets utilisé pour les séquences vidéo de test. Les courbes noires et vertes de la figure 4.21 (descripteur de représentation éparse et descripteur de couleur, respectivement) démontrent que le descripteur de couleur est plus discriminant que le descripteur de représentation éparse.

Ceci est justifié par le type des séquences vidéo utilisé (la couleur est plus discriminante que d'autre descripteur). En utilisant un seul descripteur pour construire le modèle d'apparence, même avec un détecteur d'objets parfait, on obtient seulement 70% de MOTA avec un descripteur de couleur et seulement 40% en utilisant un descripteur de représentation éparse. Ceci prouve la puissance de modèle d'apparence combinée utilisée dans notre approche. Pour les courbes du premier graphe (figure 4.21 a), toutes les courbes ont une allure décroissante. Cela signifie que la performance d'un système de MOT dépend de la qualité de détections des objets. On peut constater que notre approche est moins sensible aux fausses détections que les autres *Baseline*. En fait, le graphe associé à la méthode proposée est le plus haut (meilleure valeur de MOTA et rappel).

CHAPITRE 5 CONCLUSION

Avant d'énoncer les perspectives et les limitations de l'approche proposée, rappelons le contexte ainsi que les principales contributions effectuées. Dans ce travail, nous avons cherché à résoudre les problèmes d'une application de vidéosurveillance : le suivi multi objets. À cette fin, nous avons tout d'abord proposé une taxonomie des algorithmes de suivi multi objets dans le premier chapitre d'état de l'art. En fait, nous avons défini le suivi comme la combinaison des trois aspects suivants : un détecteur d'objets, un modèle d'apparence et une approche d'association des données (ou des détections).

Ce travail est positionné en considérant seulement le modèle d'apparence et l'association des données comme les aspects centraux de la solution algorithmique proposée. Sur cette base, le deuxième chapitre a présenté notre approche de suivi multi objets que nous avons validée sur des séquences vidéo publiques et évaluées en utilisant l'ensemble des paramètres d'évaluation *ClearMOT* (chapitre 3).

Nous avons présenté une nouvelle approche pour le suivi d'objets fondée sur la combinaison des descripteurs en plus d'une stratégie de gestion afin de gérer les différentes problématiques de suivi. La méthode réalisée est constituée de deux modules : un module de modèle d'apparence et un module d'association des données. Le premier a pour but de construire un modèle d'apparence afin de définir une représentation robuste pour les objets cibles. À cette fin, à partir des différents descripteurs, des caractéristiques sont extraites pour la région qui englobe l'objet cible. Ainsi, une mesure de similarité sera calculée pour chaque couple (objet candidat, objet cible) en comparant leurs représentations du modèle d'apparence. Puis, une matrice de similarité sera transmise au second module afin de créer les assignations entre les différents objets. Ces derniers seront par la suite optimisés afin de gérer les problèmes de suivi multi objets : de nouvelles trajectoires peuvent être créées, des trajectoires qui existent déjà peuvent être éliminées. Il faut noter qu'un module de détection des objets candidats est nécessaire afin d'initier le suivi. La particularité de notre approche réside dans l'utilisation d'un post-traitement qui consiste à interpoler les positions perdues des objets cibles (par exemple à cause d'occultation) afin de maintenir la continuité des trajectoires. Cette interpolation est basée sur les relations spatiales entre les différentes positions d'un objet cible. Nous rappelons succinctement nos contributions :

— Un nouvel algorithme MOT qui combine les avantages des descripteurs très répandus comme la représentation éparse et l'histogramme de couleur *Locality Sensitive Histograms*.

— L'adaptation de descripteur de représentation éparse afin de l'utiliser pour le suivi multi objets.
— Une fonction d'affinité (ou de similarité) basée sur la fusion de plusieurs descripteurs indépendants à savoir : l'histogramme de couleur, l'histogramme de mouvement, la représentation éparse et le descripteur spatial.
— Une stratégie d'association des données hiérarchique afin de gérer les occultations entre les objets cibles.
— Une interpolation en ligne des positions manquantes de l'objet cible. Un objet cible peut être suivi même s'il est n'a pas été détecté ou bien s'il est en occultation.

5.1 Perspectives

Un certain nombre de perspectives découlent de cette thèse. Malgré les résultats encourageants obtenus, il reste encore des pistes à explorer.

— **Sélection des objets candidats**. Dans ce travail, le suivi multi objets est défini par la mise en correspondance des objets candidats. Ces derniers sont obtenus en utilisant un détecteur d'objets. Sur cette base, nous avons utilisé les détections fournies avec les bases de données. Dans le but d'améliorer la qualité des détections, nous avons eu recours à un filtrage pour éliminer les détections qui ne sont pas fiables (des fausses alarmes, des objets font partie de l'arrière-plan, etc.). Selon les expérimentations effectuées, nous avons montré que la qualité des détections a une influence importante sur la performance de l'approche du suivi multi objets par détection. En fait, dans le chapitre résultats, nous avons vu que la performance de suivi atteint 100% quand la précision des détections utilisée est optimale. Il serait intéressant d'améliorer la qualité des détections afin de réduire le taux de fausses alarmes. En particulier, nous pouvons appliquer un détecteur d'objets qui donne des détections plus fiables.
— **Description des objets cibles**. Le modèle d'apparence d'un objet cible permet de décrire ce dernier en calculant ses caractéristiques visuelles. Comme déjà discuté dans l'état de l'art, le modèle d'apparence peut être construit en utilisant un ou plusieurs descripteurs. Toutefois, l'utilisation d'un seul descripteur peut être inefficace dans certaines situations (par exemple, des objets qui ont des couleurs similaires). À cet effet, nous avons proposé une combinaison de descripteurs afin de construire un modèle d'apparence plus robuste. Dans le but d'améliorer la robustesse du modèle d'apparence, nous pensons que la combinaison proposée peut s'étendre à un descripteur global basé sur le principe de la classification. En effet, les modèles d'apparence peuvent s'insérer globalement dans un cadre de classificateur à l'aide d'un processus d'apprentissage.

— **Poids des termes de la fonction d'affinité**. La sélection des poids pour chaque descripteur utilisé dans la représentation des objets cibles se fait d'une manière hors ligne et expérimentale. Nous avons proposé un paramètre constant : le vecteur de poids selon l'importance de chaque descripteur sur la représentation de l'objet. Nous pensons que l'importance des différents descripteurs n'est pas la même pour toutes les trames. De ce fait, une perspective pour ce travail est de construire un vecteur de poids en ligne par apprentissage et le mettre à jour au besoin afin de l'adapter aux différents scénarios.

— **Association des données**. L'une des difficultés que rencontrent les approches de suivi multi objets est l'occultation entre les objets cibles. En particulier, plus l'occultation est longue plus il sera difficile de récupérer la position d'un objet cible à la fin de l'occultation. Dans ce contexte, nous intégrons un processus de post-traitement qui permet de gérer les associations (fournis par un algorithme d'optimisation) selon l'état de chaque objet cible (en état d'occultation ou en état actif). En fait, l'association est faite entre une détection et un objet cible (qui définit une trajectoire existante). Par ailleurs, nous pensons que la hiérarchie d'association des données que nous proposons pour le suivi peut s'étendre au suivi par *long tracklet* où la mise en correspondance peut être effectuée entre des tracklets autrement dit entre des trajectoires locales. Du coup, une perspective pour ce travail est de gérer les cas d'occultation très longs en intégrant la notion de *long tracklet*.

— **La prédiction des objets en mouvements**. Après l'association des données, une étape de correction des trajectoires est appliquée : la prédiction des objets cibles. Elle consiste à préserver la continuité des trajectoires pour chaque objet cible en interpolant les positions inconnues de l'objet. Dans ce travail, l'interpolation est simplement obtenue par une estimation linéaire entre les positions de l'objet cible aux instants précédents. Elle est basée sur la contrainte suivante : la vélocité d'un objet est constante durant le temps. Nous imaginons qu'il serait intéressant d'utiliser une loi dynamique plus sophistiquée pour prédire les positions des objets.

5.2 Remarques finales

L'approche de MOT que nous avons proposée est une approche de suivi par les détections. Bien que notre proposition offre plusieurs avantages en améliorant les performances du suivi, la qualité de suivi dépend dans un premier temps de la qualité de détecteur d'objet utilisé. Pour surmonter cette limite, une voie intéressante serait d'évaluer d'autres détecteurs d'objets afin de maximiser la performance de notre approche. Pour finir, notre méthode de suivi

pourrait étendue à d'autres applications. En particulier, elle pourrait être appliquée à la stéréovision et à la reconstruction de la scène à partir de plusieurs capteurs tel que les caméras visibles et les caméras infrarouges. Ces derniers permettent d'intégrer des informations thermiques dans l'analyse d'une scène. Néanmoins, l'utilisation des capteurs infrarouges échoue dans certains cas (par exemple, lorsque les objets d'avant-plan ont presque les mêmes caractéristiques thermiques que l'arrière-plan). En conséquence, les capteurs visibles et infrarouges se comportent bien sous des situations différentes ce qui explique le fait que la fusion des informations provenant de capteurs différents est devenue une orientation importante dans les systèmes de suivi.

RÉFÉRENCES

A. Andriyenko et K. Schindler, “Multi-target tracking by continuous energy minimization”, dans *CVPR*. IEEE, 2011, pp. 1265–1272.

C. Bao, Y. Wu, H. Ling, et H. Ji, “Real time l1 tracker using accelerated proximal gradient approach”, dans *CVPR*. IEEE, 2012, pp. 1830–1837.

J. Berclaz, F. Fleuret, et P. Fua, “Robust people tracking with global trajectory optimization”, dans *CVPR*, vol. 1. IEEE, 2006, pp. 744–750.

——, “Multiple object tracking using flow linear programming”, dans *Performance Evaluation of Tracking and Surveillance (PETS-Winter), 2009 Twelfth IEEE International Workshop on*. IEEE, 2009, pp. 1–8.

J. Berclaz, F. Fleuret, E. Türetken, et P. Fua, “Multiple object tracking using k-shortest paths optimization”, *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 9, pp. 1806–1819, 2011.

S. T. Birchfield et S. Rangarajan, “Spatiograms versus histograms for region-based tracking”, dans *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 2. IEEE, 2005, pp. 1158–1163.

M. J. Black et A. D. Jepson, “Eigentracking : Robust matching and tracking of articulated objects using a view-based representation”, *International Journal of Computer Vision*, vol. 26, no. 1, pp. 63–84, 1998.

M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, et L. Van Gool, “Robust tracking-by-detection using a detector confidence particle filter”, dans *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 1515–1522.

——, “Online multiperson tracking-by-detection from a single, uncalibrated camera”, *Pattern Analysis and Machine Intelligence(PAMI), IEEE Transactions on*, vol. 33, no. 9, pp. 1820–1833, 2011.

R. Chaudhry, A. Ravichandran, G. Hager, et R. Vidal, “Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of hu-

man actions", dans *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 1932–1939.

D. Comaniciu, V. Ramesh, et P. Meer, "Kernel-based object tracking", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 25, no. 5, pp. 564–577, 2003.

C. Ó. Conaire, N. E. O'Connor, et A. Smeaton, "Thermo-visual feature fusion for object tracking using multiple spatiogram trackers", *Machine Vision and Applications*, vol. 19, no. 5-6, pp. 483–494, 2008.

N. Dalal et B. Triggs, "Histograms of oriented gradients for human detection", dans *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.

——, "Histograms of oriented gradients for human detection", dans *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.

L. Dihl, C. R. Jung, et J. Bins, "Robust adaptive patch-based object tracking using weighted vector median filters", dans *Graphics, Patterns and Images (Sibgrapi), 2011 24th SIBGRAPI Conference on*. IEEE, 2011, pp. 149–156.

P. F. Felzenszwalb, R. B. Girshick, D. McAllester, et D. Ramanan, "Object detection with discriminatively trained part-based models", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 9, pp. 1627–1645, 2010.

T. E. Fortmann, Y. Bar-Shalom, et M. Scheffe, "Sonar tracking of multiple targets using joint probabilistic data association", *Oceanic Engineering, IEEE Journal of*, vol. 8, no. 3, pp. 173–184, 1983.

G. Führ et C. R. Jung, "Combining patch matching and detection for robust pedestrian tracking in monocular calibrated cameras", *Pattern Recognition Letters*, vol. 39, pp. 11–20, 2014.

J. Gall et V. Lempitsky, "Class-specific hough forests for object detection", dans *Decision forests for computer vision and medical image analysis*. Springer, 2013, pp. 143–157.

V. Gouaillier, "La vidéosurveillance intelligente : promesses et défis", *Rapport technique, TechnoPole Defense and Security, CRIM*, 2009.

S. He, Q. Yang, R. W. Lau, J. Wang, et M.-H. Yang, “Visual tracking via locality sensitive histograms”, dans *CVPR*. IEEE, 2013, pp. 2427–2434.

M. Heikkila et M. Pietikainen, “A texture-based method for modeling the background and detecting moving objects”, *IEEE transactions on pattern analysis and machine intelligence*, vol. 28, no. 4, pp. 657–662, 2006.

B. K. Horn et B. G. Schunck, “Determining optical flow”, dans *1981 Technical symposium east*. International Society for Optics and Photonics, 1981, pp. 319–331.

——, “Determining optical flow”, dans *1981 Technical Symposium East*. International Society for Optics and Photonics, 1981, pp. 319–331.

C. Huang, B. Wu, et R. Nevatia, “Robust object tracking by hierarchical association of detection responses”, dans *Computer Vision–ECCV 2008*. Springer, 2008, pp. 788–801.

——, “Robust object tracking by hierarchical association of detection responses”, dans *Computer Vision–ECCV 2008*. Springer, 2008, pp. 788–801.

D. P. Huttenlocher, J. J. Noh, et W. J. Rucklidge, “Tracking non-rigid objects in complex scenes”, dans *Computer Vision, 1993. Proceedings., Fourth International Conference on*. IEEE, 1993, pp. 93–101.

B. Keni et S. Rainer, “Evaluating multiple object tracking performance : the clear mot metrics”, *EURASIP Journal on Image and Video Processing*, vol. 2008, 2008.

H. W. Kuhn, “The hungarian method for the assignment problem”, *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.

C.-H. Kuo et R. Nevatia, “How does person identity recognition help multi-person tracking ?” dans *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 1217–1224.

B. Leibe, A. Leonardis, et B. Schiele, “Robust object detection with interleaved categorization and segmentation”, *International journal of computer vision*, vol. 77, no. 1-3, pp. 259–289, 2008.

Z. Lin, L. S. Davis, D. Doermann, et D. DeMenthon, “Hierarchical part-template matching for human detection and segmentation”, dans *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. IEEE, 2007, pp. 1–8.

D. G. Lowe, “Distinctive image features from scale-invariant keypoints”, *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

E. Maggio, F. Smeraldi, et A. Cavallaro, “Combining colour and orientation for adaptive particle filter-based tracking.” dans *BMVC*, 2005.

S. J. McKenna, S. Jabri, Z. Duric, A. Rosenfeld, et H. Wechsler, “Tracking groups of people”, *Computer Vision and Image Understanding*, vol. 80, no. 1, pp. 42–56, 2000.

A. Milan, K. Schindler, et S. Roth, “Detection-and trajectory-level exclusion in multiple object tracking”, dans *CVPR*. IEEE, 2013, pp. 3682–3689.

A. Milan, S. Roth, et K. Schindler, “Continuous energy minimization for multitarget tracking”, *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 36, no. 1, pp. 58–72, 2014.

K. Nummiaro, E. Koller-Meier, et L. Van Gool, “An adaptive color-based particle filter”, *Image and vision computing*, vol. 21, no. 1, pp. 99–110, 2003.

K. Okuma, A. Taleghani, N. De Freitas, J. J. Little, et D. G. Lowe, “A boosted particle filter : Multitarget detection and tracking”, dans *Computer Vision-ECCV 2004*. Springer, 2004, pp. 28–39.

C. Papageorgiou et T. Poggio, “A trainable system for object detection”, *International Journal of Computer Vision*, vol. 38, no. 1, pp. 15–33, 2000.

H. Pirsiavash, D. Ramanan, et C. C. Fowlkes, “Globally-optimal greedy algorithms for tracking a variable number of objects”, dans *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 1201–1208.

F. Poiesi, R. Mazzon, et A. Cavallaro, “Multi-target tracking on confidence maps : An application to people tracking”, *Computer Vision and Image Understanding*, vol. 117, no. 10, pp. 1257–1272, 2013.

H. Possegger, T. Mauthner, P. Roth, et H. Bischof, “Occlusion geodesics for online multi-object tracking”, dans *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1306–1313.

——, “Occlusion geodesics for online multi-object tracking”, dans *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1306–1313.

D. B. Reid, “An algorithm for tracking multiple targets”, *Automatic Control, IEEE Transactions on*, vol. 24, no. 6, pp. 843–854, 1979.

D. Riahi, P. St-Onge, et G. Bilodeau, “Rectgauss-tex : Blockbased background subtraction”, *Dept. génie informatique et génie logiciel, École Polytechn. de Montreal, Montreal, QC, Canada, Tech. Rep. EPM-RT-2012-03*, 2012.

H. Salmane, Y. Ruichek, et L. Khoudour, “Object tracking using harris corner points based optical flow propagation and kalman filter”, dans *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*. IEEE, 2011, pp. 67–73.

A. Segal et I. Reid, “Latent data association : Bayesian model selection for multi-target tracking”, dans *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 2904–2911.

A. V. Segal et I. Reid, “Latent data association : Bayesian model selection for multi-target tracking”, dans *ICCV*. IEEE, 2013, pp. 2904–2911.

T. Sergey, J. Stefan, G. Venu *et al.*, “Review of classifier combination methods”, *Studies in Computational Intelligence : Machine Learning in Document Analysis and Recognition*, vol. 90, pp. 361–686, 2008.

E. Shechtman et M. Irani, “Matching local self-similarities across images and videos”, dans *Computer Vision and Pattern Recognition, 2007. CVPR’07. IEEE Conference on*. IEEE, 2007, pp. 1–8.

S. Stalder, H. Grabner, et L. Van Gool, “Cascaded confidence filtering for improved tracking-by-detection”, dans *Computer Vision–ECCV 2010*. Springer, 2010, pp. 369–382.

C. Stauffer et W. E. L. Grimson, “Learning patterns of activity using real-time tracking”, *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 8, pp. 747–757, 2000.

S. Tang, M. Andriluka, et B. Schiele, “Detection and tracking of occluded people”, *International Journal of Computer Vision*, vol. 110, no. 1, pp. 58–69, 2014.

S. Vijayanarasimhan et K. Grauman, “Large-scale live active learning : Training object detectors with crawled data and crowds”, *International Journal of Computer Vision*, vol. 108, no. 1-2, pp. 97–114, 2014.

B. Wang, G. Wang, K. L. Chan, et L. Wang, “Tracklet association by online target-specific metric learning and coherent dynamics estimation”, *arXiv preprint arXiv :1511.06654*, 2015.

B. Wu et R. Nevatia, “Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors”, *International Journal of Computer Vision*, vol. 75, no. 2, pp. 247–266, 2007.

X. Yan, X. Wu, I. A. Kakadiaris, et S. K. Shah, “To track or to detect ? an ensemble framework for optimal selection”, dans *Computer Vision–ECCV 2012*. Springer, 2012, pp. 594–607.

B. Yang et R. Nevatia, “Multi-target tracking by online learning a crf model of appearance and motion patterns”, *International Journal of Computer Vision*, vol. 107, no. 2, pp. 203–217, 2014.

——, “Multi-target tracking by online learning a crf model of appearance and motion patterns”, *International Journal of Computer Vision*, vol. 107, no. 2, pp. 203–217, 2014.

J. Yang, P. A. Vela, Z. Shi, et J. Teizer, “Probabilistic multiple people tracking through complex situations”, dans *11th IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, 2009.

M. Yang, F. Lv, W. Xu, et Y. Gong, “Detection driven adaptive multi-cue integration for multiple human tracking”, dans *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 1554–1561.

A. Yao, D. Uebersax, J. Gall, et L. Van Gool, “Tracking people in broadcast sports”, dans *Pattern Recognition*. Springer, 2010, pp. 151–161.

A. Yilmaz, O. Javed, et M. Shah, “Object tracking : A survey”, *Acm computing surveys (CSUR)*, vol. 38, no. 4, p. 13, 2006.

J. H. Yoon, M.-H. Yang, J. Lim, et K.-J. Yoon, “Bayesian multi-object tracking using motion context from multiple objects”, dans *Applications of Computer Vision (WACV), 2015 IEEE Winter Conference on*. IEEE, 2015, pp. 33–40.

——, “Bayesian multi-object tracking using motion context from multiple objects”, dans *Applications of Computer Vision (WACV), 2015 IEEE Winter Conference on*. IEEE, 2015, pp. 33–40.

H.-Y. Zhang, “Multiple moving objects detection and tracking based on optical flow in polar-log images”, dans *Machine Learning and Cybernetics (ICMLC), 2010 International Conference on*, vol. 3. IEEE, 2010, pp. 1577–1582.

S. Zhang, J. Wang, Z. Wang, Y. Gong, et Y. Liu, “Multi-target tracking by learning local-to-global trajectory models”, *Pattern Recognition*, vol. 48, no. 2, pp. 580–590, 2015.