Représenter pour suivre : Exploitation de représentations parcimonieuses pour le suivi multi-objets

Loïc Fagot-Bouquet

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)

Présentée et soutenue par :
Loïc Pierre FAGOT-BOUQUET
le 20/03/2017

Titre :

Représenter pour suivre : Exploitation de représentations parcimonieuses pour le suivi multi-objets

---

École doctorale et discipline ou spécialité :
EDSYS : Automatique 4200046

Unité de recherche :
UPR 8001 LAAS Laboratoire d'Analyse et d'Architecture des Systèmes

Directeur/trice(s) de Thèse :
Frédéric LERASLE, Romaric AUDIGIER (co-directeur)

Jury :
M. Vincent LEPETIT, Graz University of Technology, Rapporteur
M. Andrea CAVALLARO, Queen Mary University of London (QMUL), Rapporteur
Mme Alice CAPLIER, GIPSA-lab, Examinatrice et présidente du jury
M. Frédéric LERASLE, LAAS CNRS, Directeur de thèse
M. Romaric AUDIGIER, CEA LIST, Co-directeur de thèse
M. Yoann DHOME, CEA LIST, Encadrant CEA (invité)

Remerciements

Cette thèse a été réalisée au sein du CEA LIST au Laboratoire Vision et Ingénierie des Contenus (LVIC), en collaboration avec l'Université Paul Sabatier (UPS) et le Laboratoire d'Analyse et d'Architecture des Systèmes (LAAS). Je tiens à remercier ici l'ensemble des personnes qui m'ont accompagné tout au long de ma thèse et ont contribué à son aboutissement.

Je remercie tout d'abord M. Andrea Cavallaro et M. Vincent Lepetit pour avoir accepté de rapporter mes travaux de thèse, ainsi que Mme Alice Caplier pour avoir accepté d'examiner ma thèse et de présider le jury. Je tiens plus particulièrement à les remercier pour le temps nécessaire à l'évaluation du manuscrit et pour leurs remarques instructives.

Merci à mes encadrants de thèse pour m'avoir suivi tout au long de ces années, et notamment pour les nombreuses discussions et relectures d'articles. Je remercie plus spécifiquement mon directeur de thèse, Frédéric Lerasle, pour m'avoir accueilli au LAAS lors de mes déplacements à Toulouse et mes encadrants au CEA LIST, Romaric Audigier et Yoann Dhome, pour les nombreux conseils concernant les présentations de mes travaux de thèse.

Je remercie l'ensemble des personnes du CEA qui m'ont accompagné au cours de cette thèse, à savoir Adrien Chan-Hon-Tong, Hamid Odabai, Pierrick Paillet, Juliette Bertrand, Damien Raffard, Solène Chan-Lang, Florian Chabot, Geoffrey Vaquette, Junior Teudjio, Astrid Orcesi, Clément Abboud, Camille Dupont ainsi que l'ensemble des membres de l'équipe d'analyse de scènes et des autres équipes du LVIC. Merci aussi à Odile Caminondo et Hélène Thirion, respectivement au CEA et à l'EDSYS, pour avoir géré de manière efficace les différents aspects administratifs liés à cette thèse.

Je remercie aussi les membres de ma famille, notamment mon frère, mes soeurs et ma mère, ainsi que mes amis proches pour leur présence et leur soutien durant ma thèse. De plus, je tiens surtout à remercier mon frère et mes soeurs, Vincent, Hélène et Laure, pour m'avoir toujours soutenu lors des nombreuses difficultés survenues ces dernières années et pour m'avoir aidé lors des moments les plus difficiles.

Pour finir, merci à tous ceux que j'ai pu oublier de mentionner au cours de ces quelques lignes.

Résumé détaillé

Le suivi visuel d'objets est un sujet d'importance en Vision par Ordinateur dont les applications pratiques sont multiples et exploitées dans des domaines assez diversifiés. On peut citer en particulier les problématiques de vidéo-surveillance ou encore celles liées aux voitures autonomes pour lesquelles il est crucial d'analyser correctement l'environnement. Cette thèse se focalise sur le problème de suivi multi-objets en considérant plus spécifiquement le suivi de personnes multiples, cette catégorie d'objets étant l'une des plus fréquentes dans les applications déployées en pratique.

Le suivi multi-objets, en utilisant le paradigme de *suivi par détection*, a grandement profité des avancées récentes en détection d'objets. Néanmoins, le suivi multi-objets présente encore plusieurs problèmes spécifiques et reste ainsi une problématique difficile en Vision par Ordinateur. Les détecteurs donnent occasionnellement des réponses erronées, principalement des objets non détectés ou des fausses détections, face auxquelles un algorithme de suivi doit être le plus robuste possible.

Pour aboutir à des systèmes plus robustes, de nombreuses approches récentes cherchent à exploiter des modèles d'apparence spécifiques afin de mieux différencier les cibles. Cette même approche a été suivie pour cette thèse, en nous inspirant de méthodes de suivi mono-objet à base de *représentations parcimonieuses*. Bien que l'emploi de telles représentations se soit révélé efficace dans plusieurs domaines en Vision par Ordinateur, cet outil restait peu utilisé pour le suivi multi-objets.

La première contribution présentée dans ce manuscrit consiste à employer des représentations parcimonieuses collaboratives dans un système de suivi en ligne, image après image, pour définir les affinités en apparence entre les trajectoires estimées et les dernières détections. Des considérations sur les descriptions possibles des cibles, holistiques ou locales, ont de plus été examinées.

Les approches en ligne ne peuvent cependant remettre en cause les choix d'appariement effectués à chaque image contrairement à des méthodes considérant simultanément plusieurs images consécutives. Notre seconde contribution a alors été de proposer une méthode de suivi à fenêtre glissante, ou multi-images, permettant de corriger d'éventuelles erreurs d'appariement en exploitant des représentations parcimonieuses adaptées à ce cadre spécifique.

La dernière contribution développée dans ce manuscrit envisage l'emploi de dictionnaires denses pour définir les représentations parcimonieuses. Des dictionnaires denses, prenant en considération toutes les positions possibles dans une image, permettent de moins dépendre de la qualité du détecteur d'objets comparés à des dictionnaires définis à partir de détections.

De nombreuses évaluations quantitatives ont été réalisées sur des base de données publiques usuelles afin de permettre une comparaison avec d'autres approches récentes. Ces évaluations attestent des gains en performances des contributions proposées et valident ainsi les choix effectués.

Mots-clés : suivi multi-objets, suivi par détection, représentations parcimonieuses.

Detailed abstract

Visual object tracking is a subject of significant relevance in Computer Vision and its practical applications are numerous and exploited in various areas. For example, it is used in videosurveillance domain or by self-driving car technologies that require a full understanding of the vehicle surroundings. Multiple Object Tracking based on the *tracking-by-detection* paradigm has widely benefited from the recent developments in object detection. However, object detectors sometimes give erroneous responses, like missed detections, false positives, or imprecise detections. Maintaining target identities and handling occlusions are some other issues more specific to Multiple Object Tracking, which remains a challenging problem.

Many recent approaches have exploited complex appearance models to distinguish more efficiently the targets and gain in robustness. In this thesis, we have followed the same idea by considering appearance models based on *sparse representations* that have been widely used in Single Object Tracking. We focus on people tracking since most practical applications are dealing with this object category.

The first contribution of this thesis consists in designing an online, meaning frame by frame, tracking approach that takes advantage of collaborative sparse representations to define the affinity values between the estimated trajectories and the last detections. Furthermore, different possible descriptions of the targets, either holistic or local ones, have been considered.

Contrary to offline approaches that consider several frames, online approaches are not able to correct possible association errors like identity switches or track fragmentations. Therefore, we proposed for our second contribution to develop a tracking system with a sliding window, based on a MCMCDA approach, able to correct association errors by exploiting sparse representations well-suited for this specific framework.

Since the dictionaries used are composed solely of detections, the quality of the representations based on these dictionaries is highly dependent on the performance of the object detector. In order to rely less on the detector quality, we consider for the last contribution of this thesis to use dense dictionaries that are taking into account all possible locations of a target inside each frame.

Many quantitative evaluations were performed using usual and public datasets, notably those of the *MOTChallenge*, in order to provide a consistent comparison with other recent approaches. These evaluations show the gain in performances of our proposed contributions and demonstrate the relevance of the choices that had been made.

Keywords : Multi-object tracking, tracking by detection, sparse representations.

Table des matières

Chapitre I

Introduction générale

Sommaire

Ce premier chapitre présente de manière très générale le problème étudié dans cette thèse, à savoir le suivi visuel multi-objets. Bien que ce sujet ait déjà été largement étudié en Vision par Ordinateur, ce problème reste toujours complexe à traiter et les méthodes actuelles de suivi d'objets sont encore très loin d'égaler les performances de l'humain. Nous décrivons ici les principales difficultés qui rendent ce problème complexe et qui justifient son étude. Nous précisons ensuite l'axe de recherche général qui a été suivi tout au long de cette thèse et détaillons pour finir la structure de ce manuscrit.

I.1 Problème étudié

I.1.1 Suivi visuel d'objets

Le suivi visuel d'objets consiste à estimer les trajectoires de plusieurs objets d'intérêt, appelés cibles, à partir d'images de la scène observée fournies par une (ou des) caméra(s). Ce problème peut se voir comme une extension du problème de détection d'objets, où l'on cherche alors à détecter et localiser des objets d'intérêt au sein d'images. Contrairement à la détection d'objets, le suivi d'objets nécessite d'estimer à la fois la localisation des objets d'intérêt, mais aussi de leur attribuer une identité. Ces identités permettent alors d'estimer les trajectoires de chaque cible particulière, qui constituent les résultats de toute approche de suivi comme illustré en figure I.1.

Ces dernières décennies ont vu l'émergence, en grand nombre, d'ordinateurs avec une capacité de calcul suffisante pour réaliser des tâches complexes de Vision par

Ordinateur. Des caméras, ou d'autres capteurs visuels, étant de plus disponibles à des coûts abordables, il est désormais possible d'automatiser de nombreuses tâches en analysant les vidéos produites. Le suivi visuel d'objets est alors utile et nécessaire dans de nombreux domaines d'application, souvent requis comme une étape préliminaire pour permettre de raisonner ensuite sur les trajectoires estimées. On peut citer en premier lieu les problématiques de vidéo-surveillance, où le déploiement de systèmes autonomes permet la détection d'activités anormales sans nécessiter un effort fastidieux de surveillance humaine. On peut considérer aussi l'indexation automatique de vidéos, l'analyse de comportement cellulaire en micro-biologie ou encore les problématiques liées au développement de véhicules autonomes qui nécessitent une compréhension correcte de leur environnement.

Du fait de la grande diversité des utilisations possibles du suivi visuel, ce problème est traité de façons très variées afin de s'adapter au mieux aux besoins de l'application finale. Le problème du suivi visuel d'objets peut ainsi être fragmenté en un nombre important de sous-problèmes plus spécifiques. Parmi les différences principales qui séparent ces sous-problèmes, les plus significatives sont notamment :

- Le nombre de caméras utilisées (suivi mono-caméra ou multi-caméras).
- La façon dont la scène est observée, soit avec des caméras fixes, éventuellement calibrées, ou mobiles.
- Le nombre de cibles à suivre (suivi mono-objet ou multi-objets).
- La façon dont les vidéos sont analysées dans le temps, image après image ou directement avec l'ensemble de la vidéo (suivi en ligne ou hors ligne).
- La façon dont les objets à suivre sont précisés, manuellement ou automatiquement avec un détecteur d'objets.

Tous ces sous-problèmes mènent à des approches assez spécifiques, et il serait très ambitieux de chercher à étudier toutes ces différentes catégories. Cette thèse se focalise donc uniquement sur certains de ces sous-problèmes, comme détaillé dans ce qui suit.

I.1.2 Cadre de cette thèse

Dans cette thèse, nous considérons le problème de suivi d'objets avec les hypothèses suivantes :

(i) Une unique caméra couleur, non calibrée, fixe ou mobile, est considérée.

(ii) Plusieurs objets sont suivis simultanément (suivi multi-objets).

(iii) La classe des objets à suivre est connue.

(iv) Le fonctionnement de l'algorithme doit être proche du temps réel, avec un temps de latence faible.

Le premier critère (i) signifie que nous ne considérons pas les spécificités liées au suivi multi-caméras. La caméra étant supposée non calibrée, le suivi est considéré dans le repère de l'image (suivi 2D) et non de la scène observée (suivi 3D). En pratique, les approches proposées au chapitre III seront davantage adaptées pour une caméra fixe contrairement à celles proposées dans les chapitres suivants.

Le critère (ii) indique que nous nous focalisons sur le cas particulier du suivi multi-objets. Par rapport au suivi mono-objet, où une seule cible est considérée, les méthodes de suivi multi-objets supposent la présence simultanée de plusieurs cibles. Estimer la position des cibles n'est alors pas suffisant puisqu'il est aussi nécessaire de

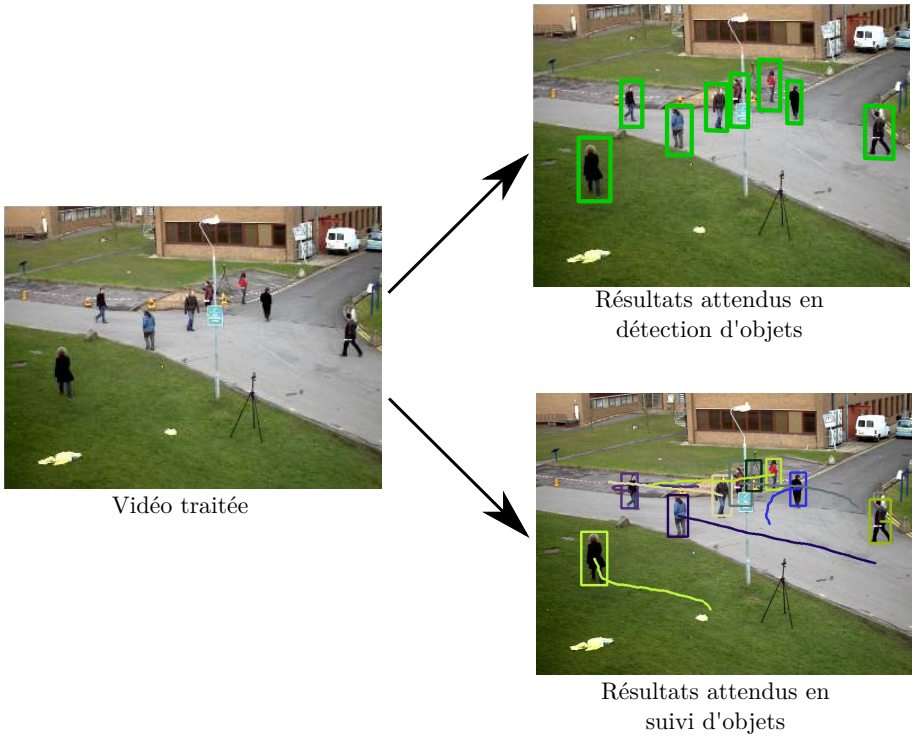


FIGURE I.1 – À partir d'une vidéo à traiter, résultats attendus en détection d'objets (en haut) où seules les positions des cibles sont à déterminer en chaque image. En bas, résultats attendus en suivi d'objets où les positions des objets doivent aussi être déterminées pour chaque image, mais en associant les positions liées à un même objet pour estimer des trajectoires tout au long de la vidéo.

pouvoir attribuer une identité spécifique à chaque cible et de maintenir ces identités au cours du temps.

La troisième hypothèse (iii) suppose que l'on dispose d'un a-priori sur la catégorie des objets que l'on cherche à suivre. Cela signifie en pratique qu'un détecteur d'objets approprié, qui génère en chaque image des détections traduisant une hypothèse de présence d'un objet, peut être utilisé (suivi par détection). Les approches de suivi proposées dans cette thèse supposent que l'on dispose d'un détecteur mono-classe. Bien que nos méthodes de suivi puissent être utilisées pour d'autres types d'objet, nous considérons principalement le cas du suivi de personnes. Ce choix est influencé par le fait que les applications utilisant des méthodes de suivi multi-objets considèrent principalement des personnes comme cibles, et que la plupart des bases de données permettant d'évaluer ces approches se concentrent sur le suivi de personnes multiples.

Les méthodes de suivi qui respectent ces trois premiers critères sont très nombreuses dans la littérature récente, et ce type de problème de suivi est très étudié actuellement. Par exemple, la base de données *MOTChallenge* (dans sa version 2015 [67]) a été proposée à la fin de l'année 2014 au cours de ces travaux de thèse. Deux années plus tard, 57 approches de suivi se sont évaluées publiquement sur cette base de données dont 30 correspondent à des publications dans des conférences ou journaux à comité de relecture. Le suivi multi-objets est ainsi un domaine assez étudié et très compétitif à l'heure actuelle.

Une contrainte supplémentaire (iv) est liée à l'utilisation concrète des méthodes

proposées. En effet, certaines méthodes de suivi nécessitent l'ensemble de la vidéo étudiée avant de pouvoir estimer les trajectoires des cibles. Cela mène à un temps de latence potentiellement important, c'est-à-dire que les résultats ne peuvent être estimés qu'après un laps de temps significatif. Dans cette thèse, nous cherchons à obtenir des méthodes de suivi dont le temps de latence est faible. Cela ne signifie pas forcément que le suivi sera réalisé image après image, mais que peu d'images futures pourront être utilisées pour déterminer les trajectoires à l'instant présent. De plus, cette contrainte impose de prendre en considération, avec attention, les temps de calcul des approches proposées et de rechercher des méthodes d'optimisation performantes afin d'aboutir à une vitesse de fonctionnement proche du temps réel.

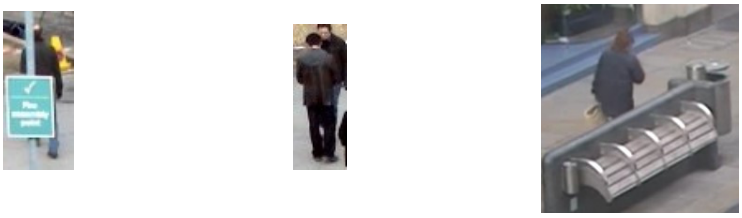
Dans le reste de ce manuscrit, le terme *suivi* ou *suivi d'objets* fait par défaut référence à un suivi visuel multi-objets mono-caméra avec détections. Les autres catégories de suivi sont néanmoins discutées, mais le type de suivi concerné est alors toujours précisé pour éviter toute confusion.

I.1.3 Principales difficultés

Nous abordons maintenant les principales difficultés qui rendent le problème de suivi multi-objets complexe. Toute approche de suivi multi-objets, par détection, est confrontée aux problèmes suivants :

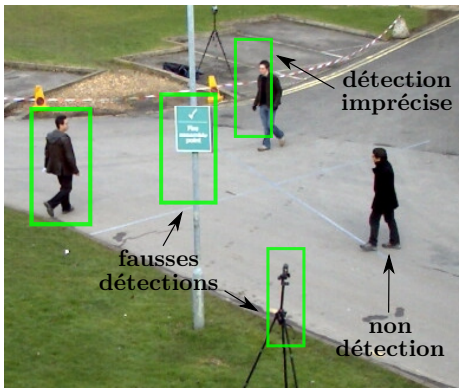
- **Occultations des cibles :** Les cibles peuvent être occultées, c'est-à-dire non visibles du fait d'éléments de la scène. Ces occultations peuvent notamment survenir lors de croisements entre différentes cibles. L'occultation peut alors être partielle, seule une partie de la cible restant visible, ou totale lorsque toute la cible n'est plus visible.
- **Maintien des identités :** Maintenir des identités correctes est compliqué du fait des croisements des cibles ou lorsque certaines cibles sont proches les unes des autres.
- **Fiabilité du détecteur d'objets :** Le détecteur d'objets employé est source d'erreurs, comme des fausses détections (faux positifs ou faux négatifs). La méthode de suivi doit alors être la plus robuste possible vis-à-vis de ces erreurs.
- **Variabilité des scènes observées :** Les scènes observées peuvent être assez variables, en particulier au niveau de l'orientation de la caméra (vue rasante ou plongeante), l'éventuel mouvement de la caméra (scènes fixes ou mobiles) et aussi en fonction de la densité des cibles observées. Une méthode de suivi pertinente doit pouvoir être suffisamment générique pour être appliquée à ces différents type de scènes, ou bien être en mesure de s'adapter automatiquement à chaque scène particulière.

Ces différentes difficultés sont illustrées en figure I.2. Comme détaillé en section II.2, généralement les méthodes de suivi visuel cherchent à exploiter des modèles d'apparence sophistiqués afin de différencier au mieux les cibles et les localiser plus efficacement. Cela permet ainsi d'être plus robuste aux occultations et de maintenir plus efficacement les identités des cibles. Cette stratégie est à la base de l'objet d'étude de cette thèse, comme précisé dans la section suivante.



Occultations des cibles

Maintien des identités



Fiabilité du détecteur d'objets

Variabilité des scènes observées
(Vidéos PETS S2L1 et TUD-Stadtmitte de la base de données *MOTChallenge* [67])

FIGURE I.2 – Principales difficultés à traiter en suivi multi-objets par détection.

I.2 Axe de recherche proposé et contributions

I.2.1 Axe d'études

L'axe d'études proposé dans cette thèse est expliqué ici, en indiquant succinctement nos motivations. Les différents points évoqués sont davantage détaillés dans le chapitre II. Les motivations plus spécifiques à chacune de nos contributions seront aussi précisées au début des chapitres associés.

Afin d'exploiter judicieusement l'information visuelle disponible, les méthodes de suivi visuel décrivent les cibles à l'aide de modèles d'apparence. Ces modèles d'apparence jouent un rôle crucial en suivi mono-objet pour localiser la cible dans les images suivantes. De nombreux modèles d'apparence ont ainsi été proposés pour le suivi mono-objet au cours des dernières années [113].

En suivi multi-objets par détection, le problème de la localisation des cibles est en grande partie traité par le détecteur d'objets utilisé. Une étape cruciale de ces approches de suivi consiste alors à déterminer des associations correctes entre les détections et les trajectoires estimées, pour éviter de confondre les cibles. Pour effectuer cette tâche plus efficacement, deux stratégies sont couramment employées. Une première consiste à raisonner sur plusieurs images futures pour traiter l'image courante, afin d'exploiter davantage d'information temporelle, ce qui mène à des approches dites à fenêtre glissante ou à logique différée. Une seconde stratégie consiste à exploiter des modèles d'apparence pour différencier les cibles et estimer leurs positions aux instants où elles ne sont pas détectées par le détecteur d'objets. Plusieurs méthodes de suivi multi-objets s'inspirent ainsi de modèles d'apparence proposés initialement pour le suivi mono-objet, et utilisent parfois même directement des méthodes de suivi mono-objet.

Les représentations parcimonieuses ont été largement utilisées en suivi mono-objet pour modéliser l'apparence de la cible [135]. De manière simplifiée, une représentation parcimonieuse vise à représenter un élément par une combinaison d'un faible nombre d'autres éléments connus, regroupés au sein d'un dictionnaire (la section II.3 donne une formalisation plus précise). Plusieurs méthodes en Vision par Ordinateur ont exploité de telles représentations, comme des problèmes de classification multi-classes, d'indexation de contenu ou encore pour définir des caractéristiques visuelles par sac de mots. Néanmoins, malgré leur emploi fréquent en suivi mono-objet, peu de méthodes de suivi multi-objets ont cherché à exploiter des représentations parcimonieuses pour modéliser l'apparence des cibles.

Dans cette thèse, nous étudions de quelle manière des représentations parcimonieuses peuvent être exploitées dans le cas du suivi multi-objets. Bien que nous nous inspirions initialement de méthodes proposées pour le suivi mono-objet, nous ne cherchons pas à employer directement de telles méthodes dans un système multi-objets. Nous cherchons plutôt à proposer des méthodes plus spécifiques au cas du suivi multi-objets afin de prendre en considération les difficultés particulières de ce type de suivi. Cela nous amène en particulier à étudier l'emploi de représentations parcimonieuses dans une approche de suivi à fenêtre glissante, en raisonnant sur plusieurs images consécutives futures, afin d'utiliser suffisamment d'information temporelle et d'exploiter judicieusement les informations visuelles qui en découlent.

I.2.2 Contributions

Les travaux effectués au cours de cette thèse peuvent se répartir en trois contributions principales.

Tout d'abord, au chapitre III, une première contribution est de proposer l'emploi de représentations parcimonieuses collaboratives globales dans une approche de suivi multi-objets en ligne (où les images sont traitées les unes après les autres sans aucune connaissance du futur). Contrairement aux méthodes de suivi mono-objet à représentations parcimonieuses, qui cherchent à lo-

caliser la cible, nous exploitons principalement les représentations parcimonieuses pour différencier les cibles. Cela permet notamment d'associer plus efficacement les détections, données par le détecteur d'objets, aux cibles adéquates. Nous utilisons alors ici des représentations collaboratives dites globales, c'est-à-dire qui font intervenir toutes les cibles suivies. Cela nous amène à considérer des représentations parcimonieuses faisant intervenir des dictionnaires composés d'un grand nombre d'éléments (plusieurs centaines voire milliers), ce qui rend le calcul des représentations parcimonieuses très coûteux. Nous proposons alors d'employer des techniques d'optimisation à base d'ensembles actifs pour accélérer significativement le calcul des représentations. Enfin, nous étendons cette première approche au cas de descriptions locales des cibles et étudions de nouveau comment les représentations parcimonieuses peuvent être adaptées pour de telles descriptions.

Plusieurs méthodes de suivi multi-objets exploitent davantage d'information temporelle, en s'autorisant à utiliser certaines images futures pour traiter l'instant présent afin de gagner en performances. **Au chapitre IV, notre seconde contribution consiste à proposer une approche de suivi à fenêtre glissante, raisonnant sur un faible nombre d'images futures, qui exploite des représentations parcimonieuses appropriées**. Employer des représentations parcimonieuses classiques ne s'avère alors pas particulièrement adapté dans ce contexte particulier. Néanmoins, il est possible de proposer des représentations parcimonieuses structurées afin de favoriser une structure de parcimonie plus appropriée à notre problème. Cela nous amène à considérer des représentations parcimonieuses définies à partir d'une norme de groupes particulière, une norme $l_{\infty,1}$ pondérée. Calculer de telles représentations s'avère plus complexe que les représentations parcimonieuses usuelles (en norme l_1). Nous montrons alors qu'il est possible d'adapter le protocole d'optimisation utilisé pour notre première contribution afin d'aboutir à une optimisation performante, suffisamment rapide pour garder un temps de latence modéré.

Dans nos précédentes approches, les représentations parcimonieuses font uniquement intervenir des détections données par le détecteur d'objets. Cela signifie que nos approches sont très pénalisées par certaines erreurs du détecteur d'objets et en particulier quand celui-ci ne détecte qu'occasionnellement certaines cibles. **Au chapitre V, notre dernière contribution est de considérer des représentations parcimonieuses qui peuvent faire intervenir des positions non détectées, afin d'être plus indépendant vis-à-vis de la qualité du détecteur d'objets employé**. Les représentations parcimonieuses ne sont plus définies à partir d'un dictionnaire composé uniquement de détections. À la place, nous considérons un dictionnaire qui inclut à la fois les détections et toutes les positions possibles dans une zone de recherche liée à la détection représentée. Les dictionnaires utilisés sont alors qualifiés de *denses* et permettent de représenter les détections par des positions non détectées par le détecteur d'objets. Le nombre d'éléments inclus dans ces dictionnaires est désormais beaucoup plus important (plusieurs centaines de milliers) et les méthodes d'optimisation précédemment utilisées sont alors inappropriées. Nous montrons qu'il est possible d'adapter ces méthodes, en s'inspirant des approches de reconnaissance de motifs (*template matching*) et de représentations parcimonieuses à convolutions, afin d'aboutir à une optimisation bien plus rapide.

Pour chacune de ces trois contributions, de nombreuses évaluations quantitatives sont effectuées. Ces évaluations permettent à la fois de justifier les choix effectués en comparant plusieurs variantes de nos approches, et aussi de comparer les approches

proposées aux autres méthodes récentes de l'état de l'art. Ces évaluations sont faites principalement à partir des métriques CLEARMOT [14] et des bases de données publiques du *MOTChallenge* [67, 86], comme précisé en section II.4. Les contributions proposées au cours de cette thèse ont permis d'aboutir à des approches très compétitives sur les bases de données du *MOTChallenge*, bases de données sur lesquelles se comparent un grand nombre d'approches très récentes de suivi multi-objets.

Les travaux réalisés au cours de cette thèse ont alors mené aux publications suivantes :

- **ORASIS 2015**, *Suivi multi-personnes à base de représentations parcimonieuses collaboratives globales*, Loïc Fagot-Bouquet, Romaric Audigier, Yoann Dhome, Frédéric Lerasle.
- **IEEE ICIP 2015**, *Online multi-person tracking based on global sparse collaborative representations*, Loïc Fagot-Bouquet, Romaric Audigier, Yoann Dhome, Frédéric Lerasle.
- **IEEE AVSS 2015 (oral)**, *Collaboration and spatialization for an efficient multi-person tracking via sparse representations*, Loïc Fagot-Bouquet, Romaric Audigier, Yoann Dhome, Frédéric Lerasle.
- **ECCV 2016**, *Improving multi-frame data association with sparse representations for robust near-online multi-object tracking*, Loïc Fagot-Bouquet, Romaric Audigier, Yoann Dhome, Frédéric Lerasle.

I.3 Structure du manuscrit

Ce manuscrit est structuré de la manière suivante :

- Le chapitre II positionne cette thèse par rapport à l'état de l'art lié au problème de suivi, principalement le suivi visuel multi-objets. Les pré-requis nécessaires pour aborder la suite du manuscrit sont de plus détaillés, notamment en ce qui concerne le principe des représentations parcimonieuses. Les protocoles d'évaluation employés sont aussi précisés à la fin de ce chapitre.
- Le chapitre III présente notre première contribution, c'est-à-dire l'emploi de représentations parcimonieuses collaboratives globales dans un système de suivi multi-objets en ligne.
- L'emploi de représentations structurées en norme $l_{\infty,1}$ au sein d'une approche de suivi à fenêtre glissante est détaillé dans le chapitre IV.
- Notre dernière contribution, sur l'utilisation de dictionnaires denses, est alors présentée dans le chapitre V.
- Le chapitre VI clôt la discussion principale de ce manuscrit, par une conclusion générale et une liste des perspectives de ces travaux.
- Certains éléments de détail, qui ne sont pas directement inclus dans les chapitres précédents, sont donnés en Annexes.

Chapitre II

Positionnement des travaux et pré-requis

Sommaire

Introduction

Ce premier chapitre présente la problématique étudiée dans cette thèse et les outils et techniques associées. Nous abordons notamment le problème de suivi de façon générale en section II.1 avant de nous focaliser au cas du suivi visuel en section II.2. Le concept de représentation parcimonieuse est ensuite précisé en section II.3 en nous focalisant sur l'utilisation de cet outil en Vision par Ordinateur, et plus précisément

en suivi mono-objet. Nous discutons enfin des méthodes d'évaluation existantes pour les approches de suivi multi-objets en section II.4, en détaillant les bases de données et les métriques usuelles pour évaluer les méthodes de suivi visuel.

II.1 Présentation générale du suivi d'objets

Dans cette section, nous présentons de manière très générale le problème de suivi d'objets et les approches classiques existantes pour traiter ce problème. Nous nous focalisons sur les approches mono-capteur, possiblement non visuel. Les méthodes présentées ici ne sont pas parmi les plus récentes, certaines étant même assez anciennes, mais restent à la base de la plupart des méthodes actuelles. Il semble ainsi judicieux d'expliquer leur fonctionnement.

II.1.1 Notions de base sur le suivi d'objets

Le problème du suivi d'objets considère l'estimation de trajectoires, notées $\mathcal{T} = \{T_1, ..., T_{n_{traj}}\}$, d'un ensemble d'objets ou cibles $\mathcal{O} = \{O_1, ..., O_{n_{obj}}\}$. Ces cibles évoluent dans un espace connu, le plus souvent multidimensionnel (par exemple 2D pour un suivi dans le repère image, 3D pour un suivi en coordonnées réelles). Ces cibles sont observables à partir d'un capteur (radar, sonar, détecteur optique d'objets visuel...) qui fournit un ensemble d'hypothèses sur la position des cibles $\mathcal{M} = \{m_1, ..., m_{n_{mes}}\}$, appelées aussi mesures ou observations. Ces mesures ne sont néanmoins pas parfaites, du fait d'erreurs liées au capteur, et sont notamment bruitées voire non pertinentes (présence d'une fausse alarme, i.e. une mesure non liée à une cible). Certaines cibles peuvent, de plus, être non détectées par le capteur. Un algorithme de suivi d'objets vise alors à estimer les trajectoires $\{T_1, ..., T_{n_{traj}}\}$ à partir des mesures $\{m_1, ..., m_{n_{mes}}\}$ effectuées au cours du temps.

Le suivi d'objets peut se traiter généralement en trois étapes principales. La première étape, d'association de données, consiste à associer les mesures obtenues $\{m_1, ..., m_{n_{mes}}\}$ aux trajectoires auparavant estimées $\{T_1, ..., T_{n_{traj}}\}$ afin de déterminer quelles mesures correspondent à quelles cibles. Une seconde étape va considérer le prolongement de chaque trajectoire à partir des mesures qui lui ont été associées. Enfin, une dernière partie considère la création de nouvelles trajectoires ou l'arrêt de certaines d'entre elles.

Il est alors possible de distinguer plusieurs grandes catégories de suivi d'objets, décrites dans ce qui suit.

Suivi mono-objet/multi-objets : Tout d'abord, une première distinction importante peut être faite au niveau du nombre de cibles suivies. Le problème de suivi mono-objet ne considère qu'une cible unique tandis que le problème de suivi multi-objets considère le cas de plusieurs cibles. Ces deux sous-catégories principales sont davantage détaillées, avec leurs propres spécificités, dans les sous-sections suivantes.

Suivi en ligne/global/à fenêtre glissante : Une autre distinction importante entre les méthodes de suivi est liée à la façon dont elles traitent les mesures au cours du temps. Si on suppose que le suivi est effectué sur une période $[0, \Delta T]$, on peut alors différencier trois principaux types d'approches :

(i) Les approches en ligne estiment les trajectoires jusqu'à l'instant t uniquement à partir des mesures présentes et passées (c'est-à-dire reçues jusqu'à l'instant t).

(ii) Les approches globales, qui estiment l'ensemble des trajectoires à partir de toutes les mesures de la période considérée $[0, \Delta T]$.

(iii) Les approches à fenêtre glissante (ou à logique différée, *multi-scan* ou par *batch*) qui considèrent les mesures présentes sur la période $[0, t + \Delta t]$ pour prédire les trajectoires jusqu'à l'instant t. Les mesures passées, présentes et celles dans un futur proche (Δt correspondant à un temps assez court) sont donc utilisées pour estimer les trajectoires jusqu'à l'instant courant.

Ces catégories d'approches sont illustrées en figure II.1, et le choix de l'une ou l'autre catégorie a en pratique un fort impact sur le temps de latence de la méthode de suivi. En effet, dans le cas d'approches en ligne, le temps de latence pour estimer les trajectoires est faible puisque les mesures sont traitées immédiatement. Dans le cas des approches hors ligne, le temps de latence est d'au moins Δt pour les approches à fenêtre glissante et d'au moins ΔT pour les approches globales.

Association de données déterministe/probabiliste : Une dernière distinction importante peut être faite vis-à-vis de la façon dont l'association de données est effectuée. Usuellement, l'association de données est formulée pour déterminer la meilleure configuration d'associations, cette configuration d'associations devant satisfaire certaines contraintes pour être admissible. Ces contraintes sont habituellement d'associer au plus une mesure de chaque instant temporel à chaque trajectoire et respectivement au plus une trajectoire à chaque mesure. Chaque trajectoire est ensuite prolongée à partir de l'unique mesure qui lui a potentiellement été associée à l'instant suivant et ce type d'association est alors dit déterministe. Le critère utilisé pour déterminer la meilleure configuration d'associations est fréquemment formulé de manière à maximiser une probabilité a-posteriori, étant données les mesures observées, et c'est pourquoi ces méthodes sont aussi appelées approches de type MAP (*Maximum A Posteriori*).

D'autres approches modélisent cette étape d'association de données pour obtenir des probabilités d'association entre chaque mesure et chaque trajectoire, probabilités obtenues en considérant l'ensemble des configurations d'associations possibles. Une trajectoire est alors prolongée en prenant en considération ses probabilités d'association avec plusieurs mesures, et cette association est alors dite probabiliste [1]. Les approches de suivi employant des méthodes d'association probabilistes sont aussi qualifiées d'approches Bayésiennes.

II.1.2 Spécificités du suivi mono-objet

Dans le cas du suivi mono-objet, une unique cible est considérée. Cela simplifie donc nettement le problème d'association de données ainsi que le problème de création des trajectoires. L'élément le plus important est donc d'arriver à prolonger correctement la trajectoire à partir des mesures qui lui sont associées.

Le suivi mono-objet est le plus souvent considéré en ligne, et on suppose dans un premier temps qu'une association de données déterministe est employée et asso-

1. Il faut faire attention à ne pas confondre ici une méthode de suivi déterministe/stochastique et une méthode de suivi utilisant une association de données déterministe/probabiliste. Une méthode de suivi est dite stochastique si l'algorithme associé est stochastique, par exemple si un échantillonnage par Monte Carlo est utilisé. L'association de données est probabiliste si on obtient en sortie des probabilités d'association. Une méthode de suivi peut ainsi être déterministe en faisant intervenir une association de données probabiliste, ou bien être stochastique et se reposer sur une association de données déterministe.

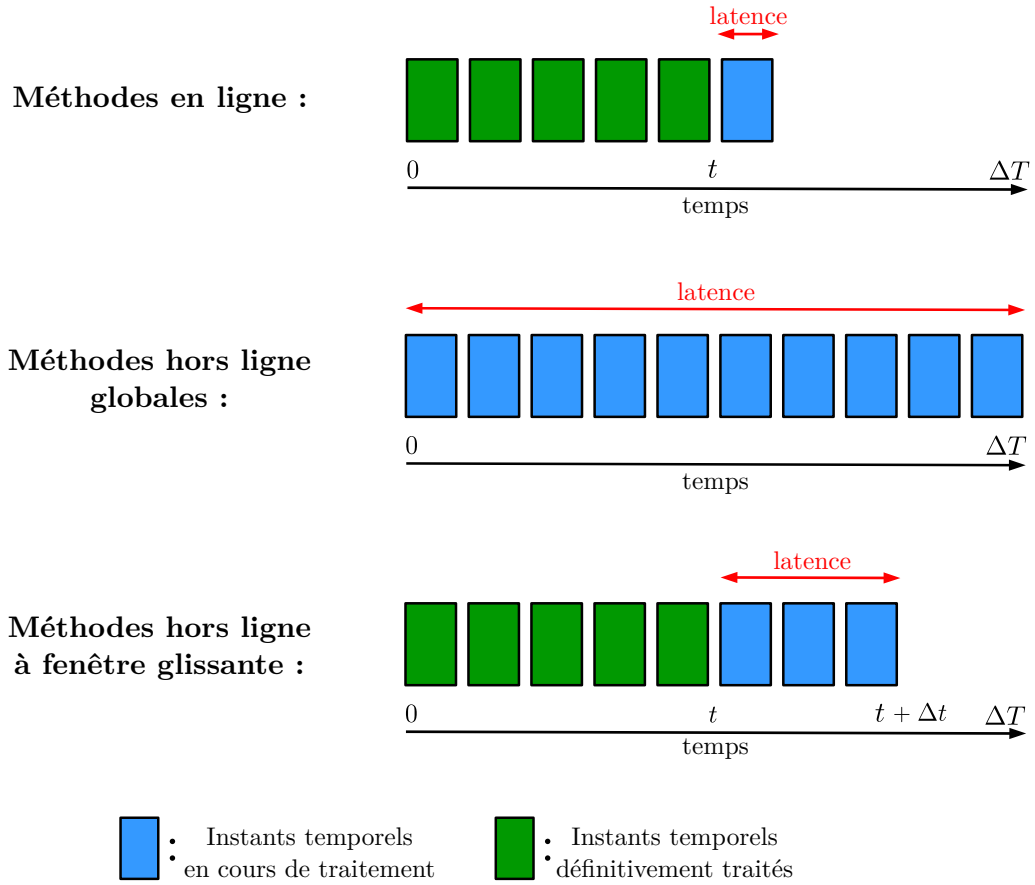


FIGURE II.1 – Différences entre les méthodes de suivi en ligne, globales et à fenêtre glissante. Ces méthodes se distinguent sur la façon dont les instants temporels sont traités, ce qui entraîne des différences en termes de temps de latence. Les approches en ligne peuvent donner des résultats pour le dernier instant temporel considéré tandis que les méthodes hors ligne (globales ou à fenêtre glissante) exploitent davantage d'information future et imposent donc un délai avant de pouvoir estimer les résultats à l'instant courant t.

cie au plus une mesure par instant temporel à la cible. Dans ce cas de figure, cela revient à vouloir déterminer la trajectoire de la cible à partir d'une série de mesures correspondant à cette cible. À l'instant courant t, après une éventuelle étape d'association de données, on dispose d'au plus une mesure m_t liée à la cible et il peut arriver qu'aucune mesure ne soit disponible. L'estimation de l'état courant x_t de la cible est alors effectuée selon les deux possibilités suivantes :

(i) **Une mesure est disponible pour l'instant courant :** estimer l'état x_t de la cible à partir des mesures $(m_i)_{i \leq t}$ en réalisant un filtrage.

(ii) **Aucune mesure n'est disponible pour l'instant courant :** estimer l'état x_t de la cible à partir des mesures $(m_i)_{i<t}$ en réalisant une prédiction.

Plusieurs filtres ont alors été proposés pour effectuer ces opérations de prédiction et de filtrage (le filtrage est usuellement réalisé à partir d'une prédiction initiale et d'une correction ou mise à jour du modèle à partir de la mesure de l'instant courant). Les deux principaux filtres employés sont le filtre de Kalman [59] et le filtre à particules [53], dont nous précisons rapidement le principe. Pour ces deux filtres, les états de la cible $(x_t)_{t \geq 0}$ sont supposés suivre un modèle de Markov du

premier ordre [2].

Filtre de Kalman : Le filtre de Kalman [59] estime l'état d'un système dynamique linéaire gaussien. Le modèle d'observation, déterminant la mesure m_t à partir de l'état x_t de la cible, est supposé lui-aussi linéaire avec un bruit gaussien. Sous ces hypothèses, le filtre de Kalman permet d'estimer récursivement les états du système de façon optimale (en considérant l'espérance de l'erreur au sens des moindres carrés). La solution déterminée par ce filtre admet une forme analytique qui fait principalement intervenir des opérations matricielles, notamment des inversions de matrices. Des extensions de ce filtre ont été ensuite proposées, comme par exemple le filtre de Kalman sans parfum UKF (*Unscented Kalman Filter*) [57] qui permet de traiter le cas où la dynamique de la cible et le modèle d'observation sont non-linéaires.

Filtre à particules : La principale limitation du filtre de Kalman, dans sa version initiale, est que la dynamique de la cible et le modèle d'observation sont supposés linéaires, et la loi a posteriori est modélisée par les deux premiers moments (donc gaussienne). Pour gérer le cas de fonctions non-linéaires, les filtres à particules sont une alternative aux approches de type filtre de Kalman, et notamment son extension UKF. Tandis que le filtre de Kalman sans parfum utilise un échantillonnage déterministe, les filtres à particules exploitent des échantillonnages stochastiques via des méthodes de Monte Carlo séquentielles. Ces filtres approximent la probabilité de présence à l'instant t, $p(x_t|(m_i)_{i\leq t})$, à partir d'un échantillonnage de cette loi de probabilité par un nombre fini de particules $\mathcal{P}_t = \{(p_1, w_1), ..., (p_{n_{par}}, w_{n_{par}})\}$. Chaque particule p_i est associée à un poids w_i qui traduit sa probabilité d'occurrence.

À l'instant courant t, durant une étape dite d'échantillonnage, un nouvel ensemble $\mathcal{P}_t$ est généré à partir des particules $\mathcal{P}_{t-1}$ en appliquant une distribution de proposition, qui traduit usuellement un modèle de transition entre l'instant $t-1$ et t. Les poids des nouvelles particules sont alors adaptés à partir de l'éventuelle mesure m_t et d'un modèle d'observation. Le nouvel état de l'objet peut être déterminé pour l'instant t en estimant la valeur $\mathbb{E}(x_t|(m_i)_{i\leq t})$ à partir des particules de $\mathcal{P}_t$.

Plusieurs variantes des filtres à particules ont été proposées, en particulier certaines qui considèrent des étapes de ré-échantillonnage afin d'éviter des cas de dégénérescence des particules. C'est notamment le cas de l'algorithme condensation (*Conditional Density Propagation*) qui effectue un ré-échantillonnage par importance [53].

PDAF (*Probabilistic Data Association Filter*) : Les méthodes de suivi mono-objet présentées ici considèrent une association de données déterministe. Dans le cas de mesures multiples à un même instant, cela signifie qu'au plus une mesure est considérée, après l'étape d'association, pour estimer le prolongement de la cible. Des méthodes d'association de données probabilistes ont été proposées, afin d'estimer le prolongement de la cible à partir de toutes les mesures disponibles et de leurs probabilités d'association. Ces approches sont de type PDAF (*Probabilistic Data Association Filter*) et prolongent habituellement la trajectoire de la cible avec un filtre de Kalman modifié pour prendre en compte des mesures multiples en chaque instant temporel [9].

2. $(x_t)_{t\geq 0}$ suit un modèle de Markov du premier ordre si $p(x_t|x_0, ..., x_{t-1}) = p(x_t|x_{t-1})$.

II.1.3 Spécificités du suivi multi-objets

Dans le cas du suivi multi-objets, une difficulté supplémentaire apparaît par rapport au suivi mono-objet. En effet, les mesures d'un même instant temporel peuvent correspondre à plusieurs cibles. L'étape d'association de données est alors cruciale pour répartir les mesures aux cibles correspondantes. Les approches de suivi multi-objets classiques ont ainsi tendance à se concentrer sur l'étape d'association de données, afin d'éviter au maximum des erreurs d'appariement entre les mesures et les cibles, et prolongent ensuite les trajectoires à partir des mesures associées en reprenant des méthodes de suivi mono-objet (filtre de Kalman, filtre à particules...). De plus, contrairement au suivi mono-objet qui est davantage réalisé en ligne, le suivi multi-objets est assez étudié dans un contexte hors ligne par des approches à fenêtre glissante ou globales. L'étape d'association de données étant ici bien plus difficile, les approches hors ligne permettent alors de résoudre cette étape de façon plus fiable en prenant en compte davantage d'information sur un horizon temporel.

Suivi multi-objets en ligne

Nous détaillons tout d'abord plusieurs méthodes usuelles de suivi multi-objets en ligne, qui estiment donc les trajectoires à l'instant courant uniquement à partir des mesures des instants précédents et de l'instant courant.

GNN (*Global Nearest Neighbors*) : Concernant les approches de suivi multi-objets en ligne, une première façon de procéder consiste à déterminer la configuration d'associations optimale C^* entre les trajectoires $\mathcal{T}_{t-1} = \{T_1, ..., T_{n_{traj}}\}$, estimées à l'instant $t-1$, et les mesures $\mathcal{M}_t = \{m_1, ..., m_{n_{mes}}\}$ à l'instant courant t. Cette configuration optimale C^* est alors déterminée parmi l'ensemble des configurations admissibles $\mathcal{C}$ de façon à minimiser une énergie E. La valeur $E(C)$ est usuellement formulée de manière à prendre en compte l'ensemble des distances ou coûts d'association $c_{T,m}$ pour chaque association (T, m) de la configuration C. La configuration C^* est ainsi déterminée comme une solution de :

$$\min_{C \in \mathcal{C}} \sum_{(T,m) \in C} c_{T,m} \,. \tag{II.1}$$

Cette méthode d'appariement est ainsi déterministe et est dénommée GNN [26]. Le principal désavantage de cette approche est que déterminer une unique configuration d'associations oblige à effectuer des choix d'appariement immédiatement entre les trajectoires de $\mathcal{T}_{t-1}$ et les mesures de $\mathcal{M}_t$. D'éventuelles erreurs d'association ne peuvent alors pas être corrigées au cours des instants suivants en prenant avantage de nouvelles mesures.

JPDA (*Joint Probabilistic Data Association*) : Une autre façon de procéder consiste à envisager une association de données probabiliste. Pour chaque trajectoire T_i de $\mathcal{T}_{t-1}$ et chaque mesure m_j de $\mathcal{M}_t$, le module d'association de données estime la probabilité β_{ij} que la mesure m_j corresponde à la cible de la trajectoire T_i. Chaque probabilité β_{ij} est définie en énumérant toutes les configurations d'associations C de $\mathcal{C}$ par :

$$\beta_{ij} = \sum_{C \in \mathcal{C}, (T_i, m_j) \in C} P(C | \mathcal{M}_0, ..., \mathcal{M}_t) \,, \tag{II.2}$$

où $P(C|\mathcal{M}_0, ..., \mathcal{M}_t)$ est la probabilité a posteriori associée à la configuration C. Une fois les probabilités β_{ij} estimées, celles-ci sont alors utilisées pour prolonger

les trajectoires à l'instant courant t en utilisant des approches de suivi mono-objet probabilistes, typiquement des filtres de Kalman à mesures multiples. Cette méthode est alors appelée JPDA [41]. Ce type d'approche évite de faire des appariements stricts entre une cible et une unique mesure en chaque instant, ce qui permet de mieux aborder les cas où l'association d'une cible vis-à-vis de plusieurs mesures est ambiguë. Néanmoins, calculer les probabilités β_{ij} de manière exacte est souvent très coûteux en temps de calcul (du fait du grand nombre de configurations C envisageables) et des heuristiques pour estimer ces probabilités sont nécessaires en pratique.

RJMCMC (*Reversible Jump Markov Chain Monte Carlo*) : Dans les approches de suivi multi-objets en ligne décrites précédemment, de type GNN ou JPDA, l'étape de prolongement des trajectoires est réalisée indépendamment pour chaque cible une fois l'association de données réalisée. Ces approches sont alors dites décentralisées, l'état de chaque cible étant modélisé individuellement. Cela est par exemple fait dans l'article [18] en utilisant une association de données de type GNN avec des filtres à particules spécifiques à chaque cible. Certaines méthodes de suivi, dites centralisées, modélisent au contraire directement l'état de l'ensemble des cibles. Plusieurs travaux ont par exemple considéré l'emploi d'un unique filtre à particules pour estimer la probabilité jointe, a posteriori, de l'état de l'ensemble des cibles $p(X_t|\mathcal{M}_0, ..., \mathcal{M}_t)$. X_t est alors ici un vecteur de l'état de l'ensemble des cibles à l'instant t. L'avantage est que le prolongement des trajectoires peut être effectué conjointement lors de l'étape d'échantillonnage. La distribution de proposition, qui génère les nouvelles particules $\mathcal{P}_t$ à partir de celles de $\mathcal{P}_{t-1}$, peut alors prendre en compte des interactions entre cibles. L'emploi de filtres à particules pour modéliser l'état joint des différentes cibles X_t est néanmoins complexe car X_t présente une dimension variable en fonction du nombre de cibles. Ce problème est alors traité dans [60] en ajoutant des mouvements de sauts réversibles du vecteur d'état qui permettent d'augmenter ou réduire le nombre de cibles durant l'échantillonnage et de suivre ainsi un nombre variable de cibles. Ce type d'approches de suivi est alors appelé RJMCMC.

PHD (*Probability Hypothesis Density*) : Comme expliqué précédemment, la modélisation de la probabilité a posteriori $p(X_t|\mathcal{M}_0, ..., \mathcal{M}_t)$ de l'état joint X_t des cibles est complexe du fait de la dimension variable de X_t. Si les approches de type RJMCMC traitent cette difficulté en modifiant l'étape d'échantillonnage de façon à pouvoir considérer l'ajout ou la suppression de cibles, une autre approche est d'aborder ce problème sans chercher à estimer l'identité des cibles mais seulement leur position. Les approches de type PHD [81] cherchent, de manière simplifiée, à estimer une fonction f qui permet de modéliser la présence d'une ou plusieurs cibles dans une zone donnée (en réalité la valeur de l'intégrale de f sur une zone donnée est une estimation du nombre d'objets présents). L'avantage est alors que cette fonction f est plus simple à estimer que l'état joint des cibles X_t, et permet de corriger le bruit des mesures tout en estimant le nombre de cibles observées. Cette estimation peut notamment être approchée avec un filtre à particules [98]. Le principal inconvénient est que les identités des cibles doivent être fixées ultérieurement par une méthode d'association de données.

Suivi multi-objets hors ligne

Nous précisons maintenant certaines approches classiques de suivi multi-objets hors ligne. Ces méthodes, globales ou à fenêtre glissante, considèrent alors plusieurs instants temporels futurs pour estimer de façon plus précise les trajectoires à l'instant courant.

MHT (*Multiple Hypothesis Tracking*) : Une première approche, appelée MHT, énumère de façon exhaustive l'ensemble des configurations d'associations possibles sur une fenêtre glissante d'une durée Δt [106]. L'association de données est alors déterministe en retenant uniquement la configuration C^* la plus probable. L'inconvénient majeur d'une telle approche est que l'énumération exhaustive de toutes les configurations sur la fenêtre glissante devient rapidement impraticable lorsque le nombre de cibles est important. Pour y remédier, l'énumération des configurations est faite sous la forme d'un arbre, et seules les hypothèses les plus probables sont gardées en élaguant l'arbre de recherche (*pruning*). L'avantage d'une approche MHT, comparée à une approche plus basique de type GNN, est de pouvoir corriger d'éventuelles erreurs d'association avec des mesures futures. Ses inconvénients sont cependant que les trajectoires sont déterminées avec un délai Δt et que le coût CPU d'une telle méthode est important. Réduire l'arbre de recherche de façon plus stricte peut améliorer le temps de calcul au détriment des performances, et un compromis doit ainsi être fait sur l'étape d'élagage de l'arbre (étape de *pruning*).

MCMCDA (*Markov Chain Monte Carlo Data Association*) : Une autre catégorie d'approches est constituée des méthodes de type MCMCDA. Plusieurs approches précédentes, en particulier les méthodes JPDA et MHT, nécessitent d'énumérer exhaustivement l'ensemble des configurations possibles que ce soit pour déterminer la configuration la plus probable (MHT) ou déterminer des probabilités d'association (JPDA). Ces deux approches sont limitées par cette énumération qui est rarement faisable exactement du fait de la complexité du problème lorsque le nombre de cibles est important. La méthode MCMCDA proposée dans l'article [99] évite cette énumération exhaustive en utilisant un échantillonnage de Monte Carlo par chaînes de Markov sur l'ensemble des configurations d'associations. La chaîne de Markov obtenue permet alors d'estimer les probabilités d'association β_{ij} comme fait par les méthodes JPDA ou de déterminer la configuration la plus probable C^* sur une fenêtre glissante, ce qui est l'objectif des méthodes MHT. Les méthodes MCMCDA peuvent ainsi traiter une association de données déterministe ou probabiliste, que ce soit en ligne ou hors ligne. Ces méthodes sont néanmoins le plus souvent employées avec une fenêtre glissante pour résoudre une association de données déterministe.

Les filtres à particules de type RJMCMC font aussi appel à un échantillonnage de Monte Carlo et à des chaînes de Markov, mais ne traitent pas exactement le même problème. Les méthodes de type RJMCMC échantillonnent la loi $p(X|\mathcal{M}_0, ..., \mathcal{M}_t)$ où X représente les positions de toutes les cibles. Les méthodes MCMCDA se concentrent sur l'association des données, en échantillonnant la loi $p(C|\mathcal{M}_0, ..., \mathcal{M}_t)$, et ne déterminent pas directement les positions des cibles mais seulement les mesures qui leur sont associées. En pratique, les méthodes de type RJMCMC se limitent usuellement à un suivi en ligne alors que les méthodes de type MCMCDA sont plutôt utilisées pour du suivi à fenêtre glissante.

BIP (*Binary Integer Programming*) : Une autre méthode de suivi multi-objets hors ligne est celle proposée dans l'article [92]. Dans cette approche, le problème d'association de données est formulé comme un problème de programmation

linéaire à valeurs booléennes (BIP). L'avantage principal est que le problème de programmation linéaire à valeurs booléennes étant étudié dans de nombreux domaines, des techniques d'optimisation efficaces existent pour le résoudre. Contrairement aux approches précédentes, cette méthode cherche explicitement à modéliser le problème de suivi sous une forme particulière afin de tirer avantage d'algorithmes d'optimisation déjà existants. Cette stratégie est très fréquemment employée dans les approches de suivi visuel multi-objets. Nous détaillons en sous-section II.2.3 plusieurs méthodes qui reformulent aussi le problème de suivi comme un problème d'optimisation appartenant à une catégorie spécifique de problèmes d'optimisation usuels (problèmes de flot maximal, problèmes de clique maximale...).

II.1.4 Positionnement de nos travaux

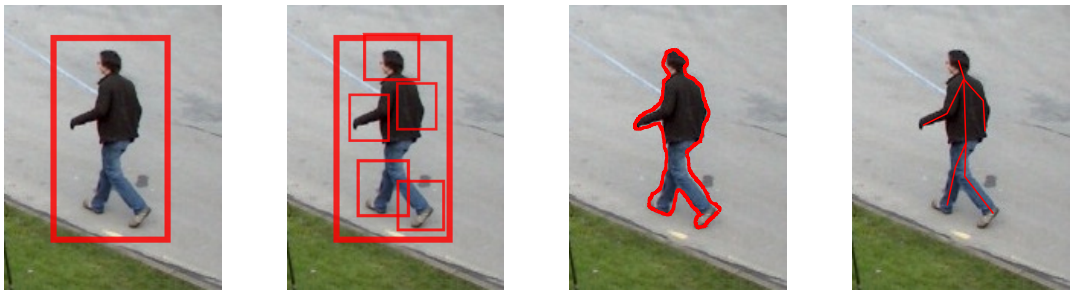
Au cours de cette thèse, nous nous focalisons sur le suivi multi-objets et considérons certaines méthodes d'association de données présentées dans cette section. Nous ne proposons pas de nouvelles méthodes d'association de données, notre stratégie principale étant de chercher à exploiter au mieux des représentations parcimonieuses, pour modéliser l'apparence des cibles, au sein de techniques d'association de données existantes.

Dans un premier temps, au chapitre III, nous étudions une approche de suivi multi-objets en ligne employant un système d'association assez basique, à savoir une association de données image après image formulée comme un problème d'appariement dans un graphe biparti. Cette première approche est alors de type GNN, présenté en sous-section II.1.3, où une solution C^* du problème d'appariement est déterminée en maximisant la somme des valeurs d'affinité de chaque couple de trajectoire et détection associées.

Afin de gagner en performances, nous exploitons ensuite au chapitre IV et au chapitre V davantage d'information temporelle au sein d'une approche hors ligne à fenêtre glissante. Nous considérons des approches à fenêtre glissante, qui raisonnent sur un faible nombre d'images futures, plutôt que des méthodes globales utilisant l'ensemble de la vidéo afin d'aboutir à un système de suivi fonctionnant avec une latence faible et respecter ainsi les contraintes précisées en sous-section I.1.2. Parmi les différentes techniques d'association de données possibles pour des approches à fenêtre glissante, comme présenté en sous-section II.1.3, nous utilisons une technique d'association de données de type MCMCDA. La raison principale qui motive ce choix est que nous privilégions une approche d'association imposant peu de contraintes sur l'énergie globale minimisée. Cela nous permet ainsi de nous concentrer sur la formulation de cette énergie et de chercher à la formuler pour exploiter au mieux les informations visuelles à notre disposition, au travers de représentations parcimonieuses plus complexes.

II.2 Méthodes existantes de suivi visuel

Après avoir discuté du suivi d'objets dans un cadre assez général, en présentant les approches classiques pour traiter ce problème, cette section aborde le cas plus spécifique du suivi visuel. Nous présentons les principales approches existantes en nous concentrant plus particulièrement sur les méthodes récentes et sur le suivi visuel multi-objets. Le cas du suivi visuel mono-objet est néanmoins abordé succinctement



(a) (b) (c) (d)

FIGURE II.2 – Illustration de différents modèles géométriques possibles des cibles en suivi visuel. (a) Boîte englobante. (b) Modèle articulé de type DPM. (c) Contour. (d) Squelette.

dans cette section. De plus, du fait des contraintes envisagées dans cette thèse, détaillées en section I.1, nous abordons uniquement les approches de suivi mono-caméra et ne détaillons pas les approches multi-caméras. Pour davantage de détails sur le suivi visuel, le lecteur peut se référer aux articles [79, 113, 129].

II.2.1 Spécificités du suivi visuel

Par rapport au cadre plus général du suivi d'objets discuté à la section précédente, il est possible de faire quelques observations spécifiques au suivi visuel. L'objectif du suivi visuel d'objets est toujours d'estimer un ensemble de trajectoires $\mathcal{T} = \{T_1, ..., T_{n_{traj}}\}$ d'objets d'intérêt à partir d'un ensemble de mesures $\mathcal{M} = \{m_1, ..., m_{n_{mes}}\}$. Ici, ces mesures résultent d'une caméra et se présentent sous la forme d'une séquence d'images $\mathcal{I} = \{I_1, ..., I_{n_{images}}\}$ de la scène observée au cours du temps. La différence principale par rapport à des techniques de suivi par radar ou sonar, par exemple, est qu'ici les mesures données par le capteur n'indiquent pas directement la présence d'une cible. Les mesures issues de la caméra, les images $\mathcal{I} = \{I_1, ..., I_{n_{images}}\}$, fournissent des informations plus riches sur la scène que la simple présence ou non présence des cibles.

Modèle géométrique des cibles

Une première considération importante concerne le modèle géométrique sous lequel les cibles sont représentées, c'est-à-dire sous quelle forme les cibles sont localisées en sortie de l'algorithme de suivi. Les modèles géométriques des cibles peuvent être en effet assez variés, et une liste non exhaustive est donnée ici :

(i) **Formes géométriques englobantes :** Le modèle géométrique de chaque cible est dans ce cas par un polygone régulier, le plus souvent un rectangle appelé boîte, ou un ellipsoïde englobant la cible. Cette forme géométrique est censée englober l'objet.

(ii) **Modèles articulés :** Le modèle géométrique considéré ici est constitué d'un ensemble de parties locales reliées entre elles, chaque partie locale étant représentée par une forme géométrique englobante. Un exemple classique d'une telle représentation est celle employée par le DPM (*Deformable Part Model*) [39] où les objets sont constitués d'un ensemble de parties locales représentées par des boîtes.

(iii) **Contour ou segmentation :** Les cibles sont ici décrites par l'ensemble des pixels qui les représentent dans l'image.

(iv) **Squelette :** Dans ce cas, le modèle géométrique considéré est un ensemble de segments qui suivent une certaine structure. Ce modèle est particulièrement employé dans la détection de pose et la reconnaissance de gestes et d'actions.

Ces différents modèles géométriques sont illustrés en figure II.2. Les modèles listés précédemment peuvent être considérés dans le repère image, en suivi 2D, ou dans un repère 3D lié à la scène. Dans la suite, nous nous limitons au modèle géométrique le plus classique en suivi visuel en supposant par défaut que chaque cible est représentée dans le plan image par une boîte englobante rectangulaire.

Caractéristiques visuelles

Si disposer d'une séquence d'images de la scène étudiée ne donne pas directement d'hypothèses sur la présence des cibles, ces images permettent néanmoins de décrire plus précisément les cibles qu'avec leur seule localisation. En Vision par Ordinateur, un grand nombre de caractéristiques visuelles ont été proposées. Les caractéristiques visuelles usuellement employées pour le suivi sont les valeurs d'intensité des pixels (en niveaux de gris ou en couleurs), des histogrammes d'intensité, des histogrammes d'orientation de gradients de type HOG (*Histograms of Oriented Gradients*) [28] ou encore des LBP (*Local Binary Pattern*) [100]. Des caractéristiques basées sur un flot optique peuvent aussi être employées, éventuellement via des HOF (*Histograms of Optical Flows*) [29]. Enfin, des caractéristiques issues de techniques d'apprentissage profond peuvent être employées, en réutilisant directement des caractéristiques apprises dans un contexte de détection d'objets ou en les apprenant de manière spécifique pour le suivi visuel.

Les cibles peuvent aussi être décrites par un ensemble d'éléments locaux, avec une description dite locale. Ces éléments peuvent être pris sur une grille régulière, déterminés via un détecteur de points d'intérêt (détecteur de Harris [47], SIFT [78], SURF [10]...), être trouvés par des méthodes de superpixels ou encore être donnés directement par un détecteur d'objets (par exemple de type DPM [39]).

Ces caractéristiques permettent de décrire chaque objet, le plus souvent en modélisant son apparence, et, dans le cas du suivi multi-objets, de différencier les différentes cibles les unes des autres.

Suivi avec ou sans détection

Une distinction peut être faite sur la façon dont les cibles sont précisées dans les méthodes de suivi visuel. Cette distinction mène à deux catégories de méthodes :

(i) Une première catégorie de méthodes considère le suivi d'une ou plusieurs classes d'objets connus et requiert un détecteur d'objets qui permet de donner des hypothèses sur la présence de ces objets. Usuellement, ces hypothèses prennent la forme de détections $Det = \{d_1, ..., d_{n_{det}}\}$ et chaque détection d est généralement associée à un score de confiance s_d. Les trajectoires des cibles sont alors estimées à partir des images et des détections, et ce genre d'approches est appelé *suivi par détection* (*tracking by detection*).

(ii) Une autre catégorie de méthodes suppose l'annotation manuelle des objets à suivre, annotation usuellement faite sur la première image de la vidéo étudiée.

Seuls les objets indiqués manuellement sont suivis, et ce type d'approches est dénommé *suivi sans modèle* (*model free tracking*) ou *suivi sans détection*.

Dans le cas du suivi par détection, le détecteur d'objets peut être en réalité employé de manières assez diverses. Tout d'abord, si la majorité des approches considèrent un ensemble de détections Det restreint aux détections les plus fiables, certaines méthodes utilisent des cartes de scores permettant d'évaluer toutes les positions envisagées. D'autres méthodes se reposent sur des détections de parties locales des cibles, ce qui est parfois considéré en suivi de personnes. Enfin, dans le cas de scènes fixes, certaines approches considèrent l'emploi d'une soustraction de fond (*background substraction*) pour indiquer les régions en mouvement de la scène. Dans ce cas, on se concentre sur le suivi de tous les objets mobiles.

Dans cette thèse, nous nous focalisons sur les méthodes du type (i), c'est à dire des méthodes de suivi par détection. Nous employons le détecteur d'objets de la façon la plus standard dans la littérature du suivi par détection en supposant que le détecteur d'objets nous fournit des boîtes englobantes des cibles avec éventuellement un score de détection.

Difficultés principales en suivi visuel

Les difficultés principales du suivi visuel, abordées en sous-section I.1.3, sont alors :

- Gérer les occultations des cibles, qui peuvent être partiellement ou totalement occultées par d'autres cibles ou éléments de la scène.
- Maintenir les identités des cibles, c'est-à-dire être capable de différencier les cibles en permanence. Malgré l'information visuelle disponible, les cibles sont généralement des objets d'une même classe (par exemple des personnes ou des voitures) et présentent donc des apparences visuelles qui partagent une structure générale et qui ne se différencient que sur certains aspects plus spécifiques.
- Gérer les variations des différents types de scènes possibles. Cela signifie que les méthodes doivent être robustes à une grande variabilité de types de scènes, notamment en terme d'angles de vue de la caméra, de mouvements de la caméra, d'illuminations de la scène...

En plus de ces difficultés générales, certaines difficultés sont plus spécifiques au type de suivi visuel envisagé. Par exemple, les approches de suivi sans détection apprennent le plus souvent un modèle d'apparence pour chaque cible qui est usuellement mis à jour en tenant compte des nouvelles positions estimées. Ces méthodes sont alors confrontées au problème de *dérive* des modèles mis à jour, ces modèles pouvant progressivement apprendre une position décalée de la cible et finir par apprendre un élément complètement différent de la scène, souvent une zone de l'arrière-plan.

Les méthodes de suivi par détection sont moins sujettes à ce problème de dérive, les modèles d'apparence propres à chaque cible reposant en grande partie sur les détections pour leur mise à jour et permettent une ré-initialisation automatique. Néanmoins, ces méthodes sont soumises aux défauts du détecteur d'objets employé et doivent alors être particulièrement robustes aux fausses détections et aux cibles non détectées. Nous avons privilégié cette approche au cours de cette thèse, en nous reposant fortement sur les détections données par le détecteur d'objets.

II.2.2 Suivi visuel mono-objet

Nous abordons maintenant les méthodes existantes pour le suivi visuel mono-objet. Le suivi visuel mono-objet est usuellement réalisé en ligne en supposant l'objet à suivre annoté sur la première image, sans a-priori sur la classe de l'objet suivi. Ce choix peut aisément s'expliquer car si un détecteur d'objets est employé, celui-ci peut détecter plusieurs objets pour chaque image et une approche de suivi multi-objets est alors privilégiée. On considère donc ici que la cible à suivre est uniquement localisée au sein de la première image de la vidéo, et on dispose seulement de la séquence d'images $\mathcal{I} = \{I_1, ..., I_{n_{images}}\}$ donnée par la caméra. Puisque l'on ne dispose pas d'hypothèses directement liées à la présence de la cible, les méthodes de suivi qui reposent sur de telles hypothèses, comme le filtre de Kalman, ne peuvent être directement employées. Les méthodes de suivi visuel mono-objet consistent alors le plus souvent à modéliser l'apparence de la cible afin d'estimer sa position aux instants suivants. Nous décrivons ici de manière très succincte le principe de ces méthodes, plus de détails sur les approches de suivi visuel mono-objet sont donnés au sein des articles [65, 113].

La plupart des approches estiment la nouvelle position x_{t+1} de la cible comme celle de l'image I_{t+1} la plus proche en apparence de la position x_t. Si y et y_x sont les caractéristiques visuelles respectivement liées à la cible et à une position candidate x de l'image I_{t+1}, on cherche usuellement à minimiser une erreur au sens des moindres carrés. Dans ce cas, x_{t+1} est déterminé comme solution du problème :

$$\min_{x \in I_{t+1}} ||y - y_x||_2^2 \,, \tag{II.3}$$

ou en maximisant une corrélation croisée (i.e. $y^\mathsf{T} y_x$). En pratique, on peut utiliser pour le choix de y la dernière vue de la cible, c'est-à-dire $y = y_{x_t}$, ou une combinaison de plusieurs vues récentes. Le principal inconvénient est ici de devoir examiner toutes les positions $x \in I_{t+1}$. Des méthodes approchées sont usuellement employées, notamment via l'emploi de l'algorithme MeanShift [17] ou de filtres à particules [53].

Au cours des dernières années, les méthodes de suivi visuel mono-objet à base de représentations parcimonieuses ont été assez populaires [135]. Plutôt que de comparer toutes les positions candidates x à une même caractéristique visuelle y, on adapte ici la caractéristique y de l'équation (II.3) à la position candidate. Pour chaque position candidate x, on considère le vecteur y qui minimise l'erreur $||y - y_x||_2^2$, tout en contraignant y à s'écrire sous la forme d'une combinaison linéaire d'un faible nombre de vues récentes de la cible $y_{x_{t-k}}, ..., y_{x_{t-1}}, y_{x_t}$. Ce vecteur y peut alors être déterminé en tant que représentation parcimonieuse de y_x, d'où appellation de ce type d'approches. Ces méthodes sont davantage précisées en sous-section II.3.3.

Des techniques d'apprentissage ont aussi été utilisées, principalement afin de déterminer un modèle d'apparence qui différencie la cible de son voisinage proche. De façon très générale, l'idée est ici de déterminer un classifieur qui attribue un score élevé aux caractéristiques visuelles pouvant provenir de la cible et un score faible à celles liées à son voisinage. Selon le modèle, les positions candidates x évaluées sont générées exhaustivement par une technique de fenêtre glissante ou bien le modèle d'apparence est utilisé avec un filtre à particules. Plusieurs techniques d'apprentissage ont été proposées, en particulier les techniques de type SVM [3], par ensemble de classifieurs faibles [58], avec instances multiples (MIL) [4], de régression d'arête (*ridge regression*) [49] ou encore de méthodes d'apprentissage profond [95]. Si la

majorité de ces méthodes visent à apprendre un modèle d'apparence discriminatif entre la cible et son voisinage proche, on peut néanmoins citer le cas particulier de la méthode Struck [46] qui cherche à modéliser directement le déplacement de la cible avec un SVM structuré.

Parmi les méthodes les plus courantes, celles qui sont récemment les plus populaires et performantes sont majoritairement issues de deux catégories d'approches. La première concerne les méthodes de type KCF (*Kernelized Correlation Filters*) [49] qui utilisent un apprentissage par régression d'arête (*ridge regression*). Ces méthodes exploitent des matrices circulantes et des techniques de convolution rapide par transformée de Fourier rapide afin de réaliser cet apprentissage avec un très grand nombre d'exemples. L'autre catégorie d'approches qui présentent actuellement de bonnes performances concerne les méthodes utilisant des techniques d'apprentissage profond, notamment pour apprendre des caractéristiques visuelles plus adaptées. On peut noter que la méthode [30], qui obtient les meilleurs résultats sur la dernière version du *Visual Object Tracking Challenge* [65], combine ces deux catégories en exploitant judicieusement des caractéristiques par apprentissage profond au sein d'un suivi de type KCF.

II.2.3 Suivi visuel multi-objets

Nous considérons maintenant le cas du suivi visuel multi-objets, qui est davantage étudié au cours de cette thèse. Tout d'abord, et contrairement au suivi visuel mono-objet, le suivi visuel multi-objets est principalement étudié en utilisant un détecteur d'objets qui fournit des hypothèses sur la position des cibles. Le suivi visuel multi-objets est traité à la fois par des approches en ligne et hors ligne, et nous présentons séparément les méthodes employées dans ces deux catégories. Le plus souvent, les méthodes de suivi visuel multi-objets reposent sur des techniques de suivi classiques, présentées en sous-section II.1.1, et utilisent l'information visuelle disponible au travers de modèles d'apparence afin de mieux modéliser et différencier les cibles.

Suivi visuel multi-objets en ligne

La majorité des approches de suivi visuel multi-objets en ligne séparent l'étape d'association de données de l'étape de gestion des trajectoires (c'est-à-dire gérer le prolongement des trajectoires et les éventuelles terminaisons et créations). Les rares approches qui considèrent conjointement ces deux étapes sont des méthodes où un unique filtre à particules est employé pour modéliser l'ensemble des cibles, en particulier les approche de type RJMCMC [60] ou de type PHD [80, 111].

Les autres méthodes en ligne réalisent le prolongement des trajectoires une fois l'association de données effectuée, et de manière indépendante pour chaque trajectoire. Le plus fréquemment, un filtre de Kalman est utilisé pour chaque trajectoire [6,51,61,112,126,128,130] mais d'autres méthodes de suivi mono-objet peuvent être employées comme des filtres à particules [18], un suivi par MeanShift [134] ou des méthodes à base de flot optique [127]. Un article récent [87] propose une autre alternative en apprenant à prolonger les trajectoires avec un modèle de type RNN (*Recurrent Neural Network*).

Concernant l'association de données, ces méthodes sont généralement de type GNN (ce type d'association de données est détaillé en sous-section II.1.3). Cela signifie que les détections à l'instant courant t, $Det_{I_t} = \{d_1, ..., d_{n_{det}}\}$ sont associées aux

trajectoires de l'instant précédent $\mathcal{T}_{t-1} = \{T_1, ..., T_{n_{traj}}\}$, en cherchant à maximiser [3] l'ensemble des affinités des appariements effectués. La configuration d'association optimale C^* est ainsi solution du problème :

$$\max_{C \in \mathcal{C}} \sum_{(T,d) \in C} Aff(T, d)\,. \tag{II.4}$$

Les approches de suivi visuel en ligne se distinguent ainsi principalement de la façon dont ces valeurs d'affinité $Aff(T, d)$ sont définies, ainsi qu'éventuellement sur la stratégie employée pour le prolongement des cibles (mais peu de méthodes n'emploient pas de filtre de Kalman à ce niveau). Ces valeurs d'affinité $Aff(T, d)$ font le plus souvent intervenir un modèle de mouvement de la trajectoire T ainsi qu'un modèle d'apparence. Le modèle de mouvement est généralement choisi de manière à favoriser une vélocité constante des cibles. Les principales différences entre les approches multi-objets en ligne se situent alors sur le modèle d'apparence employé et la description visuelle des cibles.

Dans le cas du suivi mono-objet, le modèle d'apparence est employé pour estimer la position de la cible à l'image suivante, éventuellement en apprenant à la différencier de son voisinage proche. En suivi multi-objets, les modèles d'apparence sont surtout employés pour différencier les cibles et permettre ainsi de limiter au maximum les erreurs d'appariements lors de l'étape d'association de données. Divers modèles d'apparence ont été employés dans la littérature récente. Si certaines méthodes formulent directement leur modèle d'apparence en comparant des caractéristiques visuelles [51, 126, 130], une stratégie de plus en plus employée consiste à apprendre des modèles d'apparence mis à jour tout au long du suivi. Dans certains cas, chaque cible se voit attribuer un modèle d'apparence spécifique pour la différencier des autres cibles. De tels modèles peuvent être appris par *boosting* comme proposé dans [18], par SVM [112], par apprentissage d'instances multiples (MIL) [61] ou encore par modèles de Markov caché (HMM) [128]. Une autre stratégie consiste à faire appel à des méthodes d'apprentissage de métrique [6, 15] qui cherchent à comparer les trajectoires et détections dans un espace plus approprié. Enfin, il a été proposé dans [127] d'apprendre à différencier directement les bons appariements (T, d) des appariements incorrects en exploitant plusieurs caractéristiques des trajectoires et détections. Un apprentissage par renforcement est utilisé à cet effet, et les caractéristiques employées ne sont pas limitées à la description de l'apparence des cibles (le modèle de mouvement étant appris conjointement).

Bien que les approches récentes se focalisent principalement sur les modèles d'apparence, certaines exploitent aussi les relations entre les différentes cibles dans la scène. En pratique cela revient à favoriser un mouvement d'ensemble des trajectoires, à maintenir des groupes de cibles ou encore à préserver une structure générale entre les cibles [51, 130].

Suivi visuel multi-objets hors ligne

Les approches de suivi visuel hors ligne, globales ou à fenêtre glissante, se focalisent principalement sur l'association de données. Cette association de données est dite *multi-images* puisque les éléments à associer les uns aux autres se répartissent

3. Ce problème d'association peut être formulé de façon à minimiser des coûts d'appariement, comme dans l'équation (II.1), ou de façon à maximiser des valeurs d'affinité.

sur une séquence d'images. Cette association de données est généralement formulée de manière déterministe comme un problème de minimisation d'énergie E (le plus souvent de type MAP). La configuration d'association optimale C^* est ainsi solution du problème :

$$\min_{C \in \mathcal{C}} E(C)\,, \tag{II.5}$$

et les trajectoires sont alors estimées à partir de cette configuration C^*. Ces méthodes se différencient principalement sur les points suivants :

(i) Sur quels éléments effectuer l'association de données ?

(ii) Quelle formulation pour l'énergie E ?

(iii) Quelle optimisation employer pour minimiser E ?

(iv) Quels modèles d'apparence employer pour tirer avantage de l'information visuelle ?

Eléments considérés dans l'association de données : Tout d'abord, sur le premier point (i), de nombreuses approches [19, 24, 31, 32, 43, 110, 120–122, 133] ne raisonnent pas directement sur les détections lors de l'association de données mais plutôt sur des fragments de trajectoires (*tracklets*). Ces fragments sont le plus souvent déterminés par des heuristiques basiques utilisées pour trouver de courtes trajectoires pertinentes. Certaines méthodes déterminent aussi de tels fragments en appliquant une première fois leur association de données au niveau des détections sur de courtes périodes. Ces approches font alors partie des méthodes incrémentales, qui raisonnent sur des éléments de plus en plus complexes, en partant des détections, pour obtenir les trajectoires finales des cibles [19, 84, 110, 133].

Formulation et minimisation de l'énergie : Concernant la formulation de l'énergie E et sa minimisation, c'est-à-dire les points (ii) et (iii), ces deux éléments sont en fait assez liés. Deux grandes tendances peuvent alors être distinguées. Une première catégorie de méthodes emploient des techniques de minimisation assez générales, qui n'imposent pas de contraintes particulières sur la formulation de l'énergie E. Ces approches peuvent ainsi employer des énergies plus complexes qui modélisent mieux le problème de suivi. Une seconde catégorie de méthodes formulent l'association de données sous la forme d'un problème plus spécifique, pour lequel des techniques d'optimisation efficaces existent. Ces méthodes ont le désavantage d'imposer certaines contraintes sur la formulation de l'énergie E mais permettent d'avoir davantage de garanties théoriques sur l'optimalité des solutions.

Concernant la première catégorie d'approches, qui emploient des méthodes d'optimisation assez générales sans imposer de contraintes particulières sur l'énergie E, des méthodes de suivi classiques de type MHT [106] ou MCMCDA [99] sont fréquemment employées [13, 43, 62, 77, 99, 110, 132]. Ces méthodes peuvent alors prendre facilement en compte dans l'énergie E des considérations complexes sur l'apparence ou le mouvement des cibles. Néanmoins, certaines approches se focalisent aussi sur une formulation pertinente de l'énergie E, qui est optimisée de manière approchée sans recourir à des techniques de type MHT ou MCMCDA [88, 89].

Pour la seconde catégorie d'approches, une formulation très populaire consiste à modéliser le problème de suivi multi-objets sous la forme d'un problème de flot [68, 69, 84, 102, 120]. Ce type de problème peut alors être résolu exactement par des méthodes de programmation linéaire ou par des algorithmes spécifiques. Bien que cette formulation soit l'une des rares pour lesquelles il est possible de déterminer

une solution exacte, cette formulation impose des contraintes assez limitantes pour l'énergie E. En particulier, comme relevé dans [84], il est difficile de prendre en compte des modèles de mouvements complexes (et c'est aussi le cas pour les modèles d'apparence ou d'interaction entre les cibles). Pour prendre en compte des modèles plus élaborés d'interaction, des contraintes sont ajoutées à l'écriture classique du problème de flot de poids minimal dans [122,123]. Cela revient néanmoins à considérer dans ces cas un problème IQP (*Integer Quadratic Program*) dans [122] ou MIP (*Mixed Integer Program*) dans [123], qui sont des problèmes plus complexes à résoudre. L'énergie E peut être écrite sous la forme d'autres problèmes spécifiques qui permettent d'employer des méthodes d'optimisation déjà existantes. Cette énergie E peut être formulée en tant que CRF (*Conditional Random Field*) [24,66,90], de BIP (*Binary Integer Program*) [31], de GLA (*Generalized Linear Assignment*) [32,121], de problème de clique de poids maximal [133], de problème d'ensemble indépendant de poids maximal [19] ou encore de problème de coupes multiples (*MultiCut*) [117].

Utilisation de l'information visuelle : Nous abordons maintenant le point (iv), à savoir comment les informations visuelles dont nous disposons peuvent être utilisées pour améliorer la qualité du suivi. La plupart des méthodes de suivi visuel multi-objets se ramènent à comparer en apparence les éléments concernés par l'association de données (i.e. des détections ou fragments de trajectoires), mais cette comparaison n'est réalisée en ne considérant ces éléments que deux à deux. La majorité des énergies E font en effet intervenir des coûts ou affinités d'association pour des paires d'éléments, qui peuvent être définis à partir de la cohérence en apparence de la paire considérée. L'approche la plus simple consiste à considérer la distance des caractéristiques visuelles concernées, mais des stratégies plus élaborées sont parfois employées. Des techniques d'apprentissage profond de caractéristiques ont été récemment proposées [68,120,121]. D'autres approches de suivi définissent des affinités en apparence à partir de résultats de DeepMatching [117], ou de trajectoires de points d'intérêt par flot optique [24]. Des approches de suivi à fenêtre glissante peuvent aussi utiliser des modèles d'apparence appris pour chaque cible, ce qui est par exemple fait dans l'article [62] en exploitant des modèles de type MORLS (*Multiple Output Regularized Least Squares*). Une utilisation plus originale de l'information visuelle disponible est proposée dans [69] où le mouvement de chaque détection est estimé à partir de son apparence, ce mouvement prédit étant ensuite utilisé pour définir une valeur d'affinité entre deux détections.

Détecteurs d'objets employés en suivi visuel multi-objets

Comme indiqué précédemment, la plupart des méthodes de suivi visuel multi-objets exploitent un détecteur d'objets qui donne un ensemble de détections représentant en chaque image des hypothèses de présence des cibles. Nous précisons ici certains détecteurs d'objets employés récemment par les méthodes de suivi, en nous focalisant sur le cas plus spécifique du suivi de personnes.

Pour les méthodes de suivi de personnes, les détecteurs de personnes utilisés sont généralement choisis en tenant compte des dernières avancées en détection d'objets. Ainsi, certaines méthodes de suivi exploitent un détecteur de personnes de type Dalal-Triggs [28]. Ce détecteur utilise des caractéristiques de type HOG et une méthode de classification de type SVM (*Support Vector Machines*) [25]. Les méthodes de classification de type SVM déterminent un modèle linéaire optimal vis-à-vis d'une certaine fonction objectif, fonction cherchant à classifier correctement les éléments

d'apprentissage sans pénaliser trop fortement certaines erreurs de classification liées à des éléments aberrants (via une pénalisation avec marges souples). Certaines méthodes de suivi utilisant un tel détecteur sont par exemple [13, 134].

D'autres détecteurs de personnes exploitent des caractéristiques de type histogrammes de gradient (HOG) mais avec des méthodes de classification qui diffèrent des SVM. Un tel exemple de détecteur est dénommé ACF (*Aggregate Channel Features*) [33], dont la technique de classification repose sur une méthode à base d'arbres de décisions entraînés avec des méthodes de *Boosting*. L'avantage de ce détecteur est de pouvoir traiter efficacement plusieurs échelles de détection en calculant rapidement les caractéristiques visuelles, par l'approximation de certaines caractéristiques à partir de celles calculées pour une échelle proche. Ce détecteur a notamment été employé pour générer les détections publiques données par la base de données *MOTChallenge* pour sa version 2015 [67]. De nombreuses méthodes de suivi se sont donc évaluées avec un tel détecteur, par exemple [24, 88, 102, 127].

Un autre détecteur fréquemment employé est le DPM (*Deformable Part Model*) [39] qui exploite aussi des caractéristiques de type HOG. Cependant, les personnes sont représentées sous la forme d'un ensemble de parties articulées, et le modèle fait intervenir des coûts de déformation entre ces parties. La méthode de classification fait au final appel à un modèle linéaire de type SVM, mais les parties du modèles sont déterminées durant l'apprentissage et sont donc latentes. La détection des meilleures configurations de parties dans une image peut alors être grandement accélérée par des transformées de distances (*distance transforms*) et des stratégies de type *coarse-to-fine*. Ce détecteur est utilisé pour générer les détections publiques données par la version 2016 du *MOTChallenge* et plusieurs approches récentes s'évaluent ainsi avec un tel détecteur [24, 102, 108, 117]. Certaines méthodes exploitent aussi spécifiquement les parties déterminées par ce détecteur [112].

Pour finir, les méthodes les plus récentes de détection de personnes font appel à des méthodes d'apprentissage profond, méthodes qui visent à apprendre simultanément les caractéristiques visuelles et la méthode de classification. Un détecteur de ce type, actuellement très populaire, est le *Faster-RCNN* [107]. Ce détecteur apprend notamment un détecteur rapide (*Region Proposal Network* ou RPN) qui élimine un grand nombre de positions possibles en ne retenant que quelques centaines de positions candidates. Des méthodes de suivi récentes qui se basent sur un tel détecteur sont par exemple [70, 131].

II.2.4 Positionnement de nos travaux

Les travaux de thèse présentés dans ce manuscrit suivent une stratégie habituellement employé en suivi visuel, qui consiste à exploiter l'information visuelle disponible au travers de modèles d'apparence élaborés. Des modèles d'apparence sont ainsi usuellement employés pour tous les différents types de suivi visuel, que ce soit le suivi mono-objet ou le suivi multi-objets en ligne et hors ligne. Dans les approches de suivi visuel mono-objet récentes, les modèles d'apparence jouent un rôle crucial. En effet, la plupart de ces méthodes cherchent à suivre une cible sans aucune connaissance préalable sur l'objet suivi. Des modèles d'apparence élaborés sont nécessaires pour modéliser au mieux la cible et permettre de la localiser sur les images suivantes. En suivi multi-objets par détection, la localisation des cibles est en partie traitée par le détecteur d'objets employé. Néanmoins, plusieurs méthodes

s'inspirent de modèles d'apparence proposés pour le suivi mono-objet afin de différencier efficacement les cibles et les localiser sur les images présentant des cibles non détectées.

Bien que de nombreuses méthodes de suivi mono-objet à base de représentations parcimonieuses aient été proposées [135], peu d'approches de suivi multi-objets exploitent cet outil pour modéliser l'apparence des cibles. Dans cette thèse, nous proposons de ce fait d'étudier de quelles façons des représentations parcimonieuses peuvent être exploitées au mieux au sein de méthodes de suivi multi-objets. Une approche naïve consisterait à employer directement des modèles d'apparence à base de représentations parcimonieuses de méthodes de suivi mono-objet au sein d'une approche multi-objets. Cependant, les difficultés principales auxquelles sont confrontées les méthodes de suivi mono-objet et multi-objets par détection sont assez différentes (la localisation de la cible est un aspect crucial en suivi mono-objet, tandis que l'association de données, et donc la différenciation des cibles, est un facteur important en suivi multi-objets). Nous cherchons donc à traiter plus spécifiquement les difficultés du suivi multi-objets, par détection, en adaptant de façon appropriée des modèles à base de représentations parcimonieuses. En particulier, nous proposons au chapitre III d'employer des représentations parcimonieuses collaboratives pour mieux différencier les cibles, tandis que nous proposons au chapitre IV d'utiliser des représentations structurées pour exploiter judicieusement l'information temporelle disponible dans les approches de suivi à fenêtre glissante. Pour finir, au chapitre V, nous cherchons à rendre notre approche de suivi plus robuste aux défauts du détecteur d'objets à l'aide de représentations parcimonieuses définies sur des dictionnaires denses.

II.3 Représentations parcimonieuses

Nous avons discuté dans la section précédente des méthodes de suivi visuel récentes. Ces méthodes exploitent généralement l'information visuelle disponible, via des modèles d'apparence, afin de mieux localiser ou différencier les cibles et permettre ainsi un suivi plus fiable. En suivi mono-objet, comme discuté en sous-section II.2.2, de nombreuses méthodes récentes utilisent à cette fin des représentations parcimonieuses. Nous détaillons ici le concept de représentations parcimonieuses ainsi que la façon dont il a été employé au cours des dernières années en Vision par Ordinateur.

Les représentations parcimonieuses étant employées dans des domaines très variés (par exemple en traitement du signal, en bio-informatique, en traitement d'image, en apprentissage automatique...) et de manières assez diverses, il est délicat de présenter l'ensemble des techniques associées. Dans cette section nous ne donnons qu'un aperçu général du principe des représentations parcimonieuses en nous focalisant grandement sur leur usage en Vision par Ordinateur et surtout en suivi mono-objet.

II.3.1 Principe général

On suppose ici que l'on dispose d'un vecteur $y \in \mathbb{R}^n$, par exemple un vecteur caractéristique d'un objet, et d'un dictionnaire sous forme matricielle $D \in \mathbb{R}^{n \times m}$. Ce dictionnaire D peut être vu comme la réunion de m éléments de $\mathbb{R}^n$, $\{e_1, ..., e_m\}$, chacun de ces éléments constituant une colonne de la matrice D. Une représentation

de y, par rapport au dictionnaire D, consiste à déterminer un ensemble de coefficients $\alpha \in \mathbb{R}^m$ de telle sorte que $y \approx D\alpha$. Cela revient ainsi à approximer le vecteur d'origine y comme une combinaison linéaire des éléments du dictionnaire D :

$$y \approx D\alpha \tag{II.6}$$
$$\approx \alpha_1 e_1 + \alpha_2 e_2 + ... + \alpha_m e_m \,. \tag{II.7}$$

Le vecteur α constitue alors une représentation du vecteur y par rapport au dictionnaire D, et $D\alpha$ est la reconstruction liée à cette représentation. Afin d'avoir une représentation pertinente, on cherche le plus souvent à déterminer une représentation α^* qui minimise [4] une erreur de reconstruction :

$$\min_{\alpha \in \mathbb{R}^m} \frac{1}{2} ||y - D\alpha||_2^2 \,. \tag{II.8}$$

La valeur $||y - D\alpha||_2^2$ est l'erreur de reconstruction considérée, ici au sens des moindres carrés [5].

Dans de nombreux domaines d'application, par exemple en compression de données, on cherche de plus à avoir une représentation de y qui fait intervenir un faible nombre d'éléments du dictionnaire D. Cela revient à vouloir déterminer un vecteur $\alpha \in \mathbb{R}^m$ de telle sorte que $y \approx D\alpha$ et en contraignant α à avoir seulement un faible nombre de coefficients non nuls (i.e. les coefficients α_i qui seront associés aux éléments e_i qui interviennent concrètement au sein de la reconstruction $D\alpha$). Une telle représentation est alors appelée *représentation parcimonieuse* [6] de y. Plusieurs stratégies peuvent être employées pour déterminer une représentation qui satisfait de telles conditions [82].

Une stratégie usuelle est de modifier l'équation (II.8) de manière à pénaliser les représentations faisant intervenir un grand nombre d'éléments du dictionnaire. On peut alors déterminer une représentation parcimonieuse α^* comme solution du problème :

$$\min_{\alpha \in \mathbb{R}^m} \frac{1}{2} ||y - D\alpha||_2^2 + \lambda \Omega(\alpha) \,, \tag{II.9}$$

où Ω est ici une pénalisation choisie pour favoriser la parcimonie de la solution et $\lambda \in \mathbb{R}^{+*}$ est un paramètre pondérant l'influence de cette pénalisation. Une première possibilité pour le choix de Ω est d'employer la pseudo-norme l_0, la valeur $||\alpha||_0$ étant exactement le nombre de coefficients non nuls de α. Cette pénalisation correspond précisément à la notion de parcimonie mais le problème (II.9) avec un tel choix devient très complexe à optimiser (le problème n'étant alors ni convexe ni même continu) même si des méthodes d'optimisation approchées existent néanmoins pour

4. Bien que l'on ait l'existence d'un minimum global pour cette expression, l'unicité du minimum global dépend du dictionnaire D. Les représentations définies ici, et dans le reste de ce manuscrit, désigneront un élément parmi ceux qui minimisent globalement une certaine fonction objectif. Cette fonction objectif n'admettra pas nécessairement un unique minimum global, seule l'existence d'un minimum global sera satisfaite.

5. D'autres formulations pour l'erreur de reconstruction peuvent être envisagées, notamment en employant directement la norme l_2 avec la valeur $||y - D\alpha||_2$ ou de manière plus générale avec la valeur $dist(y, D\alpha)$ en considérant une distance $dist$ quelconque.

6. Un vecteur x est dit parcimonieux s'il ne comporte qu'une faible nombre de coefficients non nuls.

le résoudre. Classiquement, on emploie la norme l_1 [7] en tant que pénalisation Ω. La représentation parcimonieuse α^* est alors déterminée dans ce cas comme solution du problème :

$$\min_{\alpha \in \mathbb{R}^m} \frac{1}{2}||y - D\alpha||_2^2 + \lambda||\alpha||_1 \,. \tag{II.10}$$

L'avantage de définir les représentations parcimonieuses à partir d'une norme l_1 est que le problème (II.10) est convexe, et des méthodes efficaces d'optimisation permettent de déterminer un minimum global de façon exacte [5, 11, 101]. De plus, il a été montré que la norme l_1 permettait effectivement de favoriser la parcimonie des solutions [82].

Dans ce manuscrit, l'expression *représentation parcimonieuse* fera alors toujours référence à une représentation déterminée en minimisant le problème (II.9) pour une certaine pénalisation Ω, qui sera précisée si nécessaire.

II.3.2 Utilisation en Vision par Ordinateur

Les représentations parcimonieuses ont été utilisées pour différentes tâches en Vision par Ordinateur au cours des dernières années. Nous présentons ici certaines applications de ces représentations. L'utilisation de représentations parcimonieuses en suivi mono-objet sera discutée de manière plus approfondie à la sous-section suivante.

Applications en traitement d'images

Plusieurs problèmes de traitement d'images ont été abordés avec des représentations parcimonieuses. Ces problèmes incluent notamment le débruitage d'images, la colorisation automatique d'images ou encore la reconstruction d'images détériorées (*inpainting*) [82]. Toutes ces applications peuvent être vues comme des problèmes où une image présente certaines informations détériorées ou manquantes et doit alors être corrigée. Le principe général est de considérer un dictionnaire D constitué de patchs issus d'images réelles (dictionnaire éventuellement appris pour être davantage adapté au contexte d'application) puis de remplacer chaque patch p de l'image à corriger à partir de sa reconstruction $D\alpha_p$. Ici, α_p est obtenu en déterminant une représentation parcimonieuse du patch p par rapport au dictionnaire D.

Description par sac de mots

Les descriptions par sac de mots sont utilisées en Vision par Ordinateur pour donner une description fiable de différents éléments (images complètes, objets, actions...). On suppose ici que chaque élément à décrire e est composé d'un ensemble de sous-éléments $\{e_1, ..., e_{n_e}\}$. Le principe est alors de considérer un dictionnaire D composé de plusieurs sous-éléments $\{c_1, ..., c_{n_c}\}$ (qui ne sont pas liés à l'élément décrit e). Ce dictionnaire peut par exemple être déterminé par une approche de type k-moyennes (*k-means*) sur un grand ensemble connu de sous-éléments possibles. Deux étapes sont alors considérées pour décrire un élément e :

(i) **Codage :** Chaque sous-élément e_i est associé à un code $c(e_i) \in \mathbb{R}^{n_c}$.

7. La norme l_1 est définie par : $||\alpha||_1 = \sum_i |\alpha_i|$.

(ii) **Agglomération (*pooling*) :** Les codes $\{c(e_1), ..., c(e_{n_e})\}$ sont agglomérés au sein d'un unique vecteur $y_e \in \mathbb{R}^{n_c}$ qui représente l'élément e.

L'étape d'agglomération des codes est usuellement faite en considérant leur somme (*sum-pooling*), leur moyenne (*average-pooling*) ou leur maximum pour chaque coordonnée (*max-pooling*). L'étape de codage peut être faite en cherchant le code $c(e_i)$ comme une représentation du sous-élément e_i, c'est-à-dire de telle sorte que $e_i \approx D.c(e_i)$. Cette représentation $c(e_i)$ peut être contrainte de manière à ne faire intervenir qu'un seul élément du dictionnaire (celui le plus proche de e_i) ce qui amène à un codage strict (*hard coding*). Une autre stratégie consiste alors à définir le code $c(e_i)$ comme une représentation parcimonieuse, c'est-à-dire en minimisant le problème :

$$\min_{c \in \mathbb{R}^{n_c}} \frac{1}{2}||e_i - Dc||_2^2 + \lambda||c||_1 \,. \tag{II.11}$$

Cela amène à un codage parcimonieux (*sparse coding*) qui a été fréquemment employé pour les descriptions par sac de mots [16].

Classification multi-classes

Les représentations parcimonieuses ont aussi été utilisées pour des problèmes de classification multi-classes, à l'origine en reconnaissance faciale [124]. On considère ici k classes distinctes $L_1, ..., L_k$ et on cherche à estimer la classe d'un nouvelle élément y. Un dictionnaire D est alors employé, qui est composé de dictionnaires D_L spécifiques à chaque classe L. En termes de matrices, le dictionnaire D peut se voir comme la concaténation (respectivement aux colonnes) de ces sous-dictionnaires :

$$D = [D_{L_1} \; ... \; D_{L_k}] \,. \tag{II.12}$$

Une représentation parcimonieuse collaborative α_y, c'est-à-dire qui considère conjointement toutes les classes $L_1, ..., L_k$, est alors déterminée pour l'élément y en tant que solution du problème :

$$\min_{\alpha} \frac{1}{2}||y - D\alpha||_2^2 + \lambda||\alpha||_1 \,. \tag{II.13}$$

Cette représentation parcimonieuse collaborative α_y peut ensuite s'écrire comme la concaténation de représentations α_y^L propres à chaque sous-dictionnaire D_L :

$$\alpha_y = [\alpha_y^{L_1} \; ... \; \alpha_y^{L_k}] \,. \tag{II.14}$$

Chaque vecteur α_y^L est la restriction de la représentation parcimonieuse α_y en ne considérant que les dimensions liées à des éléments du sous-dictionnaire D_L. La reconstruction partielle $D_L\alpha_y^L$ est alors la reconstruction résiduelle de y par rapport aux éléments du dictionnaire D_L. La classe de l'élément y est finalement estimée comme étant celle qui minimise l'erreur de reconstruction résiduelle :

$$L^* = \arg\min_{L} ||y - D_L\alpha_y^L||_2 \,. \tag{II.15}$$

Cette méthode de classification est appelée SRC (*Sparse Representation-based Classification*) [124]. L'emploi des représentations parcimonieuses permet alors de différencier efficacement un ensemble de classes similaires (dont la variation inter-classes

est faible). La représentation parcimonieuse α_y va en effet chercher à représenter y uniquement avec un faible nombre d'éléments du dictionnaire D. Cela permet ainsi de se focaliser sur un très faible nombre de classes, celles qui permettent de reconstruire au mieux y, même dans le cas où toutes les classes sont similaires. Cet effet semble particulièrement adapté au cas de la reconnaissance faciale, les classes étant assez similaires puisqu'elles correspondent toutes à des visages (chaque classe correspondant aux vues d'une même personne).

L'approche initiale considère pour chaque classe un sous-dictionnaire D_L qui est simplement constitué d'un certain nombre d'éléments de la classe L. Plusieurs extensions de cette approche ont été proposées, et en particulier certaines qui visent à apprendre le dictionnaire D de façon à rendre les représentations parcimonieuses plus discriminantes entre les différentes classes [64].

II.3.3 Représentations parcimonieuses et suivi mono-objet

Au cours des dernières années, plusieurs méthodes de suivi mono-objet à base de représentations parcimonieuses ont été proposées. Nous décrivons le principe général de ces approches en précisant les spécificités de quelques unes d'entre elles. Davantage de détails sur le suivi mono-objet à base de représentations parcimonieuses sont donnés dans l'article [135] qui analyse un grand nombre des méthodes de suivi de cette catégorie.

Principe général

Le principe de ces méthodes consiste à employer un filtre à particules dont le modèle d'observation fait intervenir des représentations parcimonieuses. L'état de la cible est supposé avoir été estimé avant l'instant courant t, par les états $(x_i)_{i<t}$. Un dictionnaire D contient alors des éléments représentant certaines vues précédentes de la cible, c'est-à-dire des éléments de $(y_{x_i})_{i<t}$ où y_x est la caractéristique visuelle à la position caractérisée par l'état x. Le modèle d'observation, utilisé pour juger la pertinence des données observées vis-à-vis de chaque hypothèse d'état x de la cible à l'instant t (modélisé par une particule), se base sur l'erreur de reconstruction $||y_x - D\alpha_{y_x}||_2^2$:

$$p(m_t|x_t = x) = \frac{1}{Z} \exp^{-\frac{1}{\sigma^2}||y_x - D\alpha_{y_x}||_2^2}, \tag{II.16}$$

où m_t représente les observations à l'instant t, Z est une constante de normalisation et où σ est choisi de façon à pénaliser plus ou moins les erreurs de reconstruction. α_{y_x} est la représentation parcimonieuse de y_x par rapport au dictionnaire D, c'est-à-dire déterminée comme solution du problème :

$$\min_{\alpha} \frac{1}{2}||y_x - D\alpha||_2^2 + \lambda||\alpha||_1 . \tag{II.17}$$

L'état x_t de la cible à l'instant t est alors estimé à partir des particules $\mathcal{P}_t$ à l'instant t, et le dictionnaire D est éventuellement mis à jour pour tenir compte de ce nouvel état.

Dans l'approche initiale de suivi mono-objet avec représentations parcimonieuses [85], les caractéristiques visuelles considérées sont simplement les valeurs d'intensité de la boîte estimant la position de la cible. Ce choix est assez fréquent pour

ces approches de suivi, même si certaines méthodes exploitent simultanément plusieurs caractéristiques visuelles classiques (HOG, histogrammes de couleur...) [50]. Le dictionnaire D n'est pas forcément constitué uniquement de vues de la cible, des éléments dits triviaux sont parfois ajoutés pour modéliser d'éventuelles occultations (cette stratégie n'est cependant pas employée par certaines approches [135]). Une limitation importante de cette technique de suivi est que le calcul des représentations parcimonieuses est assez coûteux. Des avancées récentes en optimisation convexe ont cependant été exploitées pour aboutir à des approches temps réel [8].

Extensions de l'approche initiale

Plusieurs extensions de ce principe initial ont été proposées. Une première extension [140] inclut dans le dictionnaire D des vues de la cible mais aussi des vues de son voisinage proche. Le dictionnaire s'écrit ainsi comme la concaténation de deux sous-dictionnaires, $D = [D_C\ D_V]$ avec D_C dictionnaire lié à la cible et D_V lié à son voisinage. Tout comme l'approche initiale, le modèle d'observation fait intervenir la représentation parcimonieuse α_{y_x} de y_x par rapport au dictionnaire D. Cette représentation est alors collaborative, de manière similaire à la méthode de classification SRC [124]. Pour juger de la pertinence de la position x, le modèle d'observation ne considère pas l'erreur de reconstruction complète mais les erreurs de reconstructions résiduelles liées à chacun de ces deux sous-dictionnaires D_C et D_V. Cela revient ainsi à proposer un modèle d'apparence discriminatif entre la cible et son voisinage proche.

Une autre extension [55] considère une représentation locale de la cible, à partir de patchs locaux pris sur une grille régulière. Le dictionnaire D est constitué ici des caractéristiques visuelles des patchs de certaines vues récentes de la cible. Le modèle d'observation prend alors en compte les représentations parcimonieuses de tous les patchs locaux d'une position candidate. Néanmoins, ces représentations parcimonieuses sont considérées via une erreur de reconstruction résiduelle, dont la reconstruction fait intervenir les patchs du dictionnaire qui sont localisés de manière similaire sur la cible. Les représentations parcimonieuses peuvent ici être vues comme collaboratives entre tous les patchs, et cette collaboration aide à différencier les patchs locaux fiables des autres.

Toutes les variantes évoquées jusqu'à présent ne considèrent que des représentations parcimonieuses usuelles définies à partir d'une norme l_1. D'autres pénalisations ont été proposées pour calculer des représentations parcimonieuses avec une structure plus adaptée. Des normes de groupes de type $l_{1,q}$, avec $q \geq 1$, ont notamment été utilisées pour reconstruire conjointement toutes les particules ou bien pour exploiter de façon plus appropriée des descriptions avec des caractéristiques visuelles multiples [50, 137, 138].

Pour finir, une méthode de suivi récente exploite des représentations parcimonieuses avec convolutions [136]. Le principe est alors de pouvoir représenter y_x avec les éléments du dictionnaire D, mais aussi avec tous les décalages par translation de ces éléments. L'intérêt est de pouvoir relocaliser une particule sur la cible en observant avec quelle translation les éléments de D la représentent. Cela permet de réduire significativement le nombre de particules utilisées pour estimer la nouvelle position de la cible.

II.3.4 Positionnement de nos travaux

Dans cette thèse, des représentations parcimonieuses sont exploitées au sein d'approches de suivi multi-objets. Nous étudions en particulier comment ces représentations parcimonieuses peuvent être adaptées de manière à être plus appropriées pour notre application de suivi multi-objets. Nous étudions notamment plusieurs possibilités sur le dictionnaire employé pour définir ces représentations parcimonieuses et la pénalisation Ω employée dans l'équation (II.9). Ainsi, au chapitre III, des représentations parcimonieuses usuelles en norme l_1 sont employées avec des dictionnaires collaboratifs obtenus à partir de vues de l'ensemble des cibles observées. Dans le chapitre IV, des représentations parcimonieuses structurées sont définies en employant pour pénalisation Ω une norme $l_{\infty,1}$ pondérée. Une telle pénalisation permet alors de favoriser une structure de parcimonie plus adaptée à une approche de suivi avec fenêtre glissante. Pour finir, nous utilisons au chapitre V des représentations parcimonieuses définies à partir de dictionnaires denses qui incluent un nombre bien plus important d'éléments, qui correspondent à des positions non détectées dans les images.

Afin d'aboutir à des méthodes de suivi suffisamment rapides, une attention particulière est portée à l'optimisation de ces représentations parcimonieuses. Nous utilisons des méthodes d'optimisation proximales [101], en employant notamment des variantes à base d'ensembles actifs [5] afin de pouvoir gérer des dictionnaires contenant un grand nombre d'éléments. Ces méthodes sont notamment adaptées pour traiter le cas de la pénalisation en norme $l_{\infty,1}$ pondérée proposée au chapitre IV ainsi que les dictionnaires denses proposés au chapitre V.

Bien que nous étudiions différentes possibilités pour définir le dictionnaire employé, nous ne considérons cependant pas de techniques d'apprentissage de dictionnaires. Cela est dû à deux raisons principales. Tout d'abord, les méthodes d'apprentissage de dictionnaires font appel à des optimisations assez coûteuses en temps de calcul, qui sont donc difficilement compatibles avec des méthodes de suivi devant présenter une latence faible. La seconde raison est que nous utilisons principalement les représentations parcimonieuses pour mettre en évidence les détections les plus similaires afin de favoriser leur association. Pour ce faire, nous cherchons à représenter chaque détection par l'ensemble des autres détections, c'est-à-dire à considérer un dictionnaire dont les éléments correspondent directement à une détection. L'objectif des méthodes d'apprentissage de dictionnaires n'est ainsi pas réellement approprié, ces méthodes cherchant à apprendre un dictionnaire, souvent restreint à un nombre assez limité d'éléments, qui permet de reconstruire au mieux les éléments [82] ou à être suffisamment discriminatif pour représenter un nombre restreint de classes [64].

II.4 Méthodes d'évaluation pour le suivi multi-objets

Cette section détaille les protocoles utilisés pour évaluer et comparer les approches de suivi multi-objets, en précisant les choix effectués à ce sujet pour comparer nos méthodes de suivi.

II.4.1 Bases de données

Les bases de données disponibles pour le suivi multi-objets sont majoritairement spécifiques au cas du suivi de personnes multiples, du fait du grand nombre d'applications qui considèrent les personnes comme objet d'intérêt. De nouvelles bases de données de plus en plus pertinentes ont vu le jour au cours de cette thèse, et cela explique que diverses bases de données aient été utilisées pour tester nos approches de suivi.

Au début de cette thèse, la plupart des méthodes en suivi multi-objets étaient évaluées sur un nombre réduit de vidéos, comme les vidéos de PETS [40], TownCenter [13], ParkingLot [112], TUD [1, 2] ou ETH [35]. Pour certaines de ces séquences, des jeux de détections sont disponibles pour permettre de s'évaluer avec le même détecteur d'objets. De plus, certains auteurs donnent les trajectoires estimées par leur méthode ce qui permet de se comparer avec des métriques différentes. Pour nos premiers travaux, nous avons considéré pour ces différentes raisons quatre vidéos fréquemment employées, à savoir PETS S2L1, PETS S2L2, TownCenter et ParkingLot. Nous utilisons de plus des jeux de détections publiques, donnés par les articles [88] pour les vidéos de PETS, par [13] pour TownCenter et enfin par [112] pour ParkingLot. Les vérités terrains sont alors celles fournies par [13, 88, 112]. Ces quatre vidéos sont des scènes fixes avec des vues plongeantes de la scène, cas assez fréquent dans les applications de vidéo-surveillance. Ces vidéos se distinguent notamment au niveau de leur fréquence et de la densité des cibles. Des images de ces quatre vidéos sont données en figure II.3.

Les bases de données du *MOTChallenge* ont été proposées plus récemment, avec une première version en 2015, intitulée *2DMOT15* [67], et une seconde en 2016, intitulée *MOT16* [86]. Ces bases de données incluent plusieurs vidéos réparties en un ensemble d'entraînement et un ensemble de test (22 vidéos au total pour la version 2015 et 14 pour la version 2016). Si certaines de ces vidéos sont nouvelles, d'autres proviennent d'anciens travaux [1, 2, 12, 35, 40, 45]. L'avantage de ces deux bases de données est qu'un ensemble de détections publiques est fourni pour chaque version (avec un détecteur de type ACF [33] pour la version 2015 et de type DPM [39] pour la version 2016), et que ces vidéos présentent davantage de diversité comparées aux quatre vidéos précédentes. Ces vidéos sont en effet issues de caméras fixes ou mobiles, avec des orientations de caméra variées (vue plongeante ou rasante...) et des densités variables de personnes. Certaines images des vidéos du *MOTChallenge* sont présentées en figure II.4.

De plus, une vérité terrain par vidéo est fournie, ce qui s'avère nécessaire pour évaluer les méthodes de suivi sur les vidéos d'entraînement et de test. Ces vérités terrain incluent les trajectoires de toutes les cibles qui doivent être estimées, c'est-à-dire leurs positions et identités en chaque image. Si ces vérités terrain sont issues d'anciennes bases de données pour la version 2015, et ne suivent donc pas exactement les mêmes conventions d'annotation, les vérités terrain pour la version 2016 ont été réalisées en suivant le même protocole comme décrit dans l'article [86].

II.4.2 Métriques employées

L'évaluation quantitative des méthodes de suivi multi-objets nécessite non seulement des bases de données mais aussi des métriques adaptées pour permettre de comparer les différentes approches. Cependant, il est délicat de définir une mesure

PETS S2L1

PETS S2L2

TownCenter

ParkingLot

FIGURE II.3 – Images provenant des vidéos PETS S2L1, PETS S2L2, TownCenter et ParkingLot.

unique de performance pour le problème de suivi multi-objets. La principale difficulté est ici que les résultats d'une méthode de suivi multi-objets peuvent être affectés par des erreurs de types assez divers. Certaines erreurs seront en particulier plus liées à la détection des cibles tandis que d'autres résulteront davantage de problèmes d'association des détections.

La performance d'une méthode de suivi est aussi fortement dépendante de l'application qui exploite ses résultats. Selon l'application, il peut être par exemple plus crucial de maintenir les identités des cibles (en évitant des changements d'identité) ou bien de limiter les fragmentations de trajectoires (c'est-à-dire estimer une seule trajectoire par cible) ou encore d'estimer précisément la localisation des cibles. De ce fait, la plupart des travaux qui ont cherché à évaluer quantitativement les méthodes de suivi multi-objets [14, 74, 97, 114, 115] utilisent un ensemble de métriques afin de prendre en considération les différents types d'erreurs possibles. Une synthèse des différentes métriques existantes est réalisée dans l'article [91].

Au cours des dernières années, la communauté travaillant sur le suivi multi-objets a fini par utiliser presque exclusivement les métriques CLEARMOT de l'article [14]. D'autres métriques mesurant davantage la qualité des trajectoires, proposées dans les articles [75,125], ont aussi été massivement adoptées et ajoutées le plus souvent en supplément des métriques CLEARMOT. Ces deux catégories de métriques reposent sur un appariement entre les hypothèses de trajectoires données par l'approche de suivi, notées $\mathcal{H} = H_1, ..., H_{n_H}$ et les cibles de la vérité terrain notées $\mathcal{O} = O_1, ..., O_{n_O}$.

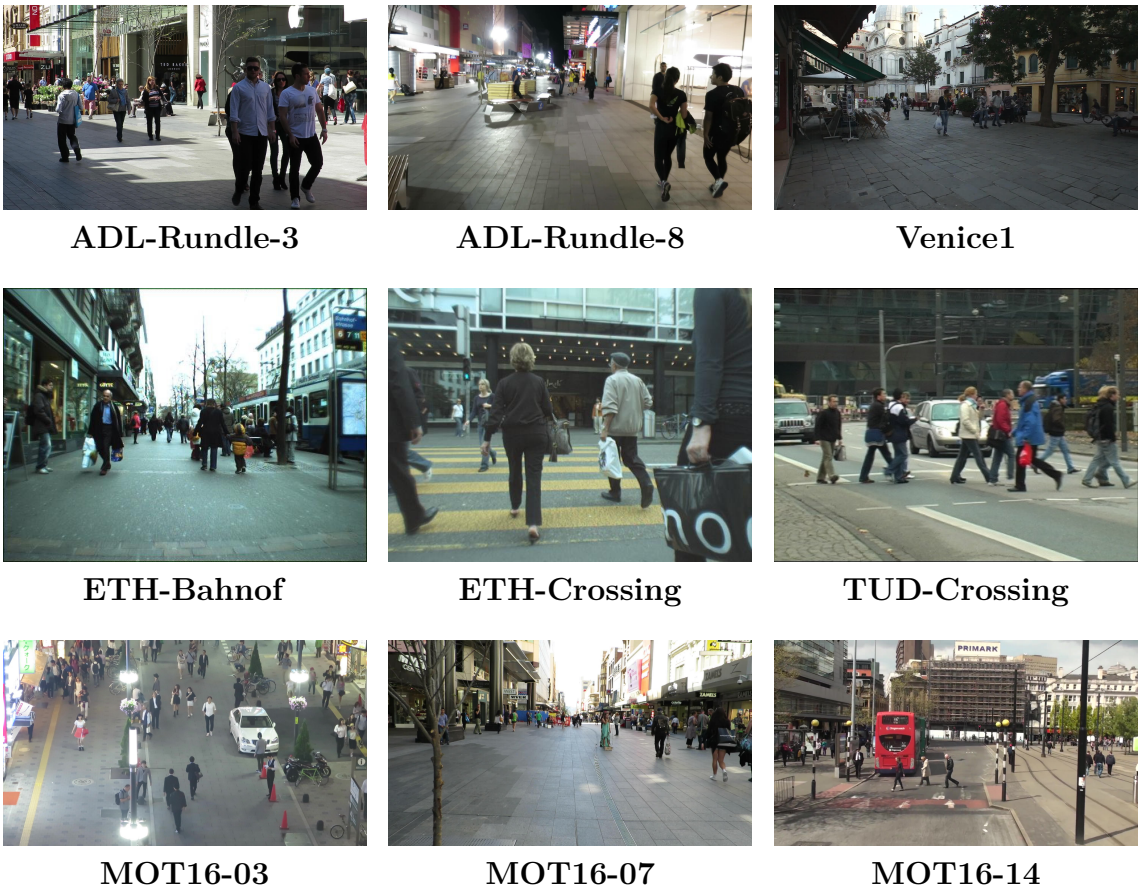


FIGURE II.4 – Images provenant de certaines vidéos du *MOTChallenge*.

Nous précisons rapidement cette étape d'appariement, en détaillant le protocole suivi par [14], dans ce qui suit.

Appariement des hypothèses de trajectoires à la vérité terrain

L'appariement des hypothèses de trajectoires aux cibles de la vérité terrain nécessite d'associer à chaque image de la vidéo traitée les positions de ces hypothèses et de ces cibles. Si cela peut être évident dans le cas où les cibles sont éloignées les unes des autres, cet appariement devient beaucoup plus ambigu dans le cas de cibles proches. Le protocole suivi pour effectuer cette étape pour les métriques CLEAR-MOT considère les images I_t de la vidéo étudiée les unes après les autres de la manière suivante :

(i) Pour toute hypothèse H associée à une cible O à l'image précédente I_{t-1}, si H et O sont toutes deux présentes pour l'image I_t et suffisamment proches (i.e. $d_{CLEARMOT}(H, O) \leq th_{CLEARMOT}$) alors considérer H et O toujours associées pour l'image I_t. $d_{CLEARMOT}$ est une distance considérant les positions des hypothèses de trajectoires et cibles dans l'image I_t, et $th_{CLEARMOT}$ est un seuil limite d'appariement.

(ii) Pour toutes les hypothèses de $\mathcal{H}$ et cibles de $\mathcal{O}$ présentes dans l'image I_t mais non associées à l'étape (i) précédente, effectuer un appariement qui minimise la somme des distances $d_{CLEARMOT}$ des hypothèses et cibles associées. On

suppose de plus que seuls des couples (H, O) qui vérifient $d_{CLEARMOT}(H, O) \leq th_{CLEARMOT}$ peuvent être associés. Ce problème peut alors être formulé comme un appariement de poids minimal qui est résolu de manière optimale par un algorithme hongrois [93].

Ce protocole permet bien d'associer en chaque image les hypothèses de trajectoires aux cibles de la vérité terrain. Le plus souvent, en suivi dit 2D, la distance $d_{CLEARMOT}$ est usuellement le ratio entre l'aire de l'intersection et l'aire de l'union des deux boîtes considérées (notée IOU, pour *Intersection Over Union*). Le seuil $th_{CLEARMOT}$ est usuellement fixé à $\frac{1}{2}$. Il est maintenant possible de définir les métriques CLEARMOT, et celles sur la qualité des trajectoires, à partir des appariements déterminés ici.

Métriques CLEARMOT et autres métriques usuelles

Les métriques CLEARMOT font intervenir certaines métriques usuelles, déterminées à partir de l'appariement des résultats de suivi à la vérité terrain, comme le nombre de faux positifs FP, le nombre de faux négatifs FN et le nombre de changements d'identité IDS (*IDentity Switches*). Toute position d'une hypothèse de trajectoire non associée à une cible de la vérité terrain est considérée comme un faux positif, tandis que toute position d'une cible de la vérité terrain non associée à une hypothèse de trajectoire compte pour un faux négatif. Enfin, une cible dont la dernière hypothèse de trajectoire associée était H_i et qui se retrouve associée à une hypothèse de trajectoire H_j avec $H_j \neq H_i$ compte pour un changement d'identité (IDS). À partir de ces éléments, deux nouvelles métriques sont définies dans les métriques CLEARMOT :

(i) La métrique MOTA (*Multiple Object Tracking Accuracy*) définie par :

$$MOTA = 1 - \frac{FN + FP + IDS}{P_{\mathcal{O}}}, \tag{II.18}$$

où $P_{\mathcal{O}}$ est le nombre total de positions de cibles de la vérité terrain sur l'ensemble de la vidéo.

(ii) La métrique MOTP (*Multiple Object Tracking Precision*), valeur moyenne des distances $d_{CLEARMOT}(H_j, O_k)$ des associations effectuées entre les positions des hypothèses de trajectoires et celles des cibles de la vérité terrain.

Ainsi, le MOTP considère la qualité de la localisation des cibles tandis que le MOTA est une valeur qui prend en compte plusieurs types d'erreurs, de détection (FN et FP) et d'appariement (IDS).

Ces métriques CLEARMOT (FN, FP, IDS, MOTA et MOTP) sont souvent considérées avec des métriques de qualité des trajectoires [75, 125]. À partir des appariements de résultats détaillés précédemment, on considère le pourcentage de cibles majoritairement suivies MT (*Mostly Tracked*) comme le taux de cibles qui sont associées à une hypothèse de trajectoire plus de 80% de leur temps de présence. Le pourcentage de cibles majoritairement perdues ML (*Mostly Lost*) est le taux de cibles associées à une hypothèse de trajectoire moins de 20% de leur temps de présence. Le nombre de fragmentations FM compte le nombre de fois où une cible passe d'un instant où elle est associée à une hypothèse de trajectoire à un instant où elle se trouve non associée (sans prendre en compte les identités des hypothèses

de trajectoires). Enfin, le nombre de fausses alarmes par image FAF (*False Alarm by Frame*) est le nombre moyen de faux positifs par image.

Pour résumer, l'ensemble des métriques usuellement considérées est donc :

— **FN↓ :** nombre de faux négatifs.
Varie entre 0 et $+\infty$.

— **FP↓ :** nombre de faux positifs.
Varie entre 0 et $+\infty$.

— **IDS↓ :** nombre de changements d'identité.
Varie entre 0 et $+\infty$.

— **MOTA↑ :** $1 - \frac{FN+FP+IDS}{P_{\mathcal{O}}}$.
Varie entre $-\infty$ et 1. [8]

— **MOTP↑ :** distance $d_{CLEARMOT}$ moyenne des appariements des résultats à la vérité terrain.
Varie entre $\frac{1}{2}$ et 1 pour une distance IOU avec un seuil d'appariement $th_{CLEARMOT}$ fixé à $\frac{1}{2}$ (cadre usuel en suivi 2D et qui est employé dans cette thèse).

— **MT↑ (%) :** taux de cibles estimées sur plus de 80% de leur temps de présence.
Varie entre 0 et 100, valeur exprimée en pourcentage.

— **ML↓ (%) :** taux de cibles estimées sur moins de 20% de leur temps de présence.
Varie entre 0 et 100, valeur exprimée en pourcentage.

— **FM↓ :** nombre de fragmentations.
Varie entre 0 et $+\infty$.

— **FAF↓ :** nombre moyen de faux positifs par image.
Varie entre 0 et $+\infty$.

Les symboles ↓ et ↑ indiquent si ces métriques diminuent ou augmentent lorsque les résultats sont considérés de meilleure qualité.

II.4.3 Considérations générales sur la comparaison des méthodes de suivi multi-objets

Bien que les bases de données et les métriques discutées précédemment permettent d'envisager des évaluations quantitatives des méthodes de suivi multi-objets, certaines difficultés demeurent pour obtenir des comparaisons pertinentes. Ainsi, certaines remarques peuvent être observées sur la pertinence des métriques proposées et sur les protocoles d'évaluation suivis.

Remarques sur les métriques usuellement employées

Plusieurs problèmes pratiques se posent lorsque les métriques CLEARMOT sont utilisées. En effet, l'association des résultats à la vérité terrain fait intervenir un problème d'appariement défini à partir d'une distance $d_{CLEARMOT}$ et d'un seuil $th_{CLEARMOT}$. Les implémentations de ces métriques varient en pratique sur la façon dont l'appariement est résolu (par algorithme hongrois ou algorithme glouton) et sur le seuil $th_{CLEARMOT}$ employé. De plus, les implémentations diffèrent aussi sur

8. Bien que des approches produisant un grand nombre de faux positifs ou de changements d'identité puissent obtenir des valeurs négatives en MOTA, les méthodes de suivi pertinentes présentent généralement des valeurs de MOTA positives (une approche de suivi n'estimant aucune cible ayant déjà un MOTA nul).

la façon d'interpréter les changements d'identité [75]. Ces différences rendent les résultats annoncés très dépendants de l'implémentation utilisée, et seuls les métriques calculées avec les mêmes codes sont au final réellement comparables.

Une autre difficulté est liée au choix de la vérité terrain employée. Sur les vidéos les plus fréquemment prises pour comparer les approches de suivi, comme les vidéos PETS S2L1 et PETS S2L2 [40], plusieurs vérités terrains existent. Il n'est alors pas évident de savoir quelle vérité terrain a été utilisée, et des différences significatives existent entre les différentes vérités terrains. Certaines peuvent par exemple uniquement considérer les cibles au centre de la scène étudiée, ce qui se traduit par des variations importantes au niveau des métriques obtenues. Des règles arbitraires, qui varient selon les vérités terrains, sont appliquées pour traiter le cas des cibles partiellement en dehors de l'image ou totalement occultées pour un très court instant, lors d'un croisement entre deux cibles par exemple. Une comparaison pertinente nécessite de ce fait d'employer la même implémentation pour calculer les métriques à partir des mêmes vérités terrains.

Néanmoins, une vérité terrain résulte d'un travail d'annotation le plus souvent réalisé manuellement. Si la présence ou non d'une cible dans l'image est un élément assez objectif, la localisation précise des boîtes autour des objets est beaucoup plus subjective. Ce travail d'annotation étant de plus très coûteux, ces annotations sont généralement faites sur un nombre restreint d'images de la vidéo puis étendues sur l'ensemble des images par interpolation linéaire. Cela rend en particulier la mesure de la qualité de localisation des cibles, MOTP, très dépendante du protocole d'annotation utilisé pour réaliser la vérité terrain.

Pour finir, la métrique MOTA est généralement considérée comme la métrique de référence pour classer les méthodes de suivi multi-objets, comme cela est fait par exemple pour le *MOTChallenge* [67, 86]. Cela est en partie justifié, car cette métrique prend en considération à la fois des erreurs de détection, liées aux nombres de faux négatifs FN et faux positifs FP, et d'association liées au nombre de changements d'identité IDS. Néanmoins, ce choix est parfois critiqué car le MOTA serait trop dépendant de la qualité du détecteur utilisé en amont de la méthode de suivi et ne permettrait pas d'évaluer suffisamment la qualité de l'algorithme de suivi indépendamment du détecteur. Le choix d'une métrique alternative est fréquemment discuté par la communauté travaillant sur le suivi multi-objets, comme par exemple lors du *MOTChallenge 2016 Workshop*, mais aucune alternative plus pertinente ne s'est pour l'instant imposée. On peut néanmoins noter que la base de données *KITTI* [45] ne considère pas le MOTA comme métrique principale et ne fait que reporter l'ensemble des métriques des différentes méthodes sans les classer.

Protocoles utilisés pour comparer les méthodes

D'autres difficultés surviennent pour comparer les méthodes de suivi multi-objets, qui ne sont pas liées aux métriques employées. Tout d'abord, un protocole usuel pour évaluer les approches de suivi multi-objets consiste à tester la méthode de suivi sur un ensemble de vidéos classiques (PETS, TownCenter...) en s'autorisant à ajuster les paramètres de la méthode pour chacune de ces vidéos. Cela mène à sur-adapter (*overtuning*) la méthode pour chaque vidéo spécifique alors qu'une évaluation plus pertinente serait de s'évaluer avec un même jeu de paramètres sur un ensemble le plus varié possible de vidéos. Adapter les paramètres pour chaque vidéo permet un gain significatif en performances comparé à la recherche d'un unique jeu

de paramètres pour l'ensemble des vidéos [91]. Il est donc délicat de comparer des méthodes ne suivant pas la même méthodologie à ce niveau.

En suivi par détection, l'algorithme de suivi prend en entrée des détections données par un détecteur d'objets qui est le plus souvent indépendant de l'approche de suivi. Séparer la partie détection de la partie de suivi dans les méthodes de suivi multi-objets n'est pas toujours faisable, notamment pour les approches [112] utilisant nécessairement des détections spécifiques en exploitant les parties d'un DPM (*Deformable Part Model*) [39] ou celles qui visent à effectuer conjointement la détection et le suivi des cibles [1, 71, 118]. Néanmoins, la grande majorité des approches de suivi multi-objets prennent en entrée des détections pouvant provenir de n'importe quel détecteur d'objets. Comme argumenté dans [91], il est alors préférable d'évaluer ces méthodes de suivi avec les mêmes jeux de détections afin de comparer les performances liées uniquement à l'algorithme de suivi.

Les bases de données *MOTChallenge* [67, 86], décrites précédemment en sous-section II.4.1, ont été proposées afin de fournir des outils d'évaluation qui prennent en compte les précédentes remarques. Dans les deux versions de cette base de donnée, 2015 et 2016, les vidéos sont réparties en un ensemble d'entraînement et de test afin d'éviter que les paramètres des approches soient déterminés directement sur les vidéos de test. Une même configuration de paramètres est supposée être trouvée sur les vidéos d'entraînement et employée pour les vidéos de test. De plus, des détections publiques sont fournies sur l'ensemble des vidéos et l'évaluation des résultats est faite en ligne, avec la même implémentation et la même vérité terrain pour calculer les métriques. Les bases de données du *MOTChallenge* permettent ainsi de comparer beaucoup plus aisément, et de façon plus juste, les méthodes de suivi multi-objets. On peut néanmoins remarquer que plusieurs approches récentes s'évaluent sur le *MOTChallenge* en adaptant certains paramètres à chaque vidéo de test, les détections étant par exemple fréquemment filtrées avec des seuils fixés manuellement pour chaque vidéo, ce qui limite légèrement la comparaison des différentes méthodes.

Méthodologie suivie dans cette thèse pour l'évaluation des résultats

Au cours de ces travaux de thèse, nous avons cherché à nous évaluer et à nous comparer de la façon la plus objective et équitable possible, que ce soit vis-à-vis des autres méthodes récentes de l'état de l'art ou vis-à-vis des variantes de nos approches. Nous avons suivi l'évolution des bases de données existantes, ce qui explique que les premiers travaux présentés au chapitre III ont été principalement évalués sur des vidéos usuelles [13,40,112] tandis que le reste des travaux, présenté au chapitre IV et au chapitre V, a été évalué avec les bases de données du *MOTChallenge*. Néanmoins, afin de permettre une comparaison de l'ensemble des méthodes proposées au cours de cette thèse, nous évaluons toutes les approches sur la version 2015 du *MOTChallenge* et présentons ces résultats au sein de la conclusion, au chapitre VI.

Concernant les travaux du chapitre III, ceux-ci sont évalués sur des vidéos usuelles en suivi de personnes [13,40,112] présentées en sous-section II.4.1. L'évaluation de nos approches utilise les seules métriques CLEARMOT avec l'implémentation de ces métriques de l'article [134]. Un unique jeu de paramètres est alors utilisé pour l'ensemble des vidéos. Nous nous comparons à d'autres méthodes récentes utilisant les mêmes jeux de détections publiques que nous avons employés, en calculant si possible les métriques à partir des trajectoires de résultats données par les auteurs de ces autres approches.

Pour les travaux présentés au chapitre IV et au chapitre V, nous utilisons les bases de données du *MOTChallenge*, à savoir la version 2015 *2DMOT2015* et 2016 *MOT16*. Nos méthodes utilisent principalement les détections publiques fournies par le *MOTChallenge*, sauf au chapitre V où des tests supplémentaires sont effectués avec des détections privées. Un unique jeu de paramètres est recherché à partir des vidéos d'entraînement et est directement employé pour les vidéos de test, aucun élément n'étant adapté pour chaque vidéo spécifique.

Afin de démontrer la pertinence des choix proposés au cours de cette thèse, nous cherchons aussi à évaluer et comparer plusieurs variantes de nos approches en modifiant certains modules. Pour que ces comparaisons soient réellement pertinentes, le jeu de paramètres employé par chaque variante devrait être déterminé indépendamment. Dans le chapitre III, nous avons limité cette recherche de paramètres en ne considérant que les paramètres dont la valeur idéale dépendait fortement des différentes variantes. La sélection de ces paramètres est alors effectuée exhaustivement sur une grille de recherche, comme détaillé en sous-section III.3.3. Au chapitre IV et au chapitre V, cette stratégie n'était plus envisageable du fait d'un nombre plus important de paramètres. Nous employons alors une recherche automatique des paramètres pour chaque variante par une procédure d'hyper-optimisation, comme détaillé en sous-section IV.4.1. Les différentes procédures décrites permettent alors d'effectuer des comparaisons en performance plus objectives des différentes variantes envisagées.

Conclusion

Nous avons présenté dans ce chapitre les approches existantes en suivi d'objets et les principales techniques associées. Le cas spécifique du suivi visuel d'objets a été davantage détaillé, et plus particulièrement le cas du suivi visuel multi-objets. Nous avons vu que de nombreuses méthodes de suivi visuel cherchent à tirer avantage de l'information visuelle disponible en définissant des modèles d'apparence afin de mieux localiser ou différencier les cibles. Une autre façon de gagner en performances est d'exploiter davantage d'information temporelle, ce qui est fait par les approches de suivi hors ligne.

Dans cette thèse, nous utilisons ces deux stratégies pour obtenir une méthode de suivi performante. Dans le chapitre III, nous cherchons à exploiter des représentations parcimonieuses dans une méthode de suivi en ligne. Les représentations parcimonieuses ayant été utilisées par de nombreuses approches de suivi mono-objet, il est naturel de chercher à étendre leur usage dans le cas du suivi multi-objets. Dans le chapitre IV, nous étudions comment ces représentations parcimonieuses peuvent être exploitées au mieux dans une approche de suivi à fenêtre glissante, afin de tirer avantage à la fois d'un modèle d'apparence sophistiqué et de l'information temporelle supplémentaire fournie par la fenêtre glissante. Enfin, dans le chapitre V, nous cherchons à exploiter des dictionnaires comportant un nombre plus conséquent d'éléments et qui présentent une certaine invariance en translation. En s'inspirant des techniques de représentations parcimonieuses à convolutions, des représentations parcimonieuses peuvent alors être calculées efficacement sur ces dictionnaires et être moins dépendantes de la qualité du détecteur d'objets employé dans notre approche de suivi.

Chapitre III

Suivi en ligne avec représentations parcimonieuses collaboratives

Sommaire

Introduction

Dans ce chapitre, une approche de suivi multi-objets en ligne, de type suivi par détection, est proposée de manière à exploiter des représentations parcimonieuses associées à chaque détection. Ces représentations parcimonieuses sont utilisées afin de définir des valeurs d'affinité plus discriminantes entre les trajectoires, avec pour objectif de limiter de ce fait les erreurs d'associations entre détections et cibles.

Tout d'abord, l'architecture générale de notre approche est présentée dans la section III.2. L'objectif est d'étudier l'impact de l'emploi d'affinités définies à partir

de représentations parcimonieuses pour le suivi multi-objets en ligne, et nous employons pour cela un système de suivi assez classique, fortement inspiré de travaux récents [134]. L'approche de suivi proposée se focalise sur le suivi de personnes multiples au sein de séquences vidéos monoculaires fixes, sans calibration de la caméra, avec une vue plongeante sur la scène (ce contexte étant rencontré fréquemment pour les applications de vidéosurveillance).

Le principe général que nous employons pour définir les valeurs d'affinité entre les trajectoires et les détections à partir de représentations parcimonieuses est précisé dans la section III.3. Plusieurs types de représentations, en particulier collaboratives entre cibles, sont étudiées pour déterminer lesquelles sont les plus performantes dans un contexte de suivi multi-objets. Les possibilités d'optimisation de ces représentations, à partir de méthodes de gradient proximal, sont aussi précisées.

La dernière section III.4 étend l'approche proposée au cas de descriptions locales des cibles. Plusieurs variantes pour prendre en compte des descriptions locales pour définir les valeurs d'affinité, en s'inspirant de certaines approches de suivi mono-objet, sont alors proposées et évaluées.

III.1 Motivations

L'emploi de représentations parcimonieuses dans une méthode de suivi multi-objets en ligne est motivé par les raisons explicitées dans ce qui suit.

III.1.1 Approches de suivi multi-objets en ligne

Les méthodes de suivi multi-objets en ligne, dont le principe général a été précisé au chapitre précédent, ont un avantage significatif par rapport aux autres types de méthodes de suivi. En effet, contrairement aux méthodes de suivi hors ligne, seules les méthodes en ligne permettent de traiter immédiatement chaque image de la vidéo considérée et peuvent ainsi donner une réponse sans attendre d'autres images futures. En pratique, cela signifie notamment que ces méthodes ont potentiellement un temps de latence très faible puisque leur temps de réponse dépend uniquement du temps nécessaire pour traiter l'information de la dernière image. Ce délai de réponse, ou temps de latence, est donc principalement dépendant du coût CPU de l'approche et de la puissance de calcul disponible. Les approches hors ligne, à fenêtre glissante ou globales, présentent un temps de latence qui dépend lui-aussi de ces éléments mais aussi du délai futur pris en compte pour traiter l'instant courant. Ainsi, seules les approches en ligne peuvent avoir un temps de réponse réellement négligeable, sous réserve d'une puissance de calcul suffisante. Cette caractéristique est particulièrement appréciable pour les applications temps réels ou celles nécessitant une réponse suffisamment rapide.

De plus, un grand nombre d'approches de suivi en ligne ont été récemment proposées, par exemple [6, 94, 112, 126, 134], et leurs performances sont assez comparables par rapport aux méthodes hors ligne. Cependant, ces bonnes performances sont principalement dues à l'emploi de modèles d'apparence, voire de mouvement, plus complexes des cibles. Ces modèles vont alors permettre d'attribuer des valeurs d'affinité pertinentes entre les trajectoires et les nouvelles détections, et ainsi robustifier le processus d'association de ces données.

Puisque nous nous intéressons au sein de cette thèse à des méthodes de suivi multi-objets en ayant pour contrainte d'avoir un temps de latence faible, envisager une approche en ligne est un choix assez naturel vis-à-vis de ce critère. Néanmoins, cela signifie que la performance de notre méthode va être largement dépendante de la pertinence des valeurs d'affinité estimées entre les trajectoires et les dernières détections. C'est pour cette raison que nous allons envisager d'employer des représentations parcimonieuses pour définir des valeurs d'affinité performantes.

III.1.2 Représentations parcimonieuses collaboratives

Les représentations parcimonieuses ont été largement employées au cours des dernières années dans de nombreux domaines en Vision par Ordinateur et il est particulièrement intéressant de voir comment elles ont été employées dans deux domaines plus spécifiques, à savoir le suivi mono-objet et la classification multi-classes. Le lecteur peut se référer au chapitre II qui présente, de façon générale, le principe de ces méthodes. Nous résumons ici les principales observations qui motivent l'étude de représentations parcimonieuses collaboratives pour le suivi multi-objets.

Les représentations parcimonieuses ont été employées dans le domaine de la classification multi-classes, notamment pour des applications de reconnaissance faciale [124]. L'idée principale de ces approches repose sur l'exploitation de représentations collaboratives entre individus. Un dictionnaire commun, composé d'éléments correspondants à plusieurs individus, est employé. Chaque nouvelle personne à reconnaître est représentée de manière parcimonieuse avec les éléments de ce dictionnaire, c'est-à-dire comme une combinaison linéaire pondérée d'un faible nombre de ces éléments. Le principe de base de ce genre d'approches est que les éléments participant le plus dans la représentation de la requête sont alors censés correspondre à des vues du même individu. Il a été argumenté dans [124] que l'emploi de représentations parcimonieuses collaboratives aide à différencier des classes dont la variance inter-classes est faible, ce qui est effectivement le cas en reconnaissance faciale.

En suivi mono-objet, les représentations parcimonieuses ont initialement été employées de façon à modéliser la cible de manière générative, le dictionnaire étant principalement constitué de différentes vues de la cible [85]. Des représentations collaboratives ont ensuite été proposées afin d'obtenir un modèle discriminatif entre la cible et son voisinage proche, le dictionnaire utilisé étant cette fois composé des éléments de ces deux différentes classes [140].

En suivi multi-objets, avec le paradigme de suivi par détection, une difficulté importante se situe au niveau de l'association de données entre les détections et les cibles. Ce problème revient à déterminer pour chaque détection quelle est la cible correspondante. Il s'apparente à un problème de classification multi-classes où chaque classe modéliserait une cible. Les différentes cibles étant le plus souvent des instances d'une même classe, toutes les cibles sont d'apparence proche. On retrouve les caractéristiques énoncées plus haut en reconnaissance faciale ou en ré-identification de personnes. Cela motive l'usage de représentations parcimonieuses collaboratives entre les cibles pour réaliser l'étape de l'association de données.

Bien que de nombreuses méthodes de suivi mono-objet aient utilisé des représentations parcimonieuses [135], peu de méthodes multi-objets ont exploité ces représentations. De plus, les rares approches qui utilisaient ces représentations parcimonieuses au démarrage de cette thèse, comme par exemple [94], se limitaient à

employer des modèles d'apparence directement inspirés de méthodes mono-objet en attribuant un modèle par cible et sans exploiter de représentations collaboratives entre ces cibles. Ce constat a motivé nos travaux sur l'usage de représentations collaboratives entre cibles pour le suivi multi-objets.

III.1.3 Principe de l'approche proposée

Notre approche vise donc à exploiter des représentations parcimonieuses, notamment collaboratives entre cibles, dans le cadre d'une méthode de suivi en ligne multi-objets. Plusieurs choix sont possibles pour définir des représentations parcimonieuses entre cibles, surtout par rapport au type de dictionnaire commun employé et au type de description des cibles. Nous étudierons plusieurs des variantes qui en découlent pour évaluer lesquelles sont les plus pertinentes pour une approche de suivi multi-objets.

Le fait de choisir une approche de suivi en ligne, dont le principal intérêt est de limiter le temps de réponse de la méthode proposée, nécessite de porter une attention particulière au calcul des représentations parcimonieuses que nous utiliserons. En effet, déterminer une représentation parcimonieuse est en général une tâche assez coûteuse en temps de calcul et nous envisagerons donc plusieurs variantes d'optimisation pour réduire ce coût de calcul et limiter au maximum la latence de notre approche.

III.2 Système de suivi multi-objets en ligne employé

Cette section détaille l'architecture générale du système de suivi multi-objets en ligne employé par notre approche. Les valeurs d'affinité entre les trajectoires et détections, définies à partir de représentations parcimonieuses, sont explicitées dans la section III.3 suivante.

III.2.1 Description générale du système

Comme expliqué dans la section précédente, nous étudions l'apport que peut avoir l'emploi de représentations parcimonieuses des cibles au sein d'un système de suivi en ligne. Notre approche, à l'exception de la partie concernant le calcul des valeurs d'affinité entre trajectoires et détections, est fortement inspirée de l'article [134] et se base sur une architecture générale fréquemment employée par les méthodes de suivi en ligne récentes.

Le système de suivi proposé exploite le paradigme de suivi-par-détection. Notre approche étant en ligne, les images de la vidéo sont considérées les unes après les autres et l'image courante est traitée en exploitant uniquement les résultats de l'image précédente (les états des trajectoires sont ainsi estimés par une inférence markovienne d'ordre 1). Un détecteur d'objets donne alors à chaque image un ensemble de détections qui constituent des hypothèses sur la position des cibles. Le principe général est ensuite d'associer à chaque image les détections aux trajectoires déjà estimées et d'utiliser cette association de données pour prolonger les trajectoires (et éventuellement en créer ou en terminer). Notre algorithme est ainsi constitué de

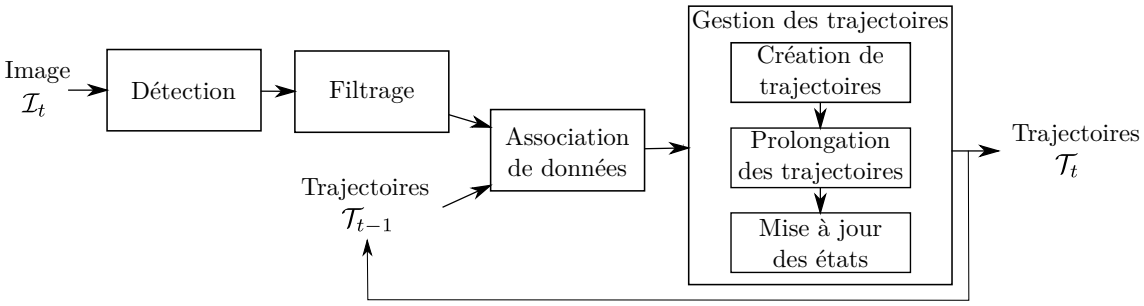


FIGURE III.1 – Synoptique de notre système de suivi par détection (suivi multi-objets en ligne).

différents modules qui sont utilisés pour traiter chaque nouvelle image, comme décrit en figure III.1. Ces différents modules sont composés de :

(i) un détecteur d'objets, qui détermine un ensemble de détection Det_{I_t} à partir de l'image courante I_t.

(ii) un module de filtrage des détections qui a pour rôle d'éliminer certaines fausses détections (faux positifs) de Det_{I_t} en s'adaptant à la scène traitée.

(iii) un module d'association de données dont l'objectif est, à partir des détections Det_{I_t} et des trajectoires estimées précédemment $\mathcal{T}_{t-1}$, d'associer chaque détection à la trajectoire qui lui correspond.

(iv) un module de gestion des trajectoires composé de trois sous-modules, qui permet à la fois de prolonger les trajectoires à l'instant t et d'estimer les nouveaux états des trajectoires, pour en décréter certaines terminées par exemple.

Une fois que tous ces modules ont été utilisés, un nouvel ensemble de trajectoires $\mathcal{T}_t$ est estimé pour l'instant t et sera utilisé avec la prochaine image I_{t+1} pour estimer les trajectoires $\mathcal{T}_{t+1}$ à l'instant suivant.

Ces différents modules sont détaillés ci-après, à l'exception du détecteur d'objets initial et de l'étape de filtrage des détections. En effet, notre méthode n'est pas spécifique à un seul détecteur d'objets et sera en pratique évaluée avec plusieurs détecteurs de personnes différents. Le filtrage des détections repose sur la méthode proposée dans l'article [134] et ne sera donc pas détaillé. Son principe est de séparer la scène dans le repère image à partir d'une grille régulière et d'estimer les tailles moyennes des détections des cibles dans chacune de ces zones pour rejeter les détections de taille anormale. Ce module remplace une éventuelle calibration de la caméra, qui permettrait d'effectuer un filtrage plus précis, et nécessite une vue plongeante sur la scène.

III.2.2 Formulation de l'association de données

Le module d'association de données a pour rôle d'associer les trajectoires estimées $\mathcal{T}_t = \{T_1, ..., T_{n_{traj}}\}$ aux détections $Det_{I_t} = \{d_1, ..., d_{n_{det}}\}$ détectées sur l'image I_t reçue à l'instant courant t. Par simplicité d'écriture, nous noterons ici $\mathcal{T} = \mathcal{T}_t$ et $Det = Det_{I_t}$ sauf dans les cas où cela mènerait à des ambiguïtés. Les méthodes de suivi en ligne traitent le plus souvent cette tâche en modélisant cette association de données sous la forme d'un problème d'appariement de poids maximal dans un graphe biparti complet, comme mentionné au chapitre II. Néanmoins, certaines

considérations doivent être prises en compte de manière à formuler correctement cette étape d'association de données comme un problème d'appariement de poids maximal dans un graphe biparti complet. En particulier, l'appariement ne sera pas formulé directement entre les éléments des ensembles $\mathcal{T}$ et Det afin de modéliser certaines situations spécifiques, comme des cibles non détectées ou bien des fausses détections par exemple.

Modélisation sous forme d'un problème d'appariement

Étant donné un ensemble de trajectoires $\mathcal{T} = \{T_1, ..., T_{n_{traj}}\}$ et de nouvelles détections $Det = \{d_1, ..., d_{n_{det}}\}$, le problème d'association de données consiste à apparier chaque détection de l'ensemble Det à la trajectoire de $\mathcal{T}$ qui lui correspond. Une stratégie basique consiste alors à attribuer une valeur d'affinité $Aff(T, d)$ à chaque couple de trajectoire-détection (T, d) afin de favoriser les appariements de couples de forte affinité. Chaque détection est alors supposée être associée à au plus une trajectoire, et inversement chaque trajectoire est supposée être associée à au plus une détection.

Il est alors possible de modéliser ce problème sous la forme d'un problème d'appariement maximal de poids maximal dans un graphe biparti complet. Ce graphe est construit de telle sorte que ses noeuds soient répartis en deux ensembles, l'un correspondant aux trajectoires de $\mathcal{T}$ et l'autre aux détections de Det. Chaque trajectoire T est reliée à chaque détection d par une arête de poids $Aff(T, d)$. On cherche alors un appariement maximal sur ce graphe, c'est-à-dire à sélectionner un ensemble maximal d'arêtes de façon à ce que tout noeud ne soit relié au plus qu'à une seule arête de cet ensemble. Parmi ces appariements maximaux possibles $\mathcal{E}$, un appariement maximal de poids maximal C^* est alors un élément de $\mathcal{E}$ dont la somme des poids des arêtes est maximale, c'est-à-dire une solution de :

$$\max_{\mathcal{C} \in \mathcal{E}} \sum_{(T,d) \in \mathcal{C}} Aff(T, d). \tag{III.1}$$

Cette formalisation ne prend néanmoins pas en compte les aspects de non détection et de fausses détections, et force à associer un maximum de trajectoires et de détections sans tenir compte de ces éventuelles erreurs de détection. Afin de prendre en compte ces éléments, il est possible de considérer n_{traj} détections virtuelles $Det_{virt} = \{d'_1, ..., d'_{n_{traj}}\}$ et n_{det} trajectoires virtuelles $\mathcal{T}_{virt} = \{T'_1, ..., T'_{n_{det}}\}$. L'appariement maximal de poids maximal est alors déterminé entre les éléments de $\mathcal{T} \bigcup \mathcal{T}_{virt}$ et $Det \bigcup Det_{virt}$ et, comme $|\mathcal{T} \bigcup \mathcal{T}_{virt}| = |Det \bigcup Det_{virt}|$, cet appariement est parfait (toute trajectoire est associée à une unique détection et inversement). Chaque trajectoire associée à une détection virtuelle est alors considérée non détectée, et de même toute détection associée à une trajectoire virtuelle est considérée comme une fausse détection.

Il reste à définir les affinités qui font intervenir des trajectoires virtuelles ou des détections virtuelles. Une stratégie basique, présentée par exemple dans [6], consiste à attribuer une affinité constante, de valeur θ, à tout couple (T, d) lorsque $T \in \mathcal{T}_{virt}$ ou $d \in Det_{virt}$. Cela a pour conséquence, en pratique, d'interdire l'appariement d'une trajectoire T à une détection d si $Aff(T, d) < \theta$. En effet, il est toujours possible d'obtenir un meilleur appariement dans ce cas, comme illustré en figure III.2. La valeur θ joue donc le rôle d'un seuil limite sur les affinités des trajectoires et détections qui peuvent être associées. Des stratégies plus élaborées, qui ne sont pas

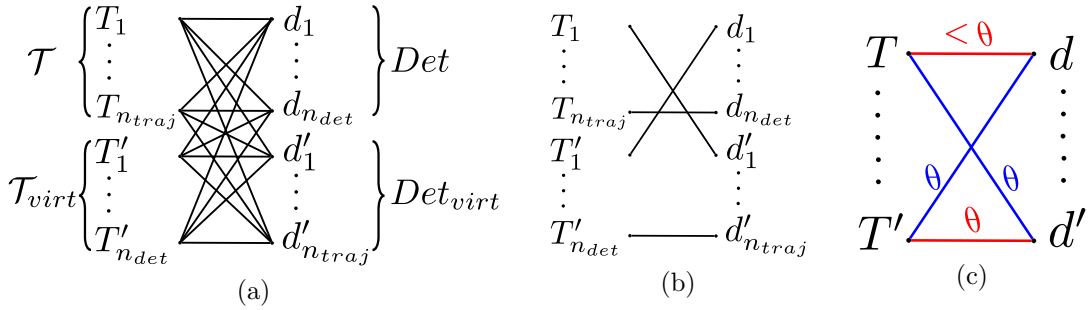


FIGURE III.2 – Modélisation sous la forme d'un appariement maximal dans un graphe biparti complet. (a) : Graphe biparti considéré. (b) : Appariement admissible. (c) : Impact de θ. Si $Aff(T, d) < \theta$, il est préférable et possible d'associer T et d à une trajectoire virtuelle T' et détection virtuelle d' (à visualiser de préférence en couleurs).

envisagées dans notre approche, n'attribuent pas une valeur d'affinité θ constante pour toutes les situations de cible non détectée ou de fausses détections.

Une solution optimale du problème d'appariement dans un graphe biparti complet peut être obtenue en appliquant l'algorithme hongrois ou de Munkres [93] qui présente une complexité en $O(n^3)$ avec ici $n = |\mathcal{T} \bigcup \mathcal{T}_{virt}| = |Det \bigcup Det_{virt}|$. Une solution sous-optimale peut néanmoins être obtenue plus rapidement avec un algorithme glouton en $O(n^2 log(n))$. Cet algorithme détermine la solution $\mathcal{C}$ itérativement, en ajoutant à chaque étape la paire de noeuds (a, b), avec a et b non inclus dans une autre paire de $\mathcal{C}$, de poids $w(a, b)$ maximal. Ces deux algorithmes sont fréquemment employés en suivi multi-objets pour résoudre le problème d'association de données, un algorithme glouton étant par exemple utilisé dans [18, 112]. On peut noter que, même dans le cas d'une solution approchée donnée par un algorithme glouton, le paramètre θ joue toujours un rôle de seuil limite en interdisant les associations (T, d) si $Aff(T, d) < \theta$.

Prise en compte d'une hiérarchie au niveau des trajectoires

Plusieurs approches de suivi en ligne [6, 134] ne réalisent pas directement l'appariement de toutes les trajectoires estimées $\mathcal{T}$ aux détections Det et utilisent plusieurs appariement successifs de façon à associer en priorité certaines trajectoires. L'idée générale est de chercher à associer en priorité les trajectoires jugées les plus fiables aux détections. Cela évite en particulier, dans le cas de détections multiples pour une même cible, que plusieurs trajectoires se concurrencent les unes les autres pour l'association en produisant un nombre important de changements d'identités. Le principe est de répartir les trajectoires de $\mathcal{T}$ en deux groupes $\mathcal{T}_{expertes}$ et $\mathcal{T}_{standards}$, l'ensemble $\mathcal{T}_{expertes}$ regroupant des trajectoires jugées fiables, appelées expertes, et $\mathcal{T}_{standards}$ regroupant le reste des trajectoires. On établit ainsi une hiérarchie des trajectoires qui va être exploitée pour favoriser les trajectoires expertes au cours de l'appariement. Bien qu'une hiérarchie plus élaborée puisse être envisagée, avec plusieurs groupes de différents niveaux de fiabilité pour les trajectoires, les approches de suivi récentes qui utilisent une hiérarchie pour les trajectoires se limitent à une séparation en deux groupes.

Cette hiérarchie est prise en compte de la manière suivante. Un premier appariement est réalisé entre les trajectoires expertes de $\mathcal{T}_{expertes}$ et les détections Det.

Cet appariement est effectué en suivant la modélisation expliquée précédemment, c'est-à-dire en considérant des trajectoires virtuelles $\mathcal{T}_{virt}$ et détections virtuelles Det_{virt} pour modéliser les cibles non détectées et les fausses détections. Une fois cet appariement effectué, un second appariement est réalisé entre les trajectoires non expertes $\mathcal{T}_{standards}$ et les détections restantes non associées à une trajectoire experte.

Avec cette stratégie, les trajectoires expertes sont favorisées lors de l'association de données et sont ainsi moins susceptibles d'être interrompues au profit d'une nouvelle trajectoire. Il est cependant nécessaire de définir avec quel critère les trajectoires sont déclarées expertes. Cela est fait dans notre approche en considérant la fréquence d'appariement des trajectoires afin de favoriser les trajectoires qui sont le plus souvent associées à des détections. Ce critère sera détaillé ultérieurement en sous-section III.2.3.

Affinités considérées

Les affinités considérées $Aff(T, d)$ pour chaque couple trajectoire-détection (T, d) vont prendre la forme suivante :

$$Aff(T,d) = \begin{cases} a(T,d) & si \ \ (T,d) \in \mathcal{L} \\ -\infty & sinon \end{cases} . \tag{III.2}$$

L'ensemble $\mathcal{L}$ rassemble tous les couples trajectoire-détection (T, d) dont la trajectoire T et la détection d sont suffisamment proches spatialement dans l'image ainsi qu'au niveau de leurs tailles respectives pour pouvoir considérer un appariement entre T et d. Cet ensemble $\mathcal{L}$ permet d'exclure des associations qui sont incompatibles avec un déplacement faible des cibles entre deux images, ce qui simplifie le problème d'association de données et aide à obtenir des résultats plus stables. Le terme $a(T, d)$ sera alors la valeur d'affinité pour les couples trajectoires-détections $(T, d) \in \mathcal{L}$ dont l'appariement peut être envisagé. La valeur $a(T, d)$ peut typiquement faire intervenir des considérations sur l'apparence et la dynamique de la cible liée à la trajectoire T. En pratique, les approches proposées dans cette partie se limiteront à des considérations sur les apparences des détections pour le terme $a(T, d)$.

L'ensemble $\mathcal{L}$ est défini par :

$$\mathcal{L} = \{(T,d), \ \ \frac{dist(T,d)}{w_T} < th_{dist}(T) \ \ et \ \ \frac{|h_T - h_d|}{h_T} < th_{taille}(T)\}, \tag{III.3}$$

où $dist(T, d)$ est la distance Euclidienne entre la dernière position estimée de la trajectoire T et le centre de la détection d. Les valeurs w_T et h_T correspondent respectivement à la largeur et hauteur de la dernière boîte estimée de T, tandis que h_d est la hauteur de la détection d. Enfin, $th_{dist}(T)$ et $th_{taille}(T)$ dépendent de la trajectoire T et augmentent plus la fréquence d'appariement de T est faible, ceci afin de permettre de considérer l'appariement de détections plus éloignées lorsque la trajectoire T se trouve non associée. Un tel mécanisme permet à des trajectoires non associées de récupérer ultérieurement des détections de leur cible. L'ensemble $\mathcal{L}$ est ainsi composé des couples (T, d) pour lesquels la détection d est incluse dans une zone de recherche spécifique à T et définie à partir des seuils $th_{dist}(T)$ et $th_{taille}(T)$.

Plus formellement, $th_{dist}(T)$ vérifie la relation :

$$th_{dist}(T) = \alpha_{dist} + \frac{\beta_{dist}}{FR}(1 - freq_{\Delta_t}(T)) . \tag{III.4}$$

Dans cette équation, FR est la fréquence de la vidéo traitée, $freq_{\Delta_t}(T)$ est la fréquence d'appariement de T (i.e. le ratio du nombre d'images pour lesquelles T est appariée durant la période Δt précédant l'instant considéré), et α_{dist} et β_{dist} sont des paramètres fixes. Le seuil $th_{taille}(T)$ est déterminé par une relation similaire.

Lorsque $(T, d) \notin \mathcal{L}$, ce qui traduit que d n'est pas dans la zone de recherche associée à T, la valeur d'affinité infinie $Aff(T, d) = -\infty$ indique que l'appariement entre T et d est impossible. En effet, comme vu précédemment, il sera toujours préférable, puisque $Aff(T, d) < \theta$, de considérer T comme non détectée et d comme une fausse détection (ces deux événements étant associés à une valeur d'affinité θ). En pratique, il est cependant délicat de traiter le problème d'appariement dans un graphe biparti avec des valeurs infinies, en particulier dans le cas de l'emploi de l'algorithme hongrois. Il suffit alors de remplacer cette valeur $-\infty$ par une valeur strictement inférieure à θ pour éviter la gestion de valeurs infinies sans changer la solution générale du problème d'appariement.

III.2.3 Gestion des trajectoires

Nous détaillons ici la façon dont les trajectoires sont gérées au sein de notre méthode, en explicitant en particulier comment ces dernières sont créées et prolongées à partir des détections qui leur sont associées. Les différents états qui peuvent être attribués aux trajectoires sont aussi précisés.

Création de trajectoires

Après l'étape d'association de données, toute détection non associée à une trajectoire entraîne la création d'une nouvelle trajectoire associée à cette détection. Comme les détections non associées sont fréquemment des fausses détections, ces nouvelles trajectoires ne sont que des hypothèses de réelles trajectoires et sont considérées non confirmées tant que leur fiabilité n'est pas jugée suffisante. Ces nouvelles trajectoires n'interviennent alors pas encore dans les résultats de suivi mais peuvent ultérieurement être considérées comme des trajectoires usuelles. Le critère le plus simple pour valider une trajectoire dans les méthodes de suivi en ligne est d'attendre qu'un nombre suffisant de détections lui aient été associées.

Prolongement des trajectoires

Une fois l'étape d'association de données effectuée, à l'instant courant t, il est nécessaire d'estimer les positions des trajectoires sur l'image I_t en exploitant les appariements des détections qui ont été effectués. Étant donnée une trajectoire T, deux cas de figures peuvent survenir : soit T a été associée à une détection d, soit la cible liée à T a été considérée non détectée. Ainsi, il faut dans le premier cas estimer la nouvelle position x_t de T en prenant en compte la nouvelle observation y_t liée à d et les observations précédente $(y_k)_{k<t}$ qui résultent d'anciens appariements, ce qui se traduit par une étape de filtrage. Dans le second cas, aucune observation y_t n'est disponible et x_t doit alors être prédite uniquement à partir des observations précédentes $(y_k)_{k<t}$. Ces deux tâches se ramènent donc à effectuer un filtrage ou une prédiction selon le cas de figure.

Nous reprenons l'approche proposée dans [134] pour effectuer ces étapes de prédiction ou de filtrage. Un filtre de Kalman linéaire est utilisé pour déterminer la

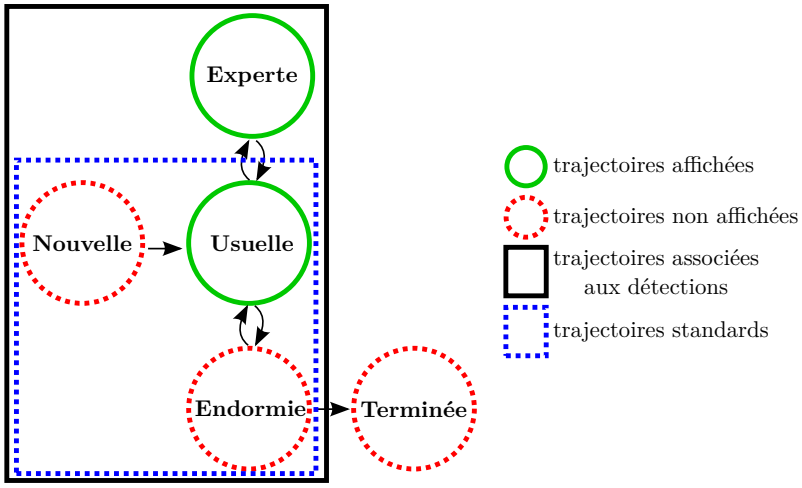


FIGURE III.3 – Groupes de trajectoires définis à partir de leurs états. Les trajectoires non affichées ne sont pas présentes dans les résultats de l'algorithme. Toutes les trajectoires sont considérées dans l'association des données à l'exception des trajectoires terminées. Au cours de cette association, les trajectoires standards $\mathcal{T}_{standards}$ sont constituées des trajectoires usuelles, nouvelles et endormies.

position de la cible, liée à T, en employant un modèle de dynamique de vélocité constante. La prédiction est alors réalisée en utilisant ce filtre de Kalman pour prédire la position de la cible en supposant sa taille constante. Dans le cas d'une étape de filtrage, cette étape est effectuée par une prédiction suivie d'une mise à jour (ou correction) du filtre de Kalman avec l'observation y_t. La taille de la cible est alors estimée en combinant sa taille précédente (à l'état x_{t-1}) et celle de l'observation y_t via un facteur d'innovation α.

Etats associés

Les trajectoires estimées peuvent prendre différents états tout au long du suivi. Nous utilisons cinq états qui peuvent être assignés aux trajectoires dans notre système :

(i) **Nouvelle :** état attribué aux trajectoires venant d'être créées.

(ii) **Usuelle :** état désignant les trajectoires de fiabilité normale.

(iii) **Experte :** état assigné aux trajectoires de forte fiabilité.

(iv) **Endormie :** état attribué aux trajectoires de faible fiabilité.

(v) **Terminée :** état donné aux trajectoires définitivement arrêtées.

Ces catégories permettent de différencier les trajectoires qui participent réellement aux résultats et sont affichées en sortie de l'algorithme. Cela permet d'exclure des résultats les trajectoires nouvelles et terminées, ainsi que des trajectoires dites endormies jugées peu fiables mais pas encore terminées. Toutes les trajectoires, à l'exception des trajectoires terminées, sont considérées lors de l'étape d'association de données. Les trajectoires standards de $\mathcal{T}_{standards}$, associées après les trajectoires expertes, regroupent alors les trajectoires nouvelles, usuelles et endormies. Ces différents états et groupes de trajectoires sont illustrés en figure III.3.

Toute trajectoire peut voir son état évoluer entre ces cinq états en parcourant l'automate indiqué en figure III.4. Les transitions entre ces états font intervenir

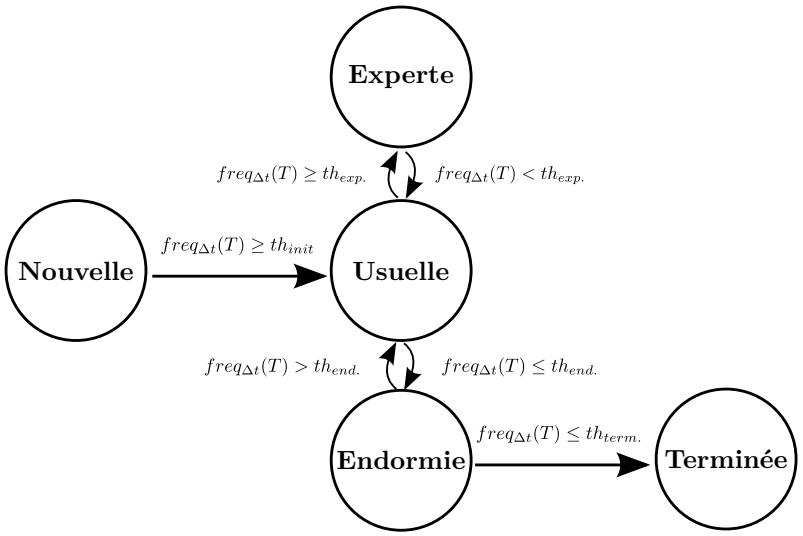


FIGURE III.4 – Automate des états attribués aux trajectoires en fonction de leur fréquence d'appariement $freq_{\Delta t}(T)$.

une évaluation de la fiabilité de la trajectoire, et cette fiabilité est ici évaluée en considérant la fréquence d'appariement $freq_{\Delta t}(T)$ de la trajectoire sur une période Δt précédant l'instant courant. Les seuils limites $th_{exp.}$, $th_{end.}$ et $th_{term.}$ sont définis à partir du seuil th_{init} et d'un facteur multiplicatif constant (on peut par exemple envisager $th_{end.} = \frac{1}{2}th_{init}$ et $th_{term.} = \frac{1}{4}th_{init}$). Le seuil th_{init} est quant à lui déterminé à partir de la moyenne μ_{freq} et de l'écart-type σ_{freq} des fréquences d'appariement des trajectoires observées jusqu'à l'instant courant, via la relation :

$$th_{init} = \alpha_{init}\mu_{freq} + \beta_{init}\sigma_{freq}, \tag{III.5}$$

où α_{init} et β_{init} sont deux paramètres constants.

III.3 Affinités à partir de représentations parcimonieuses collaboratives

Dans cette section, nous expliquons comment les valeurs d'affinité entre les trajectoires et détections peuvent être définies à partir de représentations parcimonieuses collaboratives. Si la section précédente introduisait un système de suivi en ligne assez usuel, chercher à exploiter des représentations parcimonieuses collaboratives dans le cadre d'un suivi multi-objets constitue l'originalité de notre première contribution.

III.3.1 Principe général et types de représentations envisagés

Pour rappel, notre algorithme de suivi repose sur un appariement biparti entre les trajectoires estimées et les nouvelles détections à la dernière image considérée. La valeur d'affinité pour chaque couple trajectoire-détection (T, d) est définie par :

$$Aff(T,d) = \begin{cases} a(T,d) & si \ \ (T,d) \in \mathcal{L} \\ -\infty & sinon \end{cases}, \tag{III.6}$$

où $\mathcal{L}$ est un ensemble constitué de toutes les couples (T, d) de trajectoires et détections dont l'association peut être considérée, basé sur un critère de proximité spatiale dans l'image ainsi que vis-à-vis de leurs tailles respectives. Nous proposons dans cette section de définir le terme $a(T, d)$ à partir d'une représentation parcimonieuse de la détection d.

Nous introduisons maintenant quelques notations relatives aux représentations parcimonieuses utilisées dans la suite. La notion de représentation parcimonieuse a été introduite plus précisément dans le chapitre précédent, et nous invitons le lecteur à se référer à la section II.3 pour plus de détails.

Notations pour les représentations parcimonieuses

Pour rappel, l'ensemble des trajectoires estimées est noté $\mathcal{T} = \{T_1, ..., T_{n_{traj}}\}$ et $Det = \{d_1, ..., d_{n_{det}}\}$ correspond à l'ensemble des détections données par le détecteur d'objets sur l'image courante. Chaque détection d est alors décrite par un vecteur de caractéristiques $y_d \in \mathbb{R}^m$ qui est un vecteur dépendant de l'apparence de la détection d, bien que des informations contextuelles ou de mouvement puissent être aussi considérées (par exemple avec des HOF [29]). Toute trajectoire T est associée à un dictionnaire spécifique $D_T \in \mathbb{R}^{n_{dict} \times m}$ constitué des caractéristiques des plus récentes détections (au plus n_{dict}) associées à la trajectoire T. Pour tout ensemble de trajectoires $\mathcal{S} = \{T_{i_1}, .., T_{i_l}\}$, on considère le dictionnaire commun $D_{\mathcal{S}} = [D_{T_{i_1}} ... D_{T_{i_l}}]$ obtenu en concaténant l'ensemble des colonnes des dictionnaires spécifiques.

Étant donné un dictionnaire D, on associe à chaque caractéristique y_d une représentation parcimonieuse $\alpha^D_{y_d}$ définie par :

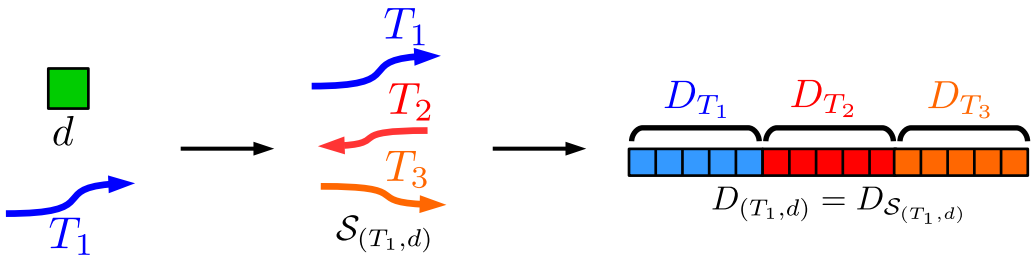
$$\alpha^D_{y_d} = \arg\min_{\alpha} \frac{1}{2}||y_d - D\alpha||_2^2 + \lambda||\alpha||_1. \tag{III.7}$$

L'erreur de reconstruction, associée à y_d et à la représentation $\alpha^D_{y_d}$, est alors définie par $\frac{1}{2}||y_d - D\alpha^D_{y_d}||_2^2$. Pour tout ensemble d'indices I d'éléments (ou colonnes) de D, on note $\delta_I(\alpha)$ le vecteur obtenu à partir du vecteur α en remplaçant par zéro tous les coefficients des indices non inclus dans I : $(\delta_I(\alpha))_i = \alpha_i$ si $i \in I$ et $(\delta_I(\alpha))_i = 0$ si $i \notin I$. L'erreur résiduelle, pour l'ensemble I associé à y_d et à la représentation $\alpha^D_{y_d}$, est alors définie par $\frac{1}{2}||y_d - D\delta_I(\alpha^D_{y_d})||_2^2$.

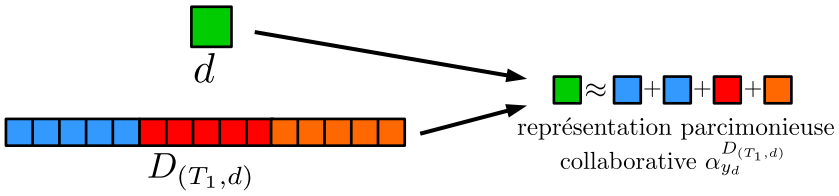
Principe général

L'approche proposée consiste, pour tout couple trajectoire-détection (T, d) considéré, à représenter la détection d à partir de détections provenant d'un ensemble de trajectoires $\mathcal{S}_{(T,d)}$. Cette représentation s'effectue au niveau des caractéristiques y_d des détections mises en jeu et prend la forme d'une représentation parcimonieuse afin de ne faire intervenir que peu de détections. À partir de cette représentation, il est alors possible d'examiner dans quelle proportion la détection d est représentée par les seules détections associées à la trajectoire T. On considère pour cela l'erreur de reconstruction résiduelle obtenue en ne considérant que les éléments de T, qui sera d'autant plus faible que les éléments de T participent à la représentation de d, et on définit la valeur d'affinité entre T et d à partir de la valeur de cette erreur de reconstruction résiduelle. Ce fonctionnement général est illustré par la figure III.5.

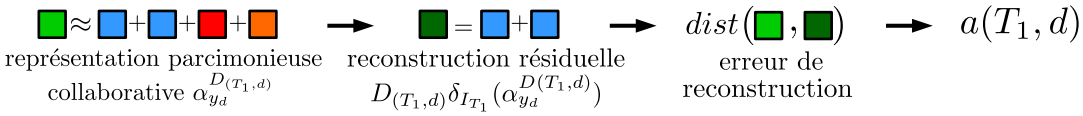
Nous détaillons maintenant plus formellement comment définir la valeur d'affinité $a(T, d)$ pour un couple trajectoire-détection (T, d) avec $(T, d) \in \mathcal{L}$. Tout d'abord,



1. Dictionnaire collaboratif considéré



2. Représentation parcimonieuse collaborative



3. Valeur d'affinité finale définie à partir d'une erreur de reconstruction résiduelle

FIGURE III.5 – Principe général de notre approche pour définir des affinités, entre détections et trajectoires, à partir de représentations collaboratives.

un dictionnaire $D_{(T,d)}$ spécifique au couple (T, d) est considéré. Ce dictionnaire est alors supposé être associé à un ensemble de trajectoires $\mathcal{S}_{(T,d)}$. Cela signifie que, si $\mathcal{S}_{(T,d)} = \{T_{i_1}...T_{i_l}\}$, $D_{\mathcal{S}_{(T,d)}} = [D_{T_{i_1}}...D_{T_{i_l}}]$. Nous supposons de plus que $T \in \mathcal{S}_{(T,d)}$.

À partir du dictionnaire $D_{(T,d)}$, nous considérons la représentation parcimonieuse $\alpha_{y_d}^{D_{(T,d)}}$ associée à la détection d et définie par l'équation (III.7). Nous considérons maintenant l'ensemble I_T qui correspond aux indices des éléments de $D_{(T,d)}$ associés à la trajectoire T. La valeur d'affinité entre T et d est alors définie comme l'opposé de l'erreur résiduelle de la représentation $\alpha_{y_d}^{D_{(T,d)}}$ vis-à-vis des éléments de la trajectoire T, par la formule :

$$a(T,d) = -\frac{1}{2}||y_d - D_{(T,d)}\delta_{I_T}(\alpha_{y_d}^{D_{(T,d)}})||_2^2. \tag{III.8}$$

Dans le cas où le dictionnaire $D_{(T,d)}$ ne comporte que les éléments de la trajectoire T, c'est-à-dire $\mathcal{S}_{(T,d)} = \{T\}$, l'erreur de reconstruction traduit à quelle point d peut être reconstruite correctement à partir de peu d'éléments de la trajectoire T. Cela peut être assimilé à une approche générative où la classe considérée serait constituée des caractéristiques des détections associées à la trajectoire T. Par contre, si le dictionnaire $D_{(T,d)}$ comporte des éléments d'autres trajectoires, l'approche proposée se rapproche des méthodes de classification de type SRC [124] à base de représentations parcimonieuses collaboratives. En effet, on peut alors considérer que l'on cherche à classifier la détection d parmi plusieurs classes, chacune d'entre elles étant associée à

une trajectoire particulière de $\mathcal{S}_{(T,d)}$, en considérant une représentation collaborative avec un dictionnaire composé d'éléments de toutes ces différentes classes.

Types de représentations parcimonieuses possibles

Cette approche peut amener à différentes variantes selon le dictionnaire $D_{(T,d)}$ que l'on considère en pratique. En effet, $D_{(T,d)}$ est un dictionnaire commun à un ensemble de trajectoires $\mathcal{S}_{(T,d)}$, $D_{(T,d)} = D_{\mathcal{S}_{(T,d)}}$, mais plusieurs choix sont possibles à ce niveau. Les différentes possibilités que nous allons considérer sont les suivantes :

(i) Une première possibilité consiste à considérer, pour tout couple trajectoire-détection (T, d),

$$\mathcal{S}_{(T,d)} = \{T\}. \tag{III.9}$$

Avec ce choix, les dictionnaires spécifiques des trajectoires sont considérés indépendamment et les valeurs d'affinité sont définies à partir d'erreurs de reconstruction non résiduelles (toute la représentation est prise en compte dans la reconstruction). Les représentations parcimonieuses ne sont ici pas collaboratives entre les trajectoires et il est nécessaire de déterminer une représentation parcimonieuse différente pour chaque couple (T, d) candidat à l'appariement. La méthode employant ce choix pour les calculs des valeurs d'affinité sera appelée **TSSR** (*Target Specific Sparse Representations*).

(ii) Une seconde possibilité est de considérer

$$\mathcal{S}_{(T,d)} = \{T',\ (T', d) \in \mathcal{L}\}. \tag{III.10}$$

Pour rappel, l'ensemble $\mathcal{L}$ est constitué des couples (T, d) tels que la trajectoire T et la détection d sont à la fois proches spatialement dans l'image et vis-à-vis de leurs tailles respectives. $D_{(T,d)}$ est alors ici composé de tous les dictionnaires spécifiques des trajectoires proches de d dont l'association peut être envisagée. Les représentations parcimonieuses sont ici collaboratives vis-à-vis des trajectoires de $\mathcal{S}_{(T,d)}$ et une seule représentation parcimonieuse devra être calculée par détection, quel que soit le nombre de trajectoires candidates à l'appariement avec cette détection (puisque le dictionnaire $D_{(T,d)}$ ne dépend que de d). Cette possibilité sera nommée **LSCR** (*Local Sparse Collaborative Representations*).

(iii) Un dernier choix est d'employer

$$\mathcal{S}_{(T,d)} = \mathcal{T} = \{T_1, ..., T_{n_{traj}}\}. \tag{III.11}$$

Cette fois, les dictionnaires $D_{(T,d)}$ sont tous identiques et consistent en un dictionnaire commun à l'ensemble des trajectoires déjà estimées. Cela signifie que les représentations parcimonieuses sont collaboratives vis-à-vis de toutes les trajectoires, sans se limiter comme précédemment aux trajectoires proches de la détection d. Tout comme le cas précédent, **LSCR**, une seule représentation parcimonieuse est considérée par détection. Ce choix de représentations sera appelé **GSCR** (*Global Sparse Collaborative Representations*).

Ces différentes possibilités sont illustrées en figure III.6 où les dictionnaires $D_{(T,d)}$ sont spécifiés avec les trajectoires à partir desquelles ils sont définis.

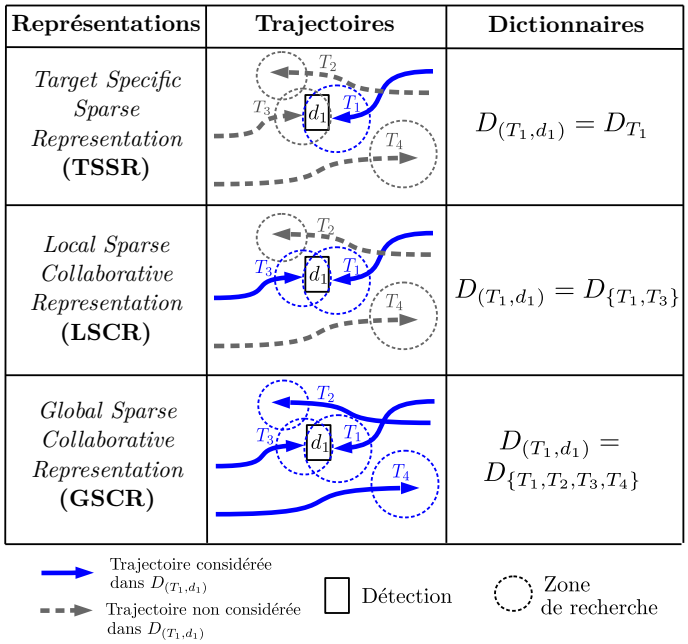


FIGURE III.6 – Types de représentations parcimonieuses considérés et dictionnaires associés.

Limitation importante de l'approche proposée

Une limitation majeure de toutes ces approches est cependant qu'elles nécessitent le calcul d'un grand nombre de représentations parcimonieuses (une par détection pour les approches collaboratives **LSCR** et **GSCR**, une par couple trajectoire-détection considéré pour l'approche non collaborative **TSSR**). De plus, les deux approches collaboratives nécessitent l'emploi de dictionnaires contenant potentiellement un très grand nombre d'éléments, et cela d'autant plus dans le cas de l'approche **GSCR**. Il est donc crucial de pouvoir calculer toutes ces représentations parcimonieuses très rapidement malgré le nombre d'éléments potentiellement important des dictionnaires envisagés. La prochaine sous-section détaille donc des méthodes d'optimisation permettant de déterminer ces représentations efficacement.

III.3.2 Optimisation par méthodes de gradient proximal

Calculer les représentations parcimonieuses envisagées précédemment, pour représenter chaque détection comme décrit par l'équation (III.7), nécessite de résoudre le problème suivant :

$$\min_{\alpha} \frac{1}{2}||y - D\alpha||_2^2 + \lambda||\alpha||_1. \tag{III.12}$$

La présence d'un terme faisant intervenir une norme l_1 rend la fonction objectif non-différentiable et complexifie sa minimisation en empêchant de recourir à des méthodes d'optimisation classiques comme la descente de gradient usuelle. Cette sous-section discute des stratégies qui peuvent être mises en oeuvre pour résoudre efficacement ce problème. Tout d'abord, nous expliquerons comment l'optimisation globale, c'est-à-dire exacte, de l'équation (III.12) est possible avec l'emploi de méthodes de descente de gradient proximal. Nous verrons ensuite comment le calcul de

représentations parcimonieuses définies sur de grands dictionnaires peut être fortement accéléré à l'aide de stratégies à base d'ensembles actifs.

Méthodes proximales de descente de gradient

L'optimisation du problème (III.12), définissant le calcul d'une représentation parcimonieuse dérivée d'une norme l_1, est complexe pour deux raisons principales. Premièrement, aucune formule explicite pour la solution n'est connue ce qui nécessite de recourir à des méthodes d'optimisation itératives. Secondement, ce problème est non-différentiable du fait de la non-différentiabilité de la norme l_1 et il n'est donc pas possible de définir le gradient de cette fonction objective. Néanmoins ce problème est convexe, ce qui permet d'envisager des méthodes de résolution exactes et certains critères d'optimalité des solutions.

Ce problème peut se mettre sous la forme plus générale suivante :

$$\min_u \ f(u) + g(u), \tag{III.13}$$

avec f et g deux fonctions convexes propres [1] fermées [2], et avec de plus f différentiable. Le problème (III.12) correspond en effet à un cas particulier du problème (III.13) avec f et g définies par :

$$f(u) = \frac{1}{2}||y - Du||_2^2 \tag{III.14}$$

$$g(u) = \lambda||u||_1. \tag{III.15}$$

Le problème (III.13), avec ces hypothèses sur les fonctions f et g, constitue la formulation générale des problèmes pouvant être résolus par les méthodes de gradient proximal [101]. Au lieu d'utiliser exclusivement le gradient de la fonction à minimiser, ces méthodes exploitent à la fois le gradient de la fonction f, supposée différentiable, et un opérateur proximal de la fonction γg, avec $\gamma > 0$, défini par :

$$prox_{\gamma g}(u) = \arg\min_v g(v) + \frac{1}{2\gamma}||u - v||_2^2. \tag{III.16}$$

Puisque g est supposée convexe propre fermée, il est possible de montrer que le problème (III.16) admet un unique minimum global. Cela justifie que l'opérateur proximal est effectivement bien défini et unique pour tout vecteur u sous ces hypothèses. Les opérateurs proximaux peuvent être vus comme une extension du gradient pour des fonctions non-différentiables. L'opérateur $prox_g$ peut en effet être relié dans tous les cas aux sous-gradients de g ainsi qu'à son gradient lorsque g est différentiable [101].

Pour résoudre le problème général (III.13), l'idée principale des méthodes de gradient proximal est de partir d'une solution initiale u_0 et de modifier itérativement cette solution, à l'aide du gradient de f et de l'opérateur proximal de g, en suivant l'équation :

$$u_{k+1} = prox_{\rho_k g}(u_k - \rho_k \nabla f(u_k)), \tag{III.17}$$

où ρ_k est un pas de descente qui peut être choisi suivant différentes stratégies. Cette approche est alors appelée méthode de gradient proximal, mais est plus fréquemment dénommée ISTA (*Iterative Soft Threshold Algorithm*), et est décrite par

1. Une fonction $f : \mathbb{R}^n \to \bar{\mathbb{R}}$ est propre si $\forall x \in \mathbb{R}^n,\ f(x) > -\infty$ et $\exists x \in \mathbb{R}^n : f(x) < \infty$.
2. Une fonction $f : \mathbb{R}^n \to \bar{\mathbb{R}}$ est fermée si $\forall \alpha \in \mathbb{R}$, $\{x \in \mathbb{R}^n,\ f(x) \leq \alpha\}$ est un ensemble fermé.

données : f, g, u_0
$k = 1$, $u_k = u_0$;
répéter
  trouver un pas ρ_k optimal par recherche linéaire;
  $u_{k+1} = prox_{\rho_k g}(u_k - \rho_k \nabla f(u_k))$;
  $k = k + 1$;
jusqu'à *critère d'arrêt*;
retourner u_k;

Algorithme III.1 : Méthode de gradient proximal (ISTA).

données : f, g, u_0
$k = 1$, $u_k = u_0$;
répéter
  $\mu_k = \frac{k}{k+3}$;
  $v_k = u_k + \mu_k(u_k - u_{k-1})$;
  trouver un pas ρ_k optimal par recherche linéaire;
  $u_{k+1} = prox_{\rho_k g}(v_k - \rho_k \nabla f(v_k))$;
  $k = k + 1$;
jusqu'à *critère d'arrêt*;
retourner u_k;

Algorithme III.2 : Méthode de gradient proximal accélérée (APG ou FISTA).

l'algorithme III.1. Cette dernière dénomination fait référence au problème spécifique du Lasso, car appliquer l'opérateur proximal d'une norme l_1 revient à réaliser un seuillage souple (*soft thresholding*). Ce nom est aussi employé de manière quelque peu abusive pour la méthode d'optimisation générale indépendamment du problème spécifique résolu. En supposant que le problème (III.13) admet au moins une solution, que ∇f est une fonction lipschitzienne et en choisissant ρ_k par une recherche linéaire appropriée, comme celle présentée dans [11], alors l'algorithme ISTA converge effectivement vers un minimum global du problème (III.13).

Bien que la convergence de la méthode de gradient proximal (ou ISTA) vers un minimum global soit garantie sous respect des hypothèses évoquées précédemment, cette convergence s'effectue à une vitesse linéaire en $O(\frac{1}{k})$ où k dénote le nombre d'itérations effectuées. Les méthodes de descente de gradient usuelles, avec une vitesse de convergence elle-aussi linéaire sous des hypothèses assez générales, ont été grandement accélérées suite aux travaux de Nesterov [96] qui garantissent une vitesse de convergence quadratique. Ces travaux ont été étendus aux méthodes de gradient proximal [11], menant à une méthode garantissant, sous les mêmes hypothèses, une vitesse de convergence quadratique en $O(\frac{1}{k^2})$. Cette méthode, décrite par l'algorithme III.2, est appelée méthode de gradient proximal accélérée (APG) ou, quelque peu abusivement, FISTA (*Fast Iterative Soft Threshold Algorithm*). Passer d'une vitesse de convergence linéaire en $O(\frac{1}{k})$ à une vitesse de convergence quadratique en $O(\frac{1}{k^2})$ est crucial en pratique, une approche en $O(\frac{1}{k^2})$ pouvant aboutir en quelques centaines d'itérations seulement à une précision suffisante comme illustré en figure III.7.

Concernant l'arrêt de l'algorithme, que ce soit pour la méthode de gradient proximal (ISTA) ou la variante accélérée (FISTA), un critère d'arrêt peut être défini à partir de considérations sur des écarts de dualité (*duality gap*) qui permettent de borner l'erreur par rapport à la solution optimale [5]. Il est possible, de manière plus naïve, de considérer un nombre limite d'itérations voire combiner un nombre limite d'itérations avec un critère sur des écarts de dualité.

Considérations spécifiques pour les représentations parcimonieuses

Les représentations parcimonieuses que nous souhaitons utiliser pour notre méthode de suivi, définies par le problème (III.12), peuvent être efficacement calculées par une méthode de gradient proximal accélérée (FISTA). Cependant, plusieurs considérations spécifiques à ce problème particulier permettent d'aboutir à des méthodes d'optimisation encore plus rapides sous certaines hypothèses.

Nous revenons donc maintenant au cas du problème spécifique (III.12) dont l'expression est, pour rappel :

$$\min_{\alpha} \frac{1}{2}||y - D\alpha||_2^2 + \lambda||\alpha||_1.$$

Nous précisons que nous considérons ici un dictionnaire $D \in \mathbb{R}^{n \times m}$ de n éléments $e_1, .., e_n$ de dimension m. Pour rappel, nous pouvons décomposer notre fonction objective comme la somme de deux fonctions f et g avec

$$f(\alpha) = \frac{1}{2}||y - D\alpha||_2^2$$
$$g(\alpha) = \lambda||\alpha||_1.$$

Dans toute la suite de ce chapitre, f et g feront désormais référence à ces fonctions spécifiques. Ces deux fonctions vérifient bien les hypothèses requises pour appliquer l'algorithme FISTA, à savoir f et g sont bien des fonctions propres convexes fermées et f est différentiable avec de plus ∇f fonction lipschitzienne. Nous pouvons maintenant détailler les différents éléments utilisés pour employer la méthode de gradient proximal accélérée, à savoir :

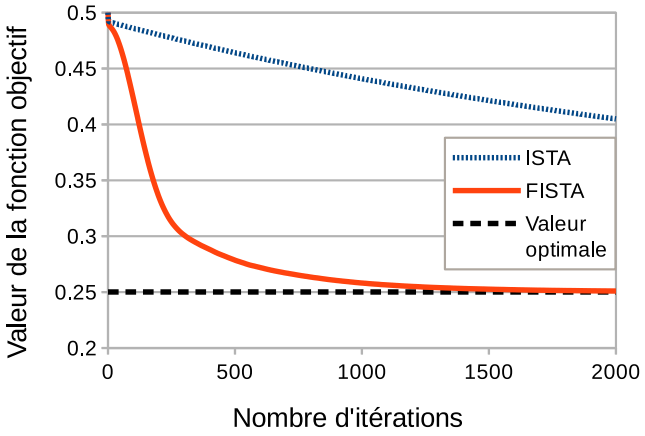


FIGURE III.7 – Comparaison des vitesses de convergence des algorithmes ISTA et FISTA en fonction du nombre d'itérations, pour un exemple d'optimisation de l'équation (III.12). Le dictionnaire D considéré ici comporte $n = 10000$ éléments de dimension $m = 1000$, normalisés en norme l_2, et la constante λ est fixée à 0.01. L'exemple y reconstruit est choisi aléatoirement à partir d'une loi normale, puis normalisé en norme l_2. Les allures des deux courbes illustrent clairement les vitesses de convergence de ces deux méthodes, linéaire pour la méthode ISTA et quadratique pour la méthode FISTA. La méthode FISTA permet ainsi une convergence avec un nombre beaucoup plus restreint d'itérations.

(i) l'opérateur proximal de g, $prox_{\gamma g}(\alpha) = \arg\min_\beta g(\beta) + \frac{1}{2\gamma}||\alpha - \beta||_2^2$,

(ii) le gradient de f, $\nabla f(\alpha) = D^\mathsf{T}(D\alpha - y)$,

(iii) l'évaluation de la fonction f, $f(\alpha) = \frac{1}{2}||y - D\alpha||_2^2$, requise pour la recherche linéaire du paramètre ρ_k.

Ces trois fonctions sont les principaux éléments calculés à chaque itération de l'algorithme et la complexité générale dépendra fortement de la façon dont ces éléments sont évalués en pratique.

Concernant l'opérateur proximal associé à la norme l_1, son évaluation, comme expliqué dans [101], correspond à :

$$prox_{\gamma l_1}(\alpha)_i = \begin{cases} \alpha_i - \gamma & si\ \alpha_i > \gamma \\ 0 & si\ |\alpha_i| \leq \gamma \\ \alpha_i + \gamma & si\ \alpha_i < -\gamma \end{cases}, \tag{III.18}$$

ce qui revient à effectuer un seuillage souple sur chacune des dimensions de α. Calculer cet opérateur est ainsi une opération très peu coûteuse effectuée en $O(n)$. L'expression très simple de cet opérateur proximal justifie en partie l'intérêt des approches proximales pour optimiser les problèmes pénalisés par une norme l_1.

Concernant le gradient ∇f et la fonction f, leurs évaluations directes se ramènent à des multiplications de vecteurs par les matrices D et D^T, et sont donc réalisées en $O(nm)$. Chaque itération de l'algorithme FISTA faisant intervenir un nombre équivalent d'évaluations d'opérateurs proximaux, de gradient ∇f et d'évaluations de f (en prenant en compte l'étape de recherche linéaire), le temps de calcul de chaque itération s'effectue alors en $O(nm)$.

Ce temps de calcul peut être fortement réduit dans certains cas, comme expliqué dans [101], à l'aide du pré-calcul initial de la matrice de Gram de la matrice du dictionnaire D, $D^\mathsf{T}D$. En effet, les évaluations de f et de son gradient sont alors effectuées en $O(n^2)$ en suivant les formules :

$$f(\alpha) = \frac{1}{2}\alpha^\mathsf{T}(D^\mathsf{T}D)\alpha + \frac{1}{2}y^\mathsf{T}y - \alpha^\mathsf{T}(D^\mathsf{T}y) \tag{III.19}$$

$$\nabla f(\alpha) = (D^\mathsf{T}D)\alpha - D^\mathsf{T}y, \tag{III.20}$$

les éléments $y^\mathsf{T}y$ et $D^\mathsf{T}y$ étant eux-aussi supposés initialement pré-calculés au début de l'optimisation. Chaque itération de l'algorithme FISTA s'effectue maintenant en $O(n^2)$, avec un coût de pré-calcul en $O(mn^2)$. Ce pré-calcul de la matrice de Gram est donc particulièrement intéressant lorsque $n \ll m$, c'est-à-dire lorsque le dictionnaire comporte peu d'éléments qui sont de grande dimension, et mène à des gains significatifs en temps de calcul comme illustré en figure III.8.

Optimisation avec ensembles actifs

Bien que le pré-calcul de la matrice de Gram $D^\mathsf{T}D$ permette d'accélérer significativement le calcul des représentations parcimonieuses dans le cas de dictionnaires comportant peu d'éléments, cette stratégie ne peut être employée pour de grands dictionnaires où la condition $n \ll m$ n'est pas vérifiée. Or, dans le cas des représentations collaboratives globales (**GSCR**) introduites à la section précédente, nous employons un dictionnaire contenant un grand nombre d'éléments, éventuellement plusieurs milliers si le nombre de cibles à suivre est important. L'algorithme FISTA

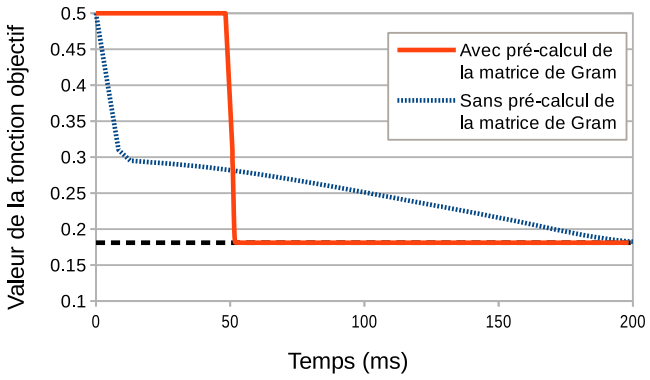


FIGURE III.8 – Comparaison de la vitesse de convergence de l'algorithme FISTA avec et sans pré-calcul de la matrice de Gram, sur un exemple d'optimisation de l'équation (III.12). Le dictionnaire D considéré ici comporte $n = 100$ éléments de dimension $m = 10000$, normalisés en norme l_2, et la constante λ est fixée à 0.01. L'exemple y reconstruit est une combinaison linéaire d'une dizaine d'éléments du dictionnaire, choisie aléatoirement, à laquelle est ajoutée un bruit gaussien. l'élément y est enfin normalisé en norme l_2. Avec pré-calcul de la matrice de Gram, on observe un plateau au début de la courbe qui correspond au temps de pré-calcul de cette matrice. Malgré ce temps de pré-calcul, l'approche avec pré-calcul de la matrice de Gram est ici beaucoup plus rapide car chaque itération de l'optimisation est ensuite bien moins coûteuse en temps de calcul (puisque $n \ll m$).

usuel, avec ou sans pré-calcul de la matrice de Gram, est alors bien trop lent pour permettre d'envisager l'emploi de telles représentations collaboratives dans une application de suivi multi-objets avec un fonctionnement proche du temps réel. Cette situation oblige à recourir à d'autres stratégies pour accélérer ces méthodes, en envisageant des méta-algorithmes à base d'ensembles actifs [5].

Les méta-algorithmes à base d'ensembles actifs reposent sur une idée assez basique. Les solutions α du problème (III.12) ont un support limité à un faible nombre de dimensions du fait de la pénalisation en norme l_1 qui induit des solutions parcimonieuses, alors que les méthodes proximales (ISTA ou FISTA) considèrent toutes ces dimensions de manière similaire. Les méta-algorithmes à base d'ensemble actifs vont au contraire se focaliser sur un sous-ensemble de dimensions, dites actives, qui sera progressivement augmenté en ajoutant des dimensions non-actives.

Étant donné un sous-ensemble d'indices $\mathcal{A}$, il est possible de définir le sous-problème associé au problème (III.12) restreint à $\mathcal{A}$:

$$\min_{\alpha} \frac{1}{2}||y - D_{\mathcal{A}}\alpha||_2^2 + \lambda||\alpha||_1. \tag{III.21}$$

Le dictionnaire $D_{\mathcal{A}}$ est obtenu en se restreignant aux éléments $\{e_i, i \in \mathcal{A}\}$, c'est-à-dire en ne considérant que les colonnes de D dont l'indice est inclus dans $\mathcal{A}$. Si $\alpha_{\mathcal{A}}$ est une solution du sous-problème (III.21), une question qui se pose naturellement est alors de savoir si le vecteur $\widetilde{\alpha_{\mathcal{A}}}$, déduit de $\alpha_{\mathcal{A}}$ en considérant des coefficients nuls pour les dimensions non-actives (non considérées dans $\mathcal{A}$), est aussi solution du problème général (III.12). Cela correspond au cas où $\mathcal{A}$ est le support d'une solution optimale du problème général, mais est-il possible de le vérifier en pratique ?

Dans le cas du problème (III.12), il existe des conditions nécessaires et suffisantes

données : D, y
$\mathcal{A} = \varnothing$, $\alpha_{\mathcal{A}} = 0$;
répéter
  $\mathcal{S} = \{$au plus n_{sel} indices $i \notin \mathcal{A}$ maximisant $|e_i^\mathsf{T}(D\widetilde{\alpha_{\mathcal{A}}} - y)|$
  avec $|e_i^\mathsf{T}(D\widetilde{\alpha_{\mathcal{A}}} - y)| > \lambda\}$;
  Utilisant $\alpha_{\mathcal{A}}$ comme position initiale, trouver la solution optimale $\alpha_{\mathcal{A}\cup\mathcal{S}}$ du problème $\min_\alpha \frac{1}{2}||y - D_{\mathcal{A}\cup\mathcal{S}}\alpha||_2^2 + \lambda||\alpha||_1$;
  $\mathcal{A} = \mathcal{A} \cup \mathcal{S}$;
jusqu'à $||D^\mathsf{T}(D\widetilde{\alpha_{\mathcal{A}}} - y)||_\infty \leq \lambda$;
retourner $\widetilde{\alpha_{\mathcal{A}}}$;

Algorithme III.3 : Méta-algorithme avec ensembles actifs pour le calcul de représentations parcimonieuses.

d'optimalité [5]. Un vecteur α est en effet une solution globale du problème (III.12) si et seulement si :

(i) $||\nabla f(\alpha)||_\infty \leq \lambda$,

(ii) $-\frac{1}{\lambda}(\nabla f(\alpha))^\mathsf{T}\alpha = ||\alpha||_1$.

Cependant, dans le cas spécifique où le vecteur $\alpha_{\mathcal{A}}$ est une solution optimale du sous-problème (III.21), $\widetilde{\alpha_{\mathcal{A}}}$ est une solution optimale du problème général (III.12) si et seulement si la seule condition (i) est satisfaite [5].

En démarrant l'optimisation avec un ensemble actif $\mathcal{A}$ initialement vide, l'idée générale va être à chaque itération du méta-algorithme, étant donné la solution $\alpha_{\mathcal{A}}$ du sous-problème (III.21), d'ajouter à $\mathcal{A}$ certains indices des dimensions qui ne satisfont pas la condition (i) pour $\widetilde{\alpha_{\mathcal{A}}}$ et de résoudre de nouveau le sous-problème (III.21) sur ce nouvel ensemble $\mathcal{A}$ de variables actives. Le méta-algorithme s'arrête lorsque la condition (i) est finalement satisfaite pour $\widetilde{\alpha_{\mathcal{A}}}$, et le vecteur $\widetilde{\alpha_{\mathcal{A}}}$ correspond alors à une solution optimale du problème général (III.12). Une possibilité pour sélectionner les indices i ajoutés à $\mathcal{A}$ est de choisir ceux qui maximisent $|(\nabla f(\widetilde{\alpha_{\mathcal{A}}}))_i|$ (c'est-à-dire $|e_i^\mathsf{T}(D\widetilde{\alpha_{\mathcal{A}}} - y)|$) avec $|(\nabla f(\widetilde{\alpha_{\mathcal{A}}}))_i| > \lambda$ (c'est-à-dire $|e_i^\mathsf{T}(D\widetilde{\alpha_{\mathcal{A}}} - y)| > \lambda$), et en considérant au maximum n_{sel} nouveaux indices à chaque itération. Ce principe est décrit par l'algorithme III.3.

En pratique, l'algorithme III.3 est adapté de manière à ne pas résoudre exactement le sous-problème (III.21) à chaque itération. Il est possible de se limiter à un certain nombre d'itérations de la méthode de gradient proximal accélérée pour le sous-problème (III.21), donnant une solution $\alpha_{\mathcal{A}}$ approchée. On continue le méta-algorithme même si la condition d'optimalité (i) est vérifiée pour $\widetilde{\alpha_{\mathcal{A}}}$, l'ensemble actif n'étant alors pas agrandi dans ce cas ($\mathcal{S} = \varnothing$). Le méta-algorithme est finalement arrêté lorsqu'un critère d'arrêt est vérifié, comme pour les méthodes de gradient proximal. Ce critère d'arrêt peut typiquement faire intervenir un écart de dualité ou un nombre limite d'itérations.

L'avantage considérable d'une telle approche est que le sous-problème (III.21) fait intervenir un dictionnaire $D_{\mathcal{A}}$ composé d'un faible nombre d'éléments. Il est donc possible d'employer, pour résoudre ce sous-problème à chaque itération principale du méta-algorithme, la méthode de gradient proximal accélérée ainsi que le pré-calcul de la matrice de Gram discuté précédemment. Une optimisation à base d'ensemble actifs est donc particulièrement adaptée lorsque les dictionnaires impliqués font intervenir un grand nombre d'éléments comme illustré en figure III.9.

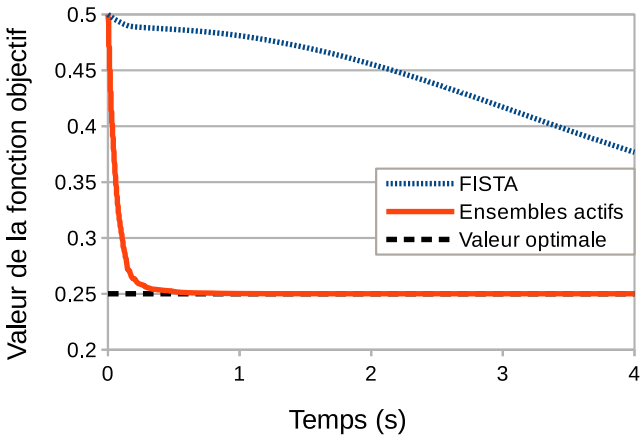


FIGURE III.9 – Comparaison des vitesses de convergence de la méthode FISTA classique comparée à une approche FISTA avec ensembles actifs, pour un exemple d'optimisation de l'équation (III.12). Le dictionnaire D considéré comporte $n = 10000$ éléments de dimension $m = 1000$, normalisés en norme l_2, et la constante λ est fixée à 0.01. L'exemple y reconstruit est choisi aléatoirement à partir d'une loi normale, puis normalisé en norme l_2. L'emploi d'un méta-algorithme à base d'ensembles actifs permet effectivement de converger bien plus rapidement vers une solution optimale.

III.3.3 Évaluations et analyse des résultats

Nous décrivons ici certains détails de l'implémentation de l'approche proposée, avec les trois variantes **TSSR**, **LSCR** et **GSCR**. Ces trois approches sont ensuite comparées entre elles, en étant évaluées sur des séquences usuelles pour le suivi de personnes. Les résultats obtenus sont aussi comparés à certaines approches récentes de suivi.

Implémentation

L'approche considérée utilise le système général de suivi en ligne décrit à la section III.2 et les affinités décrites en sous-section III.3.1. Cette méthode est implémentée en C++ et est testée sur une machine en n'exploitant qu'un seul CPU (un seul coeur) à 2.7 GHz. Ce code repose sur la bibliothèque OpenCV [3] pour la gestion des images, l'extraction des caractéristiques et le filtre de Kalman, ainsi que la bibliothèque Eigen [4] de calcul matriciel qui est principalement utilisée pour réaliser notre propre implémentation des méthodes d'optimisation de gradient proximal.

Nous n'employons pas ici de caractéristiques visuelles élaborées pour les caractéristiques (y_d) et utilisons directement les valeurs d'intensités RGB normalisées des boîtes des détections (redimensionnées à 30×30 pixels). Ce choix basique est assez fréquent dans les méthodes de suivi mono-objet à base de représentations parcimonieuses [135]. L'optimisation des représentations parcimonieuses est effectuée avec une méthode de gradient proximal accélérée (FISTA) à base d'ensembles actifs et avec le pré-calcul de la matrice de Gram. Ce choix se révèle crucial pour le temps de calcul, notamment en ce qui concerne l'approche avec les représentations parcimo-

3. URL : *opencv.org*
4. URL : *eigen.tuxfamily.org*

nieuses collaboratives globales (**GSCR**). En pratique, l'algorithme III.3 est utilisé en limitant le nombre d'étapes d'agrandissement, ou d'étapes de sélection, de l'ensemble actif $\mathcal{A}$ à dix étapes en sélectionnant au plus dix nouveaux indices ($n_{sel} = 10$). L'optimisation des représentations parcimonieuses sur l'ensemble actif $\mathcal{A}$ entre deux de ces étapes est effectuée en limitant l'algorithme FISTA à dix itérations. Enfin, nous employons dans notre approche un algorithme glouton pour résoudre l'association de données image après image, comme explicité en sous-section III.2.2.

Protocole expérimental

La méthode que nous proposons dépend de nombreux paramètres, comme par exemple les critères employés pour gérer les transitions entre les différents états des cibles définis en sous-section III.2.3, le nombre maximal n_{dict} de détections considérées dans les dictionnaires spécifiques à chaque trajectoire... Ces différents paramètres ont été optimisés manuellement, de manière empirique, en cherchant la configuration de paramètres optimale pour l'ensemble des vidéos, le même jeu de paramètres étant employé pour toutes les vidéos afin d'éviter d'être trop spécifique à certaines scènes. En particulier, le nombre maximal de détections dans les dictionnaires spécifiques à chaque trajectoire, n_{dict}, est fixé à 30 et le paramètre λ de l'équation (III.7) est fixé à 0.1.

Si la majorité des paramètres semblent indépendants du type de représentations employé (**TSSR**, **LSCR** et **GSCR**), le paramètre θ optimal est particulièrement spécifique à chacune de ces variantes. En effet, ce paramètre, comme décrit en sous-section III.2.2, est un seuil limite pour l'appariement des détections et des trajectoires puisqu'un appariement entre une détection d et une trajectoire T n'est possible que si $a(T, d) \geq \theta$. Selon le type de représentations parcimonieuses employé, les valeurs $a(T, d)$ prennent des valeurs très différentes et fixer un unique seuil θ est inapproprié. Chacune des variantes proposées (**TSSR**, **LSCR** et **GSCR**) a été évaluée sur l'ensemble des vidéos avec plusieurs valeurs de θ, également distribuées sur l'intervalle $[-0.5, 0]$, et celles qui donnaient les meilleurs résultats en moyenne en MOTA ont été sélectionnées. Ce paramètre θ est ainsi fixé à -0.45 pour les représentations de type **GSCR**, -0.35 pour celles de type **LSCR** et enfin -0.2 pour celles de type **TSSR**.

Nous avons utilisé des séquences vidéo usuelles en suivi multi-personnes qui étaient employées à l'époque de ces travaux afin de comparer au mieux notre approche aux autres méthodes existantes. Cependant, la comparaison des différentes approches de suivi était alors difficilement réalisable. En effet, de nombreuses méthodes se comparaient à d'autres approches en n'employant pas le même type de détecteur de personnes en entrée de l'algorithme de suivi (par exemple un DPM [39] comparé à un HOG+SVM de type Dalal-Triggs [28]), ce qui faussait la pertinence de l'évaluation. De même, des implémentations différentes des métriques CLEARMOT étaient utilisées avec parfois des modifications significatives (comme le critère de recouvrement IOU) qui rendaient les résultats non comparables. Nous nous sommes limités à l'utilisation des détections publiques les plus employées, fournies par les articles [13,88,112,134]. Notre méthode est comparée à des approches récentes (principalement [13,94,112,134]) qui utilisent ces détections, et nous nous sommes de plus limités aux approches qui fournissaient les trajectoires de leurs résultats afin d'évaluer les métriques CLEARMOT en utilisant la même implémentation provenant du code de l'article [134]. Néanmoins, contrairement au protocole suivi dans [134], nous

estimons les métriques CLEARMOT avec un seuil standard de 0.5 pour le critère de recouvrement IOU.

Notre approche est ainsi évaluée sur plusieurs types de vidéos : PETS S2L1, PETS S2L2, TownCenter et ParkingLot. Ces vidéos se différencient principalement en terme de densité de personnes à suivre et en terme d'orientation et de fréquence de la caméra. Concernant les détections, deux jeux de détections publiques différents sont employés pour les vidéos PETS S2L1, PETS S2L2 et TownCenter, qui sont ceux utilisés dans [134]. Pour ParkingLot, nous employons les mêmes détections que [112].

Comparaison entre les différents types de représentations

Les résultats entre les différentes variantes de notre approche sont indiqués au tableau III.1. Seules les valeurs de MOTA et des changements d'identité (IDS), qui sont deux des principales métriques CLEARMOT, sont indiquées par soucis de clarté. On remarque que les représentations collaboratives (**LSCR** ou **GSCR**) présentent des résultats légèrement meilleurs sur l'ensemble des séquences comparées aux représentations non collaboratives (**TSSR**). Concernant les représentations collaboratives, on observe que les représentations globales (**GSCR**) donnent dans la majorité des cas des résultats meilleurs comparées aux représentations locales (**LSCR**).

Pour expliquer ce gain en performances observé avec l'emploi de représentations collaboratives globales, nous avons étudié la répartition des valeurs d'affinité $a(T, d)$ pour les couples trajectoire-détection vérifiant $(T, d) \in \mathcal{L}$, en séparant les couples corrects des couples incorrects à partir de la vérité terrain. Les distributions obtenues sont alors indiquées en figure III.10. Le paramètre θ dont nous avons discuté précédemment devrait être choisi de manière à séparer ces deux distributions. On remarque que les distributions des couples corrects et incorrects sont moins facilement séparables dans le cas des représentations non collaboratives (**TSSR**) tandis qu'au contraire ces distributions sont facilement séparables dans le cas des représentations collaboratives globales (**GSCR**). De plus, dans ce dernier cas, une large plage de valeurs pour θ permet de séparer assez correctement ces distributions, ce qui se traduit par une certaine robustesse vis-à-vis du choix de ce paramètre.

En pratique, il semble complexe de décider si un couple (T, d) correspond effectivement à un bon appariement en considérant uniquement l'erreur de reconstruction de la représentation de y_d vis-à-vis du dictionnaire spécifique à T (comme effectué avec la variante **TSSR**). En effet, toutes les cibles étant des objets d'une même catégorie, ici des personnes, elles partagent déjà une apparence commune et peuvent être représentées avec une erreur assez faible par des vues d'autres cibles. Au contraire, les représentations parcimonieuses collaboratives forcent les détections à être représentées uniquement par les cibles les plus ressemblantes, en mettant celles-ci en compétition. Ces représentations ont ainsi un pouvoir discriminatif plus fort entre les cibles car une détection sera dans tous les cas représentée par peu de cibles, du fait du caractère parcimonieux des représentations, même si toutes les cibles sont proches en apparence de cette détection.

Il est possible d'expliquer le gain en performances des représentations collaboratives globales (**GSCR**) par rapport aux représentations collaboratives locales (**LSCR**) en observant qu'il est peu probable qu'une fausse détection d soit représentée, dans le cas des représentations globales, par une trajectoire T proche (c'est-à-dire avec $(T, d) \in \mathcal{L}$) et la détection d ne pourra alors être associée à aucune trajectoire. Dans le cas des représentations collaboratives locales (**LSCR**), la détec-

Métrique	Rep.	S2L1		S2L2		Town Center		Parking Lot
		[88]	[134]	[88]	[134]	[134]	[13]	[112]
MOTA	**TSSR**	0.688	0.711	0.383	0.421	0.607	0.651	**0.857**
	LSCR	**0.702**	0.712	0.405	0.427	0.606	0.655	**0.857**
	GSCR	0.695	**0.713**	**0.413**	**0.439**	**0.613**	**0.661**	0.856
IDS	**TSSR**	37	**18**	215	214	211	225	18
	LSCR	29	22	**214**	210	214	210	18
	GSCR	**25**	19	225	**194**	**192**	**201**	**17**

Tableau III.1 – Résultats en termes de MOTA et IDS pour les représentations proposées **TSSR**, **LSCR** et **GSCR** (meilleures valeurs en gras et rouge). Seconde ligne : détections employées.

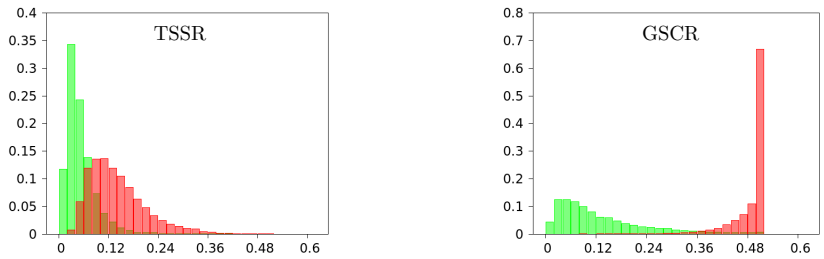
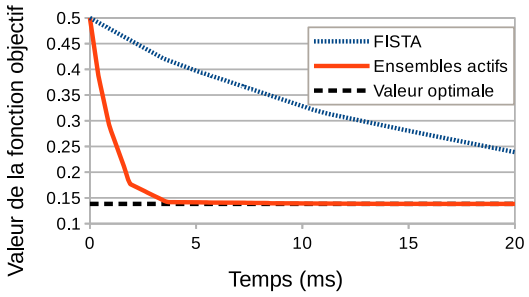


FIGURE III.10 – Distribution des coûts d'association $-a(T, d)$ pour les couples trajectoire-détection corrects (en vert) et incorrects (en rouge) sur la vidéo PETS S2L2. La distribution de ces coûts pour les représentations de type **TSSR** est indiquée à gauche, et celle pour les représentations de type **GSCR** est indiquée à droite.

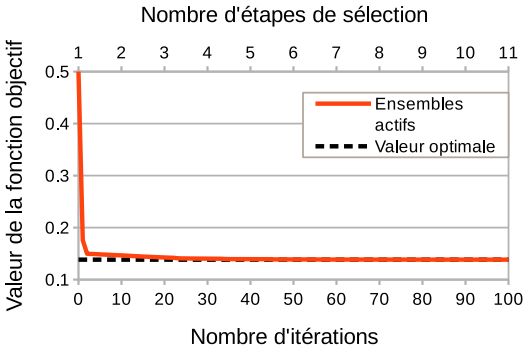
tion d sera quand même représentée par certaines des trajectoires proches auxquelles elle pourra être éventuellement associée. Les représentations collaboratives globales semblent ainsi plus naturellement "filtrer" les mauvaises détections par rapport aux représentations collaboratives locales.

Temps d'exécution

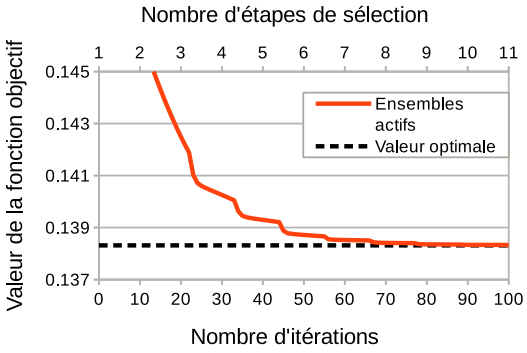
Les représentations parcimonieuses collaboratives globales (**GSCR**) donnent de meilleures performances comparées aux autres types de représentations mais nécessitent d'optimiser des représentations parcimonieuses sur des dictionnaires comportant un grand nombre d'éléments (plusieurs centaines voire milliers suivant le nombre de cibles à suivre). En figure III.11, nous montrons tout d'abord que la stratégie d'optimisation à base d'ensembles actifs proposée en sous-section III.3.2 permet effectivement d'accélérer significativement le calcul des représentations parcimonieuses pour des détections de la vidéo PETS S2L2 (vidéo comportant le plus de personnes et pour laquelle l'optimisation des représentations parcimonieuses est la plus coûteuse en temps de calcul). La figure III.11 indique aussi que les choix effectués en pratique, c'est-à-dire limiter l'optimisation à 10 étapes de sélection pour l'ensemble actif et limiter l'optimisation des sous-problèmes associés à 10 itérations avec une approche FISTA, sont suffisants pour atteindre une précision d'optimisa-



(a)



(b)



(c)

FIGURE III.11 – (a) : Optimisation des représentations avec et sans ensembles actifs pour les représentations de type **GSCR**. Les valeurs moyennes de la fonction objective sont indiquées ici en fonction du temps d'optimisation pour dix détections choisies aléatoirement sur la vidéo PETS S2L2. L'emploi d'une stratégie d'optimisation avec ensembles actifs permet effectivement d'accélérer significativement ces optimisations. (b) et (c) : Valeurs moyennes de la fonction objective en fonction du nombre d'itérations et d'étapes de sélection de l'optimisation avec ensembles actifs. Le graphique (c) est identique au (b) à l'exception de l'échelle des ordonnées, modifiée pour juger plus efficacement la vitesse de convergence.

tion correcte.

Nous indiquons aussi le temps d'exécution de notre approche **GSCR** au tableau III.2. Selon le nombre de cibles à suivre, notre méthode fonctionne entre 5 et 29 images par seconde (en excluant le temps de détection). Néanmoins, ces valeurs correspondent à une implémentation non parallèle exécutée sur un seul coeur d'un CPU. Comme le calcul des représentations parcimonieuses est ici l'étape limitante et que les représentations des détections sur une même image peuvent être calculées indépendamment les unes des autres, il est possible d'améliorer grandement ces chiffres avec une implémentation parallèle.

Comparaison à d'autres approches de l'état de l'art

La méthode proposée (**GSCR**) est comparée avec certaines approches récentes [13, 94, 112, 134] qui utilisent les mêmes détections publiques, les résultats associés étant présentés dans le tableau III.3. Notre approche donne le plus souvent de meilleurs résultats en termes de MOTA et se trouve dans tous les cas comparable aux autres méthodes. De plus, notre méthode produit dans l'ensemble un nombre

Vidéo	**S2L1**		**S2L2**		**Town Center**		**Parking Lot**
Détections	**[88]**	**[134]**	**[88]**	**[134]**	**[134]**	**[13]**	**[112]**
Temps de calcul (images par sec.)	24	29	5.1	8.7	8.4	6.5	10

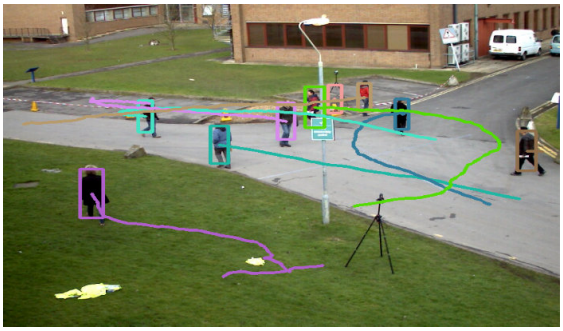
Tableau III.2 – Temps de calcul de l'approche **GSCR** en fonction des différentes vidéos et des détections employées (en exploitant un seul coeur CPU sans parallélisation). La vitesse de traitement dépend principalement de la densité de personnes présentes dans les vidéos, densité plus importante en particulier pour les vidéos PETS S2L2 et TownCenter.

bien plus faible de changements d'identité (IDS) ce qui traduit un meilleur maintien des trajectoires tout au long du suivi. Notre approche a tendance à produire légèrement plus de détections manquantes (FN) et donne par contre légèrement moins de faux positifs (FP). Considérer des détecteurs différents sur les vidéos de PETS et TownCenter ne modifie pas significativement nos résultats, ce qui indique que notre méthode de suivi est robuste, dans une certaine mesure, au détecteur d'objets employé en entrée de l'approche. La figure III.12 illustre les résultats de notre approche.

En résumé, on peut dire que notre approche **GSCR** se compare favorablement à d'autres approches de suivi récentes et produit surtout un nombre significativement plus faible de changements d'identité (IDS). La méthode **GSCR** montre ainsi que des résultats performants peuvent être obtenus en exploitant des représentations parcimonieuses collaboratives pour le suivi multi-objets, ce qui nous motive à étudier davantage les possibilités qu'offre ce type de représentations.

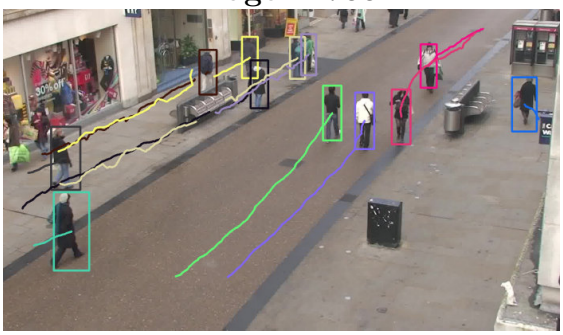
Vidéo	Det.	Méthodes	MOTA	IDS	MOTP	FP	FN
S2L1	[88]	[134]	**0.699**	35	**0.712**	805	**557**
		GSCR	0.695	**25**	0.656	**757**	631
	[134]	[134]	0.700	21	0.717	543	**827**
		GSCR	**0.713**	**19**	**0.732**	**457**	852
S2L2	[88]	[134]	**0.431**	347	**0.695**	**1318**	**4189**
		GSCR	0.413	**225**	0.660	1502	4291
	[134]	[134]	0.393	287	0.690	1416	4536
		GSCR	**0.439**	**194**	**0.711**	**1044**	**4514**
Town Center	[134]	[134]	0.607	212	0.712	7295	**20549**
		GSCR	**0.613**	**192**	**0.716**	**3983**	23476
	[13]	[134]	0.634	446	0.745	9359	16302
		[13]	0.613	318	**0.805**	12309	**14982**
		GSCR	**0.661**	**201**	0.748	**6682**	17309
Parking Lot	[112]	[94]*	0.845	**4**	0.732	-	-
		[112]*	0.793	-	**0.741**	-	-
		GSCR	**0.856**	17	0.713	266	773

Tableau III.3 – Métriques CLEARMOT sur différentes vidéos (meilleures valeurs en gras et rouge pour le MOTA et les IDS). Seconde colonne : détections employées. Le signe * indique que les scores ont été directement repris à partir des articles associés.

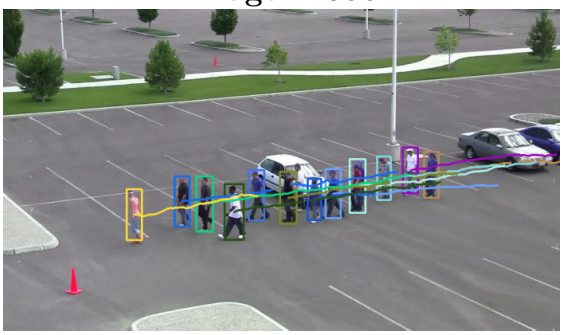


PETS S2L1, détections de [88]
Image n°700

PETS S2L2, détections de [88]
Image n°350



TownCenter, détections de [13]
Image n°2000



ParkingLot, détections de [112]
Image n°500

FIGURE III.12 – Exemples de certaines trajectoires trouvées par notre approche **GSCR**.

III.4 Extension au cas de descriptions locales

Cette section propose une extension de notre approche qui exploite des descripteurs locaux des cibles. Plusieurs méthodes de suivi, que ce soit dans un contexte mono-objet ou multi-objets, cherchent à gagner en performances en décrivant localement les cibles, c'est-à-dire en considérant un ensemble de descripteurs locaux pour décrire chaque cible. Nous envisageons donc ici de proposer des valeurs d'affinité, entre trajectoires et détections, définies à partir de considérations sur des représentations parcimonieuses qui exploitent de telles descriptions des cibles.

III.4.1 Motivations

Bien que la plupart des méthodes de suivi multi-objets s'appuient sur des descriptions holistiques des cibles, quelques approches récentes emploient des descriptions locales. La méthode proposée dans [112] propose en particulier d'employer les parties de personnes, données par un détecteur d'objets de type DPM [39], afin d'estimer de manière plus robuste les trajectoires des différents individus avec un modèle d'occultation. Une autre approche [126] utilise une description locale des cibles, sous forme de patchs pris sur une grille régulière, et analyse la cohérence du déplacement de chaque patch entre deux images successives pour déterminer une valeur d'affinité entre deux détections. Ces approches justifient en partie leurs gains en performances par l'emploi de telles descriptions locales des cibles et cela mène donc à s'interroger sur un possible emploi de descriptions locales au sein de l'approche proposée précédemment à la section III.3.

Certaines méthodes de suivi mono-objet [55, 76] ont aussi exploité des descriptions locales de la cible. Une approche basée sur des représentations parcimonieuses [55] a en particulier considéré une description de la cible sous la forme de patchs locaux sélectionnés sur une grille régulière. L'idée principale de cette méthode est de considérer comme nouvelle position de la cible la position candidate dont les patchs locaux sont les mieux reconstruits par leur représentation parcimonieuse. Pour ce faire, le dictionnaire considéré n'inclut plus seulement différentes vues récentes de l'objet suivi mais plusieurs caractéristiques des patchs locaux de vues récentes de la cible. De plus, certaines contraintes spatiales sont considérées de façon à ne reconstruire les différents patchs qu'avec ceux partageant une position relative similaire (au sein des détections). Ainsi, les patchs représentés de manière cohérente, i.e. avec des patchs partageant la même position relative, sont favorisés par rapport à des patchs représentés par des éléments à des positions variées. Selon [55], une telle stratégie a pour avantage de pénaliser bien plus efficacement les patchs non fiables et amène à une localisation plus précise et robuste de la cible.

Dans cette section, nous proposons de nous inspirer de l'approche proposée par l'article [55] en étudiant l'influence de l'emploi de descriptions locales au sein de notre méthode. Différentes variantes de descriptions locales, basées sur des grilles régulières ou des points d'intérêt, sont examinées et l'accent est mis principalement sur la façon d'exploiter de telles descriptions au sein des représentations parcimonieuses et des valeurs d'affinité associées.

(a) (b) (c)

FIGURE III.13 – Exemples de descriptions des cibles. (a) : description holistique, qui décrit la cible dans son ensemble. (b) : description locale avec patchs pris sur une grille régulière. (c) : description locale avec patchs pris autour de points d'intérêt.

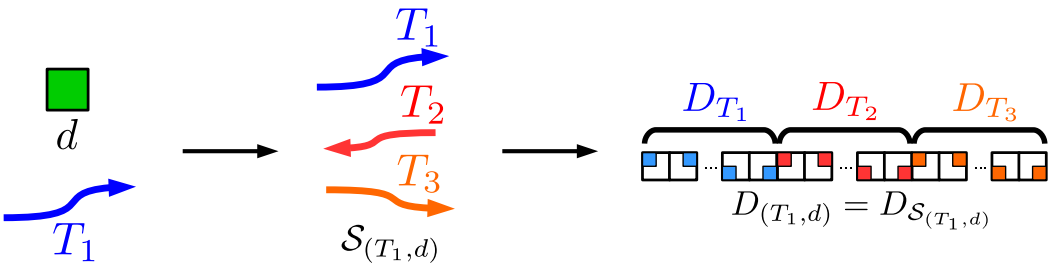
III.4.2 Descriptions locales des cibles et affinités associées

Dans la section III.3, les descriptions des cibles sont toujours considérées holistiques, c'est-à-dire globales. Nous précisons maintenant comment définir les affinités $a(T, d)$ pour tout couple trajectoire-détection (T, d) avec des représentations parcimonieuses collaboratives dans le cas de descriptions locales des cibles. Dans ce contexte, chaque détection d n'est plus associée à une unique caractéristique $y_d \in \mathbb{R}^m$ mais à un ensemble de p caractéristiques locales $(y_d^k)_{k=1..p} \in (\mathbb{R}^m)^p$. Ces caractéristiques locales décrivent chacune un patch local particulier au sein de la boîte englobante associée à la détection d.

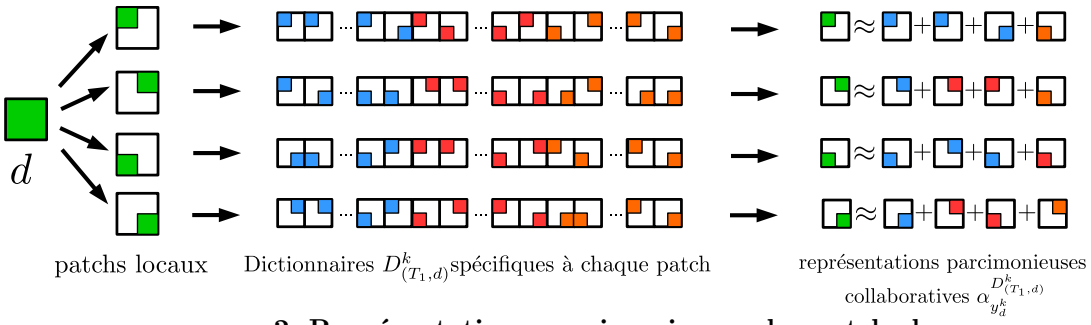
Dans les travaux récents de suivi utilisant des descriptions locales, il est possible de distinguer en particulier trois grandes stratégies pour choisir les patchs locaux les plus intéressants pour décrire les cibles. La première est d'employer une grille régulière avec des patchs de taille fixe, en autorisant éventuellement des recouvrements partiels des patchs entre eux. Une autre possibilité usuelle est d'utiliser des patchs localisés au niveau de points d'intérêt (dont la position est fréquemment déterminée par un détecteur de coins de type Harris [47]). Un dernier choix possible est d'exploiter directement des patchs locaux trouvés par le détecteur d'objets pour localiser les détections si celui-ci utilise déjà une certaine description locale de la catégorie d'objets considérée. C'est notamment le cas pour les détecteurs de type DPM [39] qui ont été particulièrement utilisés pour la détection de personnes. Nous ne considérerons pas ici cette possibilité, afin de garder une approche complètement indépendante du détecteur d'objets utilisé, et nous utiliserons donc des descriptions locales à base de points d'intérêt ou définies à partir de grilles régulières. Ces deux possibilités sont illustrées en figure III.13.

Inspiré de la méthode de suivi mono-objet [55], nous allons définir les valeurs d'affinité $a(T, d)$ pour tout couple trajectoire-détection (T, d) en exploitant des reconstructions parcimonieuses pour chaque caractéristique locale y_d^k de d. Contrairement au cas d'une description holistique des cibles, où une seule erreur de reconstruction est considérée, l'ensemble des erreurs de reconstructions des caractéristiques locales va être utilisé pour définir cette valeur d'affinité comme illustré à la figure III.14. À la section précédente, dans le cas d'une description holistique, les affinités $a(T, d)$ sont définies par l'équation (III.8) :

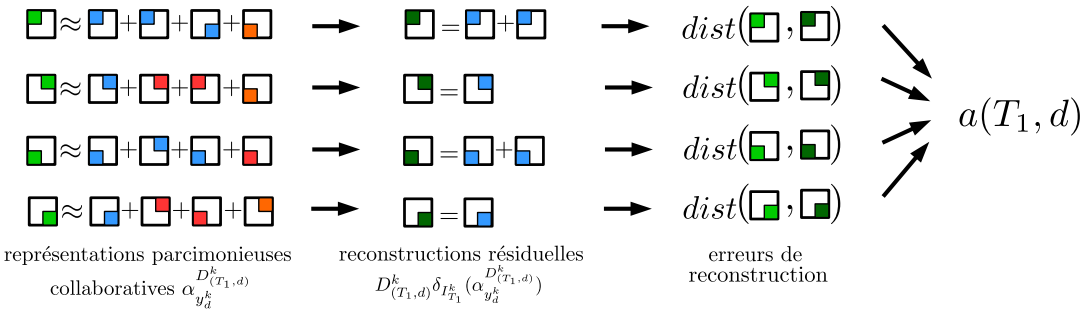
$$a(T, d) = -\frac{1}{2}||y_d - D_{(T,d)}\delta_{I_T}(\alpha_{y_d}^{D_{(T,d)}})||_2^2.$$



1. Dictionnaire collaboratif considéré à partir de patchs locaux des trajectoires



2. Représentations parcimonieuses des patchs locaux



3. Valeur d'affinité finale définie à partir des erreurs de reconstruction résiduelle des patchs locaux

FIGURE III.14 – Principe général de l'approche proposée pour définir les affinités, entre détections et trajectoires, dans le cas de descriptions locales. La valeur d'affinité $a(T_1, d)$ est définie en considérant une représentation parcimonieuse par patch local de la détection d.

Nous étendons ici cette équation (III.8) au cas d'une description locale et définissons les valeurs d'affinités à partir de la formule :

$$a(T,d) = -\sum_{k=1}^{p} \frac{1}{2} ||y^k_d - D^k_{(T,d)} \delta_{I^k_T}(\alpha^{D^k_{(T,d)}}_{y^k_d})||^2_2. \tag{III.22}$$

Les principaux éléments de cette équation sont alors les suivants :

- $(y^k_d)_{k=1..p}$: caractéristiques locales de la détection représentée d.
- $D^k_{(T,d)}$: dictionnaire spécifique au couple (T, d) et à y^k_d.
- $\alpha^{D^k_{(T,d)}}_{y^k_d}$: représentation parcimonieuse de y^k_d vis-à-vis de $D^k_{(T,d)}$.
- $D^k_{(T,d)}\delta_{I^k_T}(\alpha^{D^k_{(T,d)}}_{y^k_d})$: reconstruction résiduelle de y^k_d en ne faisant participer que les éléments de $D^k_{(T,d)}$ correspondant aux indices de l'ensemble I^k_T.

III.4.3 Considérations spatiales pour les représentations

La formule proposée en équation (III.22) pour calculer les affinités pour les couples (T, d) nécessite de fixer le dictionnaire $D^k_{(T,d)}$ employé pour la reconstruction de la caractéristique locale y^k_d, ainsi que l'ensemble I^k_T employé pour la reconstruction résiduelle. La caractéristique locale y^k_d modélise une sous-partie de la détection d et est donc localisée au sein de cette boîte. L'idée principale est ici de considérer cette localisation particulière pour définir $D^k_{(T,d)}$ et I^k_T.

Tout d'abord, nous allons considérer que chaque dictionnaire D_T, spécifique à une trajectoire T, n'est plus constitué des caractéristiques des détections récentes de la trajectoire T mais est à la place constitué de toutes les caractéristiques locales de ces détections (au plus n_{dict}). De la même manière qu'à la section précédente, lorsque $\mathcal{S}$ désigne un ensemble de trajectoires $D_\mathcal{S}$ désignera le dictionnaire commun à l'ensemble de ces trajectoires (qui sera composé des éléments de leur dictionnaire spécifique). Il est alors possible de considérer pour un couple trajectoire-détection (T, d) un dictionnaire commun $D_{\mathcal{S}(T,d)}$ où $\mathcal{S}(T, d)$ est un ensemble de trajectoires dépendant de (T, d). $\mathcal{S}(T, d)$ peut être choisi de manière à n'inclure que T (**TSSR**), uniquement les trajectoires proches $\{T,\ (T, d) \in \mathcal{L}\}$ (**LSCR**), ou bien encore l'ensemble des trajectoires estimées $\mathcal{T}$ (**GSCR**) comme détaillé à la sous-section III.3.1. Cependant, ces considérations ne s'effectuent qu'au niveau des trajectoires sans prendre en considération la localisation de la caractéristique locale représentée.

Afin de définir les dictionnaires $D^k_{(T,d)}$ et les ensembles d'indices I^k_T qui seront spécifiques à la caractéristique locale y^k_d, on considère la transformation $\mathcal{F}_k$ qui va éliminer certains éléments d'un dictionnaire ou d'un ensemble d'indices, comme illustré en figure III.15. En pratique, cette application élimine les éléments associés à un patch local dont la position relative dans la boîte de sa détection sera trop éloignée de la position relative du patch associé à y^k_d, lorsque la distance Euclidienne entre ces deux positions relatives sera supérieure à une distance d_{loc}. Cette transformation a ainsi pour effet de ne conserver que les éléments cohérents d'un point de vue spatial (au sein de leur détection respective) avec la caractéristique locale y^k_d.

Pour définir les dictionnaires $D^k_{(T,d)}$ et les ensembles d'indices I^k_T qui seront utilisés dans l'erreur de reconstruction résiduelle, on propose de considérer initialement les dictionnaires et ensembles $D_{\mathcal{S}(T,d)}$ et I_T (ensemble des indices des éléments de $D_{\mathcal{S}(T,d)}$ associés à la trajectoire T) comme fait à la section précédente dans le cas de descriptions holistiques. Ensuite, ce dictionnaire $D_{\mathcal{S}(T,d)}$ et cet ensemble I_T sont éventuellement restreints, avec des considérations spatiales, en utilisant la transformation $\mathcal{F}_k$. On va ainsi, selon les différents cas de figure, contraindre le dictionnaire à n'incorporer que des éléments dont la position relative est proche de celle de y^k_d et éventuellement contraindre à n'effectuer la reconstruction résiduelle de y^k_d qu'avec ces seuls éléments. Appliquer ou non $\mathcal{F}_k$ sur le dictionnaire ou l'ensemble considéré pour la reconstruction résiduelle mène à quatre possibilités. Deux d'entre elles sont identiques, ce qui nous laisse seulement trois cas envisageables :

(i) Le premier choix possible est de considérer simplement

$$D^k_{(T,d)} = D_{\mathcal{S}(T,d)}\,, \tag{III.23}$$

$$I^k_T = I_T\,. \tag{III.24}$$

Cela signifie qu'aucune considération spatiale n'est employée pour le dictionnaire $D^k_{(T,d)}$ ou pour l'erreur de reconstruction résiduelle effectuée sur les indices

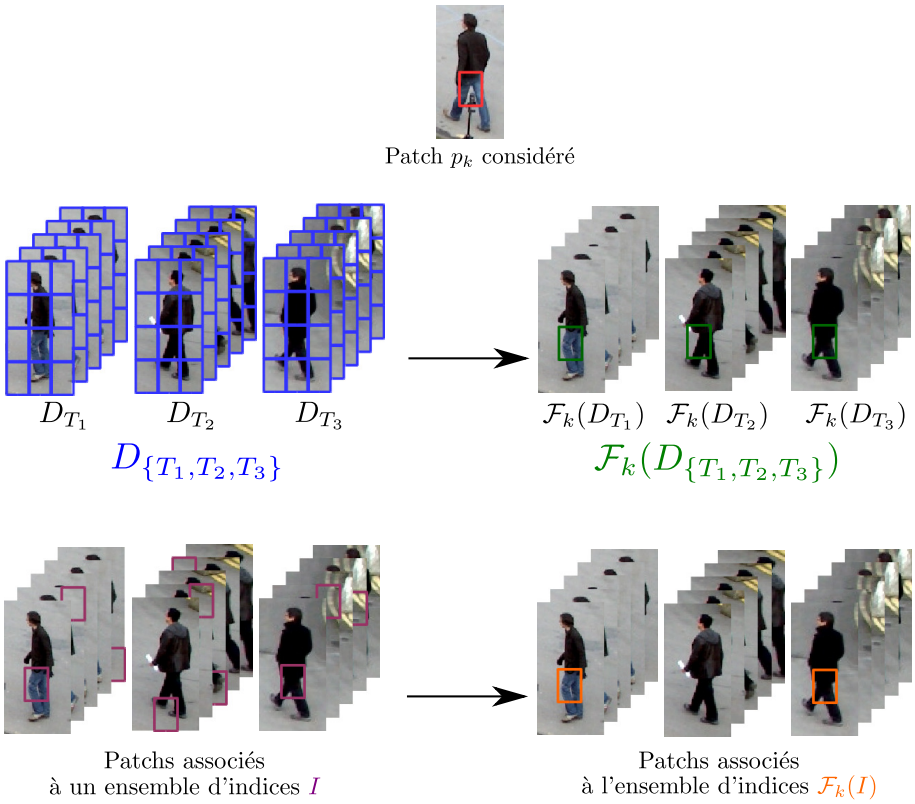


FIGURE III.15 – Effets de l'application $\mathcal{F}_k$ sur les dictionnaires D et les ensembles d'indices I. **Première ligne :** patch p_k associé à l'application $\mathcal{F}_k$. **Seconde ligne :** Dictionnaire $D_{\{T_1,T_2,T_3\}}$ composés des patchs de plusieurs vues récentes de chaque trajectoires (dans le cas d'une description locale par grille régulière). Dictionnaire $\mathcal{F}_k(D_{\{T_1,T_2,T_3\}})$ obtenu en ne retenant que les patchs de position similaire à p_k. **Troisième ligne :** Patchs associés à un ensembles d'indices I et patchs associés à l'ensemble d'indice $\mathcal{F}_k(I)$, obtenu en ne conservant que les indices des patchs de position similaire à celle de p_k.

de I_T^k. Toutes les caractéristiques locales peuvent participer à la représentation de y_d^k et toutes les caractéristiques locales des détections appartenant à la trajectoire T sont considérées pour la reconstruction résiduelle. Cette variante sera ainsi appelée **NSL** (*Non-Spatial Local description*).

(ii) Une seconde possibilité est d'envisager cette fois

$$D_{(T,d)}^k = \mathcal{F}_k(D_{\mathcal{S}(T,d)})\,, \tag{III.25}$$

$$I_T^k = I_T\,. \tag{III.26}$$

Dans ce cas, le dictionnaire est restreint aux éléments dont la position relative est proche de celle du patch associé à la caractéristique locale y_d^k. Chaque caractéristique locale est ainsi représentée uniquement par les caractéristiques qui sont positionnées de façon similaire dans leur détection respective. La reconstruction résiduelle s'effectue alors sur les éléments associés à T, et on a ici $I_T^k = I_T = \mathcal{F}_k(I_T)$ puisque tous les éléments du dictionnaire ont une position relative proche de celle de y_d^k. Cette variante est alors dénommée **SSL** (*Spatially Static Local description*).

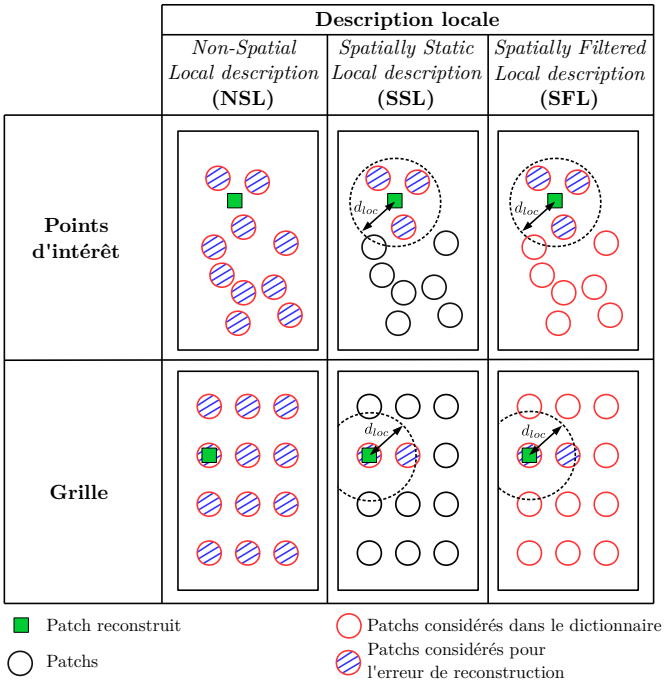


FIGURE III.16 – Descriptions locales considérées pour les représentations parcimonieuses, avec éventuellement prise en compte d'une information spatiale au sein du dictionnaire et de l'erreur de reconstruction.

(iii) Le dernier choix est intermédiaire entre les approches précédentes. On considère en effet ici

$$D^k_{(T,d)} = D_{\mathcal{S}(T,d)}\,, \tag{III.27}$$

$$I^k_T = \mathcal{F}_k(I_T)\,. \tag{III.28}$$

Cela signifie que toutes les caractéristiques locales sont conservées dans le dictionnaire mais que seules les caractéristiques locales dont la position relative est proche de celle du patch associé à la caractéristique locale y^k_d sont considérées lors de la reconstruction résiduelle. Cela revient ainsi à pénaliser les éléments locaux qui seraient représentés par des éléments localisés trop différemment au sein des détections, puisque ceux-ci ne participent pas à la reconstruction résiduelle considérée au final. Cette approche est appelée **SFL** (*Spatially Filtered Local description*).

Ces trois possibilités sont détaillées en figure III.16, où les éléments locaux, sélectionnés au sein des dictionnaires et impliqués au sein des reconstructions résiduelles, sont explicités.

III.4.4 Évaluations et analyse des résultats

Implémentation

L'approche proposée ici est implémentée en C++ et évaluée avec un CPU à 2.7 GHz en employant un seul coeur, de manière similaire à l'approche précédente présentée en section III.3.

Concernant les descriptions locales, toutes les détections sont redimensionnées à une taille de 64×128 pixels sur lesquelles un certain nombre de patchs de dimension

32×32 sont considérés. Dans le cas d'une description locale basée sur une grille, celle-ci sera de 3×4 patchs répartis régulièrement au sein de la détection. Dans le cas de points d'intérêt, nous utilisons un détecteur de coins de Harris afin de déterminer les positions les plus pertinentes avec une étape de non-maxima suppression pour éviter d'obtenir des positions de points d'intérêt trop proches. Dans les expérimentations effectuées, nous utilisons directement les valeurs d'intensité RGB normalisées des patchs.

Tout comme l'approche présentée à la section précédente, qui exploite une description holistique des cibles, nous calculons les représentations parcimonieuses avec une méthode de gradient proximal accélérée et avec une stratégie à base d'ensembles actifs. Afin de réduire le temps de calcul, nous considérons comme caractéristiques locales les valeurs d'intensités des patchs redimensionnés à une taille de 16×16 pixels. Concernant l'optimisation avec ensembles actifs, les choix faits précédemment, décrits en sous-section III.3.3, sont utilisés de nouveau. Le nombre d'étapes d'agrandissement de l'ensemble actif est limité à dix, en ajoutant au plus dix nouveaux indices à chacune de ces étapes. L'algorithme FISTA est employé entre chacune de ces étapes d'agrandissement de l'ensemble actif pour résoudre le sous-problème sur l'ensemble actif courant $\mathcal{A}$ en se limitant à dix itérations.

Protocole expérimental

Les différentes possibilités de représentations parcimonieuses (**NSL**, **SSL** ou **SFL**) peuvent être combinées avec les considérations prises au niveau des trajectoires envisagées à la sous-section III.3.1 (**TSSR**, **LSCR** ou **GSCR**). On aboutit ainsi à neuf possibilités de représentations parcimonieuses pour déterminer les valeurs d'affinité $a(T, d)$, qui peuvent être évaluées à la fois dans le cas de descriptions locales des cibles avec points d'intérêt ou avec une grille régulière. Au final, nous comparons ces 18 variantes aux trois étudiées précédemment (**TSSR**, **LSCR** ou **GSCR**) dans le cas d'une description holistique des cibles. Pour ces trois cas particuliers, nous considérons alors des valeurs d'intensité RGB normalisées des détections redimensionnées à une taille de 32×32 pixels, ce qui modifie très légérement les résultats de la section précédente. Ce changement de dimension est effectué de manière à ce que l'image considérée dans les descriptions holistiques soit redimmensionnée à une même dimension que celles des patchs locaux dans les descriptions locales.

Nous suivons le même protocole expérimental que celui auparavant présenté en sous-section III.3.3. Nous évaluons ainsi les 21 variantes proposées, avec les métriques CLEARMOT, sur quatre séquences vidéos usuelles, PETS S2L1, PETS S2L2, TownCenter et ParkingLot. Pour chaque variante, une seule configuration de paramètres est utilisée sur l'ensemble des séquences. Nous utilisons la même configuration de paramètres pour l'ensemble des variantes, déterminée manuellement de manière empirique, avec en particulier $\lambda = 0.1$ et $n_{dict} = 30$, sauf dans le cas du paramètre θ dont la valeur optimale dépend fortement du type de représentations considéré. Ce paramètre est donc déterminé indépendamment pour chaque variante pour permettre une comparaison plus juste de ces méthodes. Pour chaque variante, le paramètre θ optimal est sélectionné en testant, automatiquement, plusieurs valeurs possibles également distribuées sur l'intervalle $[-0.5, 0]$.

Description		TSSR	LSCR	GSCR
Holistique		0.615	0.622	0.627
Points d'intérêt	**NSL**	0.588	0.613	0.626
	SSL	0.600	0.618	0.631
	SFL	0.624	0.624	**0.634**
Grille	**NSL**	0.581	0.609	0.622
	SSL	0.597	0.616	0.629
	SFL	0.627	0.623	0.627

Tableau III.4 – Valeurs moyennes en MOTA. Meilleure valeur en gras et rouge, seconde meilleure soulignée en bleu.

Description		TSSR	LSCR	GSCR
Holistique		129.0	127.5	125.7
Points d'intérêt	**NSL**	157.5	118.1	107.2
	SSL	135.2	122.8	104.7
	SFL	116.4	107.4	**101.8**
Grille	**NSL**	172.7	137.1	114.8
	SSL	147.2	127.0	108.2
	SFL	119.8	110.8	107.5

Tableau III.5 – Valeurs moyennes en changements d'identité (IDS). Meilleure valeur en gras et rouge, seconde meilleure soulignée en bleu.

Comparaison entre les approches proposées

Certaines métriques CLEARMOT pour l'ensemble des types de représentations proposés sont indiquées dans les tableaux III.4 et III.5. Seulement deux des principales métriques CLEARMOT sont utilisées afin de garder les résultats lisibles, à savoir la valeur en MOTA et le nombre de changements d'identité (IDS). Chacune de ces valeurs correspond à une moyenne sur l'ensemble de sept scénarios, chaque scénario combinant une des quatre vidéos utilisées avec un jeu de détections publiques, comme explicité au tableau III.6. Pour chacun de ces tableaux, les colonnes (**TSSR**, **LSCR** ou **GSCR**) indiquent le choix effectué au niveau des cibles pour les représentations (plus ou moins collaboratives entre cibles). Les différentes lignes indiquent le type de description des cibles (holistique, locale avec points d'intérêt ou locale avec grille) ainsi que les considérations effectuées pour les représentations des descriptions locales (**NSL**, **SSL** ou **SFL**). Ainsi, chaque case des ces tableaux correspond à un type de représentations particulier pour un type de descriptions. Par exemple, la valeur en MOTA pour une description locale avec points d'intérêt, des représentations collaboratives globales entre cibles (**GSCR**) et un filtrage spatial pour les erreurs résiduelles (**SFL**) vaut 0.634 et est indiquée en rouge dans le tableau III.4.

Trois observations principales peuvent alors être faites à la lecture de ces résultats :

(i) Tout d'abord, on retrouve la tendance indiquée à la sous-section III.3.3 pour les descriptions holistiques, à savoir que les représentations collaboratives faisant intervenir le plus de trajectoires (**GSCR**) donnent de meilleurs résultats. Cette tendance semble même amplifiée puisque les variations de résultats en IDS entre les représentations de type **TSSR**, **LSCR** ou **GSCR** sont plus importants dans le cas des descriptions locales.

(ii) En termes de considérations sur l'exploitation d'information spatiale dans les représentations des caractéristiques locales, on observe une tendance assez nette qui donne un avantage certain à l'emploi de considérations sur la localisation relative de ces caractéristiques locales (**SSL** ou **SFL**). Cependant, les meilleurs résultats sont atteints dans la grande majorité des cas en n'exploitant ces considérations spatiales qu'au niveau de la reconstruction résiduelle (**SFL**), c'est-à-dire en incluant toutes les caractéristiques locales dans les dic-

tionnaires mais en limitant les reconstructions résiduelles aux éléments dont la position relative est similaire à celle de l'élément représenté.

(iii) Concernant le type de descriptions, les descriptions locales permettent d'atteindre de meilleurs résultats comparées aux descriptions holistiques. Parmi les descriptions locales, les descriptions à base de points d'intérêt sont alors plus performantes que celles utilisant une grille régulière.

Nous retrouvons certaines observations faites dans l'approche de suivi mono-objet [55]. Selon cet article, il est préférable d'employer des représentations collaboratives avec un dictionnaire qui inclut l'ensemble des caractéristiques locales puis effectuer des reconstructions résiduelles ne prenant en compte que les caractéristiques partageant la même position avec l'élément reconstruit. Une explication avancée par cet article est qu'une telle approche permet de pénaliser naturellement les patchs locaux non fiables, ceux-ci étant à priori reconstruits avec des patchs de toutes les positions et présentant donc une mauvaise reconstruction résiduelle. Cette explication reste valide dans notre contexte de suivi multi-objets et reste cohérente avec les observations faites en termes de représentations au niveau des trajectoires (**TSSR**, **LSCR** ou **GSCR**). En effet, les patchs locaux non fiables auront tendance à être reconstruits par des éléments provenant aussi de trajectoires quelconques et présenteront donc des reconstructions résiduelles très mauvaises vis-à-vis des trajectoires proches.

Des évaluations supplémentaires avec des caractéristiques visuelles différentes (HOG, LBP, histogrammes de couleurs) sont données en annexe A. On retrouve alors les deux premières tendances (i) et (ii) observées ici, c'est-à-dire que l'on gagne en performances avec des représentations collaboratives et en prenant en compte une information spatiale des patchs. Le point (iii) n'est cependant pas toujours vérifié avec certaines caractéristiques visuelles.

Comparaison à d'autres approches de l'état de l'art

L'approche proposée avec les types de représentations les plus performants, c'est-à-dire la variante avec description locale à base de points d'intérêt et les représentations de type **GSCR** et **SFL**, est dénommée **CSSR** (*Collaboration and Spatialization for Sparse Representation based tracking*). Nous comparons cette approche à la méthode proposée à la section précédente (**GSCR**) ainsi que certaines autres approches de suivi en ligne et indiquons les différents résultats de ces méthodes dans le tableau III.6. Quelques exemples de trajectoires estimées par cette nouvelle approche sont affichées en figure III.17.

Par rapport à l'approche **GSCR**, la méthode proposée ici (**CSSR**) présente moins de changements d'identité (IDS) sur l'ensemble des vidéos. Sur les vidéos PETS S2L2 et TownCenter cette réduction du nombre de changements d'identité est significative. Concernant le MOTA, par rapport à la méthode **GSCR**, cette nouvelle méthode donne des résultats dans tous les cas comparables (parfois très légèrement inférieurs) mais avec des gains significatifs sur certaines vidéos et jeux de détections comme par exemple sur PETS S2L1 avec les détections données par [134] et sur PETS S2L2 avec les détections données par [88].

Par rapport aux autres méthodes de l'état de l'art avec lesquelles nous avons pu nous comparer [13, 94, 112, 134], l'approche **CSSR** donne dans tous les cas de meilleurs résultats en MOTA et présente dans la plupart des cas une réduction très

Vidéo	Det.	Méth.	MOTA	IDS	MOTP	FP	FN
S2L1	**[88]**	**[134]**	0.699	35	**0.712**	805	**557**
		GSCR	0.702	25	0.656	**716**	641
		CSSR	**0.707**	**19**	0.656	732	606
	[134]	**[134]**	0.700	21	0.717	543	**827**
		GSCR	0.711	20	**0.731**	461	857
		CSSR	**0.725**	**18**	**0.731**	**419**	838
S2L2	**[88]**	**[134]**	0.431	347	**0.695**	1318	**4189**
		GSCR	0.407	230	0.660	1553	4292
		CSSR	**0.438**	**177**	0.661	**1316**	4263
	[134]	**[134]**	0.393	287	0.690	1416	4536
		GSCR	**0.437**	191	0.711	1056	**4526**
		CSSR	0.436	**164**	**0.712**	**872**	4743
Town Center	**[134]**	**[134]**	0.607	212	0.712	7295	**20549**
		GSCR	**0.613**	193	**0.716**	**3984**	23472
		CSSR	0.612	**157**	**0.716**	4053	23487
	[13]	**[134]**	0.634	446	0.745	9359	16302
		[13]	0.613	318	**0.805**	12309	**14982**
		GSCR	0.660	204	0.748	6784	17286
		CSSR	**0.666**	**162**	0.748	**6636**	17065
Parking Lot	**[112]**	**[94]***	0.845	**4**	0.732	-	-
		[112]*	0.793	-	**0.741**	-	-
		GSCR	**0.856**	17	0.713	266	774
		CSSR	0.854	16	0.712	287	771

Tableau III.6 – Métriques CLEARMOT sur différentes vidéos (meilleures valeurs en gras et rouge pour le MOTA et les IDS). Seconde colonne : détections employées. Le signe * indique que les scores ont été directement repris à partir des articles associés.

importante du nombre de changements d'identités (IDS) sauf sur la vidéo ParkingLot (où le nombre de changements d'identité reste stable).

Temps d'exécution

Le temps d'exécution est la principale limitation de l'approche proposée ici. En effet, dans le cas d'une description locale et de représentations parcimonieuses collaboratives de type **GSCR** et **SFL**, les dictionnaires impliqués sont composés d'un nombre bien plus important d'éléments comparé à l'approche **GSCR** avec une description holistique des cibles. Toutes les caractéristiques locales de chaque description étant incluses dans le dictionnaire, le nombre d'éléments est multiplié d'un facteur p (le nombre de patchs par détection, typiquement de l'ordre de la dizaine) par rapport à la méthode **GSCR**. De plus, il est nécessaire de calculer non pas une représentation parcimonieuse par détection mais une par patch local considéré, ce qui multiplie d'un facteur p le nombre de représentations parcimonieuses à calculer. Pour ces différentes raisons, la stratégie d'optimisation par ensembles actifs n'est plus suffisante pour permettre un fonctionnement proche du temps réel. La méthode **CSSR**, en exploitant un seul coeur d'un CPU à 2.7 GHz et les détections de [134], fonctionne ainsi à 2.7 images par seconde pour la vidéo PETS S2L1 et à seulement

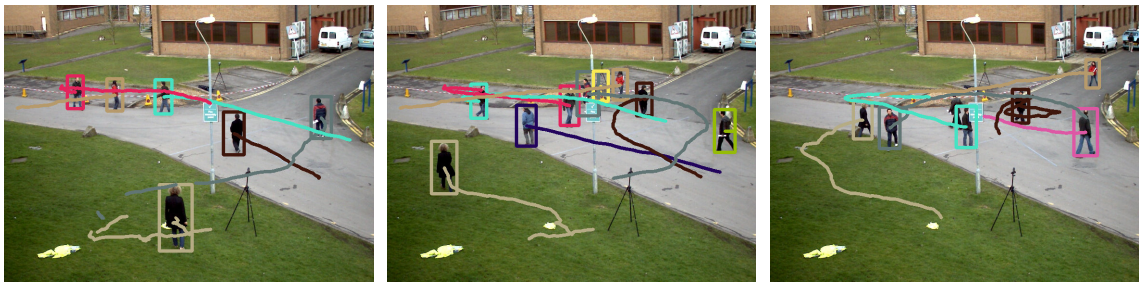


Image n°650 | **Image n°700** | **Image n°750**

S2L1

Image n°1500 | **Image n°1750** | **Image n°2000**

Town Center

FIGURE III.17 – Exemples de certaines trajectoires estimées par notre approche **CSSR** pour les vidéos PETS S2L1 et TownCenter, en utilisant respectivement les détections de [88] et [13].

0.4 images par seconde pour la vidéo PETS S2L2.

Bien qu'il soit possible d'effectuer en parallèle le calcul des représentations parcimonieuses sur une même image, le temps d'exécution reste une limitation importante de cette approche (notamment pour les scènes avec une densité élevée de cibles). Une perspective serait d'envisager l'emploi de techniques d'acquisition comprimée (*compressed sensing*) pour accélérer le calcul de ces représentations, technique qui a été employée en suivi mono-objet [73].

Conclusion

Dans ce chapitre, une approche de suivi multi-objets en ligne, qui exploite des représentations parcimonieuses pour définir les valeurs d'affinité entre les trajectoires et les détections, a été proposée. Plusieurs types de représentations, plus ou moins collaboratives entre les cibles, ont été étudiés. Ces investigations indiquent que des représentations collaboratives globales, entre toutes les cibles, donnent de meilleures performances de suivi. Malgré le fait que ces représentations font intervenir des dictionnaires contenant un grand nombre d'éléments, nous avons montré que des considérations sur l'optimisation de ces représentations parcimonieuses permettent néanmoins d'aboutir à une méthode avec une vitesse de fonctionnement proche du temps réel. Des évaluations quantitatives indiquent de plus que l'approche proposée se compare très favorablement par rapport à d'autres approches de suivi récentes.

Une extension de cette approche a été de plus proposée pour exploiter des descriptions locales des cibles et envisager des représentations parcimonieuses collaboratives à la fois entre les cibles et les patchs locaux de ces dernières. Plusieurs

façons d'employer de telles descriptions pour définir des valeurs d'affinité à base de représentations parcimonieuses ont alors été explorées. Utiliser des représentations parcimonieuses à la fois collaboratives au niveau des cibles mais aussi au niveau des descriptions locales, en prenant en compte des considérations spatiales pour les erreurs de reconstruction résiduelles, permet alors de gagner de nouveau en performances notamment en termes de changements d'identité (IDS). Ce gain s'effectue néanmoins au prix d'une dégradation de la vitesse de fonctionnement de l'approche, un nombre bien plus important de représentations devant être calculées dans ce cas.

La première contribution de ce chapitre, à savoir employer des représentations parcimonieuses collaboratives globales dans un système de suivi multi-objets en ligne, a été publiée dans IEEE ICIP 2015 [36]. La seconde contribution de ce chapitre, qui consiste à étudier de quelles façons des représentations parcimonieuses collaboratives peuvent tirer avantage d'une description locale des cibles, a permis une publication dans IEEE AVSS 2015 [38].

Chapitre IV

Suivi par fenêtre glissante et représentations structurées en norme $l_{\infty,1}$

Sommaire

Introduction

Nous proposons d'étendre l'approche proposée dans le chapitre précédent, qui exploite des représentations parcimonieuses collaboratives image après image, en raisonnant ici sur plusieurs images successives. L'objectif de ce chapitre est donc d'étudier comment des représentations parcimonieuses collaboratives peuvent être

combinées judicieusement au sein d'une méthode de suivi avec une fenêtre temporelle glissante. La méthode proposée tire alors avantage à la fois de l'information temporelle supplémentaire au sein de la fenêtre glissante et de représentations parcimonieuses structurées adaptées à une telle approche.

La sous-section IV.1.1 précise tout d'abord les principales motivations qui nous incitent à envisager une approche de suivi à fenêtre glissante. Nous indiquerons en particulier pour quelles raisons notre choix s'est porté sur une approche dont l'étape d'association de données, entre les détections et les trajectoires, est réalisée par une méthode de Monte-Carlo par chaînes de Markov (MCMC).

Le principe général de cette nouvelle méthode est ensuite détaillé en section IV.2, en insistant sur l'énergie globale considérée pour résoudre l'association de données multi-images qui se pose au sein de la fenêtre glissante. Cette énergie exploitera notamment des représentations parcimonieuses de chaque détection de la fenêtre glissante afin d'améliorer l'étape d'association de données.

Au sein de la section IV.3 nous nous intéressons particulièrement au type de représentations parcimonieuses qui serait le plus adapté pour ce problème de suivi à fenêtre glissante. Cette réflexion nous amène à proposer l'emploi de représentations parcimonieuses structurées déduites d'une norme $l_{\infty,1}$ pondérée. Nous indiquons aussi comment les méthodes d'optimisation par gradient proximal avec ensembles actifs, utilisées au chapitre précédent, peuvent être adaptées ici pour calculer efficacement ces représentations.

La dernière section IV.4 est dédiée à l'évaluation de cette nouvelle approche sur les bases de données du *MOTChallenge*. Des évaluations quantitatives montrent les gains en performances liés à l'emploi des représentations parcimonieuses structurées proposées et permettent de vérifier que cette méthode est compétitive par rapport aux autres approches récentes de l'état de l'art.

IV.1 Motivations

Les principales motivations qui nous amènent à proposer une approche de suivi à fenêtre glissante, en exploitant des représentations parcimonieuses des détections, sont détaillées ici. Nous justifions aussi plus particulièrement le choix d'employer une méthode d'association de données par échantillonnage de type MCMC (ou MCMCDA pour *Markov Chain Monte Carlo Data Association*, [99]).

IV.1.1 Limitations du suivi multi-objets en ligne

Les approches de suivi en ligne, comme expliqué en sous-section III.1.1, présentent l'avantage d'avoir un délai de latence très faible (sous réserve de moyens de calculs adaptés). Cette caractéristique, ainsi que leurs performances proches des méthodes hors ligne, sont les principales raisons qui nous ont amenés à considérer initialement une approche de suivi en ligne au chapitre précédent. Néanmoins, ces méthodes sont limitées par leur module d'association de données effectuée en considérant uniquement les deux dernières images de la scène observée. Cela nécessite de faire des hypothèses d'association immédiatement sans pouvoir attendre davantage d'information future qui pourrait désambiguïser des situations délicates, comme des occultations ou des croisements de cibles. En pratique, ces méthodes ont tendance à produire plus d'erreurs d'association comme des changements d'identité ou des

fragmentations de trajectoires. Par exemple, sur les versions 2015 et 2016 du *MOT-Challenge*, les approches présentant le moins de changements d'identité ou le moins de fragmentation sont majoritairement des méthodes hors ligne [24, 62, 89, 90, 108] (parmi les méthodes publiées au 01/12/2016).

Les méthodes hors ligne globales, qui considèrent l'ensemble de la vidéo pour estimer les trajectoires, peuvent exploiter les images futures pour mieux analyser une situation complexe. Dans le cas de croisements de cibles, pouvoir exploiter les prochaines images sur lesquelles les cibles sont bien différenciées visuellement est par exemple particulièrement utile. Ces méthodes ont ainsi le plus souvent des performances légèrement supérieures aux méthodes en ligne, en limitant surtout les erreurs d'association. Ces méthodes globales payent cependant un lourd tribu pour ce gain en performances puisque le délai de latence est ici significatif, aucun résultat n'étant estimé avant la fin de la séquence étudiée.

Il est cependant possible de chercher un compromis entre les approches en ligne avec une latence faible mais des erreurs d'association plus nombreuses, et des approches hors ligne globales avec des erreurs d'association plus limitées au prix d'un temps de latence très important. Dans de nombreuses situations délicates, comme le croisement de cibles, l'occultation d'une cible ou encore une période de non-détection d'une cible, avoir accès à un petit nombre d'images futures est souvent suffisant pour gérer cette situation. Cela nous amène à considérer des méthodes à fenêtre glissante qui raisonnent sur un nombre limité d'images, typiquement de l'ordre d'une dizaine. Ces approches permettent ainsi de maintenir un temps de latence faible sous réserve de moyens de calculs adaptés, ce temps étant au moins égal à la durée de la fenêtre glissante considérée. Plus de détails sur les approches existantes à fenêtre glissante sont indiqués en sous-section II.2.3.

Du fait de ces observations, nous cherchons à étendre l'approche proposée au chapitre précédent en une méthode de suivi à fenêtre glissante. Notre objectif est alors d'obtenir une méthode plus robuste qui combine simultanément les concepts de fenêtre glissante temporelle et de représentations parcimonieuses pour réaliser l'étape d'association de données de façon plus précise.

IV.1.2 Formulations de l'association de données multi-images

Les méthodes possibles pour formuler et résoudre le problème d'association de données multi-images, employées par les approches de suivi hors ligne, ont été détaillées en sous-section II.1.3 et en sous-section II.2.3. Nous énumérons ici de manière plus succincte certaines de ces méthodes en nous concentrant sur celles se ramenant à une minimisation d'énergie.

L'association de données multi-images consiste à associer entre eux un ensemble d'éléments, le plus souvent constitué de détections et/ou de trajectoires, présents au sein d'une série d'images. L'objectif est alors que les éléments associés soient relatifs à une même cible. La plupart des approches de suivi hors ligne récentes formulent ce problème d'association comme un problème de minimisation d'une énergie E. Cette énergie attribue une valeur à toute configuration possible d'associations C, et la solution du problème d'association est alors une configuration C^* qui minimise l'énergie E.

Une méthode de suivi hors ligne nécessite ainsi de formuler une énergie E, qui permet de juger de la qualité des configurations d'associations, et de définir une stra-

tégie optimisant cette énergie. En fonction des choix effectués sur ces deux points, deux grandes catégories d'approches peuvent alors être distinguées. La première catégorie rassemble les méthodes qui cherchent à formuler le problème de minimisation de l'énergie E comme un problème plus générique. Ce problème de minimisation peut par exemple être vu comme un problème de flot de coût minimal [69, 84, 102, 122], de programmation linéaire à valeurs booléennes [31], de CRF (*Conditional Random Field*) [24,66], de clique de poids maximal [133] ou encore d'ensemble indépendant de poids maximal [19]. L'avantage principal de ces formulations est que ces problèmes classiques ont été extensivement étudiés et des méthodes d'optimisation efficaces, parfois même exactes, existent. Néanmoins, ces problèmes imposent le plus souvent d'importantes contraintes sur la formulation de l'énergie E. Ces contraintes peuvent rendre difficile la prise en compte de certains aspects importants du problème de suivi multi-objets comme, par exemple, les interactions entre cibles ou encore des modèles d'apparence et de mouvements élaborés.

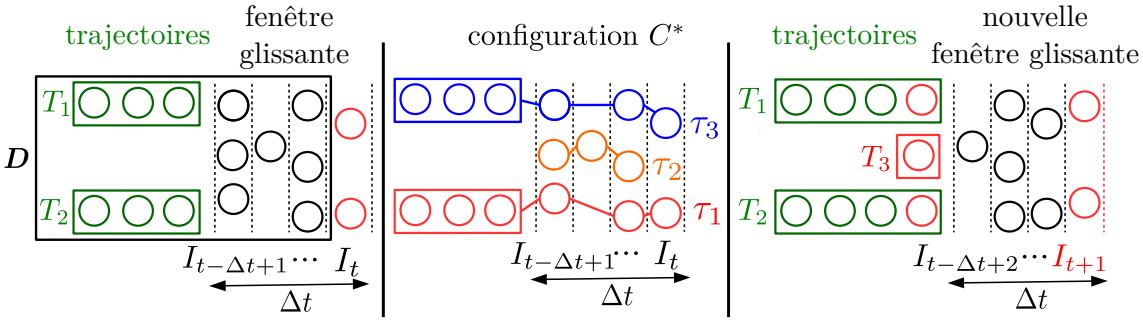
La seconde catégorie d'approches réunit les méthodes de suivi qui n'imposent pas d'hypothèses particulières sur l'énergie E à minimiser. Cependant, la formulation a priori non-convexe de l'énergie employée rend sa minimisation exacte délicate. Des solutions approchées peuvent néanmoins être déterminées par des méthodes d'optimisation assez générales qui n'imposent pas une formulation spécifique de l'énergie E. Des approches de suivi qui emploient ce genre de procédé sont par exemple les méthodes de type MHT (*Multiple Hypothesis Tracking*) [62, 106] qui utilisent une recherche exhaustive des configurations C possibles en élaguant l'arbre de recherche pour limiter le nombre de configurations envisagées. D'autres approches utilisent des méthodes MCMC [13, 43, 77, 99, 110, 132] en échantillonnant une probabilité associée à E. L'avantage de toutes ces approches est qu'elles peuvent employer des énergies E complexes, qui modélisent plus fidèlement le problème de suivi multi-objets. Leur principal désavantage est que l'optimisation est alors approchée en ayant moins de garanties d'optimalité des solutions que pour les approches formulant la minimisation de E comme un problème plus classique.

Dans ce chapitre, nous définissons une énergie E qui exploite les représentations parcimonieuses des détections présentes au sein d'une fenêtre glissante. Afin de ne pas être limité pour définir cette énergie E à partir de ces représentations parcimonieuses, nous employons une méthode de type MCMCDA, analogue à l'approche proposée dans [99], pour résoudre l'association de données.

IV.1.3 Principe de l'approche proposée

Nous employons ici une approche de suivi hors ligne à fenêtre glissante dans le but d'obtenir des résultats plus robustes tout en maintenant un temps de latence faible. L'association de données multi-images est formulée comme un problème de minimisation d'énergie E avec une énergie qui exploite des représentations parcimonieuses des détections de la fenêtre glissante. Cette association de données, entre les détections de la fenêtre glissante et les trajectoires, est alors résolue en suivant une approche de type MCMCDA [99] afin de garder une certaine liberté dans le choix de la formulation de l'énergie E.

Prendre en considération des représentations parcimonieuses collaboratives des détections a pour objectif de favoriser, pour chaque détection d, son association à un faible nombre de détections plus anciennes qui participent activement à sa



Etape 1 : Calcul des représentations parcimonieuses

Etape 2 : Optimisation de E par échantillonnage MCMC

Etape 3 : Gestion des trajectoires à $t - \Delta t + 1$

FIGURE IV.1 – Etapes suivies par l'approche de suivi proposée. Tout d'abord, les représentations parcimonieuses des dernières détections (symbolisées par des cercles) de la dernière image I_t sont calculées. Ensuite, l'énergie globale E est optimisée par échantillonnage de type MCMC, en déterminant la configuration idéale C^*. Pour finir, les trajectoires (symbolisées par des rectangles) sont définitivement estimées pour la première image $I_{t-\Delta t+1}$ de la fenêtre glissante en suivant la configuration C^*.

représentation. Un tel procédé favorise les configurations C cohérentes avec les représentations parcimonieuses des détections, c'est-à-dire les configurations C qui associent les détections qui sont cohérentes en apparence.

Nous employons de plus des représentations parcimonieuses structurées particulières, déduites d'une norme $l_{\infty,1}$ pondérée, afin de favoriser le fait que chaque détection d soit représentée uniquement par l'ensemble des détections liées à sa cible. Ainsi, regrouper les détections qui se représentent mutuellement correspondra bien à déterminer les détections liées à chaque cible. Nous détaillerons de plus comment optimiser efficacement ces représentations structurées en employant des méthodes de gradient proximal avec ensembles actifs.

IV.2 Système de suivi par fenêtre glissante employé

Cette section décrit notre système de suivi multi-objets à fenêtre glissante. Nous détaillons dans un premier temps le fonctionnement général de ce système, puis nous formulons l'énergie globale E utilisée pour l'association de données. Enfin nous décrirons l'optimisation par méthode MCMC employée pour optimiser cette énergie E.

IV.2.1 Description générale du système

Aperçu d'ensemble

Le système de suivi multi-objets proposé est une méthode hors ligne utilisant une fenêtre glissante, basée sur le paradigme de suivi par détection et utilisant donc les détections estimées par un détecteur d'objets de la catégorie des cibles. Cela signifie

qu'une fenêtre glissante, qui est composée des Δt dernières images $\{I_{t-\Delta t+1}, ..., I_t\}$, est considérée à chaque instant t ainsi que les trajectoires $\mathcal{T}_{t-\Delta t} = \{T_1, ..., T_{n_{traj}}\}$, estimées avant cette fenêtre glissante (c'est-à-dire avant l'instant $t - \Delta t + 1$). Un détecteur d'objets fournit de plus pour chaque image I de la fenêtre glissante un ensemble de détections Det_I. À partir de ces éléments, l'objectif est d'estimer les trajectoires $\mathcal{T}_{t-\Delta t+1}$ pour la dernière image $I_{t-\Delta t+1}$ de la fenêtre glissante, puis de recommencer ce procédé en décalant d'une image la fenêtre glissante, composée alors des images $\{I_{t-\Delta t+2}, ..., I_{t+1}\}$. Ce système permet ainsi d'estimer les trajectoires avec un léger retard de Δt images et cette valeur Δt sera maintenue assez faible, de l'ordre de la dizaine d'images, afin de garder un temps de latence raisonnable.

L'étape d'association de données consiste ici à déterminer un ensemble $C = \{\tau_1, \tau_2, ..., \tau_M\}$ de pistes qui prolongent les trajectoires $\mathcal{T}_{t-\Delta t+1}$ au sein de la fenêtre glissante. Dans tout ce chapitre, on désigne par *trajectoire*, notée T, les positions estimées définitivement d'une cible au-delà de la fenêtre glissante. Une *piste*, notée τ, est alors un prolongement d'une trajectoire au sein de la fenêtre glissante (ou une hypothèse de nouvelle trajectoire) qui peut être remis en cause ultérieurement.

De façon plus détaillée, les différentes étapes suivies par notre approche sont illustrées à la figure IV.1. Ces étapes sont constituées par :

(i) Le calcul des représentations parcimonieuses de l'ensemble des détections Det_{I_t} estimées sur la nouvelle image I_t.

(ii) La résolution du problème de l'association de données multi-images en exploitant ces représentations.

(iii) La gestion et le prolongement des trajectoires pour l'image $I_{t-\Delta t+1}$.

Nous précisons brièvement ces trois étapes dans ce qui suit.

Représentations parcimonieuses des dernières détections

Tout d'abord, les représentations parcimonieuses de l'ensemble Det_{I_t} des détections de la nouvelle image I_t sont calculées. Chaque détection d est alors associée à un vecteur caractéristique $y_d \in \mathbb{R}^m$, normalisé, qui modélisera la détection d. Ces représentations parcimonieuses sont définies à partir d'un dictionnaire D_t collaboratif, commun pour chaque détection de l'ensemble Det_{I_t}, qui fait intervenir les caractéristiques y_d de toutes les autres détections de la fenêtre glissante ainsi que les dernières détections des trajectoires de $\mathcal{T}_{t-\Delta t}$.

Le dictionnaire D_t s'écrit comme la concaténation de plusieurs dictionnaires spécifiques aux trajectoires ou aux images de la fenêtre glissante :

$$D_t = [D_{\mathcal{T}_{t-\Delta t}} \; D_{FG_t}] = [D_{T_1} \; ... \; D_{T_{n_{traj}}} \; D_{I_{t-\Delta t+1}} \; ... \; D_{I_{t-1}}]. \tag{IV.1}$$

Le dictionnaire D_t est ainsi issu de la concaténation entre deux dictionnaires $D_{\mathcal{T}_{t-\Delta t}}$ et D_{FG_t}, le premier étant associé aux trajectoires $\mathcal{T}_{t-\Delta t}$ tandis que le second est associé aux détections, au sein de la fenêtre glissante, des images $\{I_{t-\Delta t+1}, ..., I_{t-1}\}$. On a ainsi $D_{\mathcal{T}_{t-\Delta t}} = [D_{T_1} \; ... \; D_{T_{n_{traj}}}]$ avec D_T dictionnaire spécifique à la trajectoire T (dont les colonnes correspondent aux caractéristiques y_d des N_{tr} dernières détections associées à T). De même, $D_{FG_t} = [D_{I_{t-\Delta t+1}} \; ... \; D_{I_{t-1}}]$ avec D_I dictionnaire spécifique à l'image I (dont les colonnes correspondent aux caractéristiques y_d des détections Det_I). Dans la suite, pour simplifier les notations, on omettra éventuellement les indices temporels pour les dictionnaires sauf en cas de possible confusion. On écrira

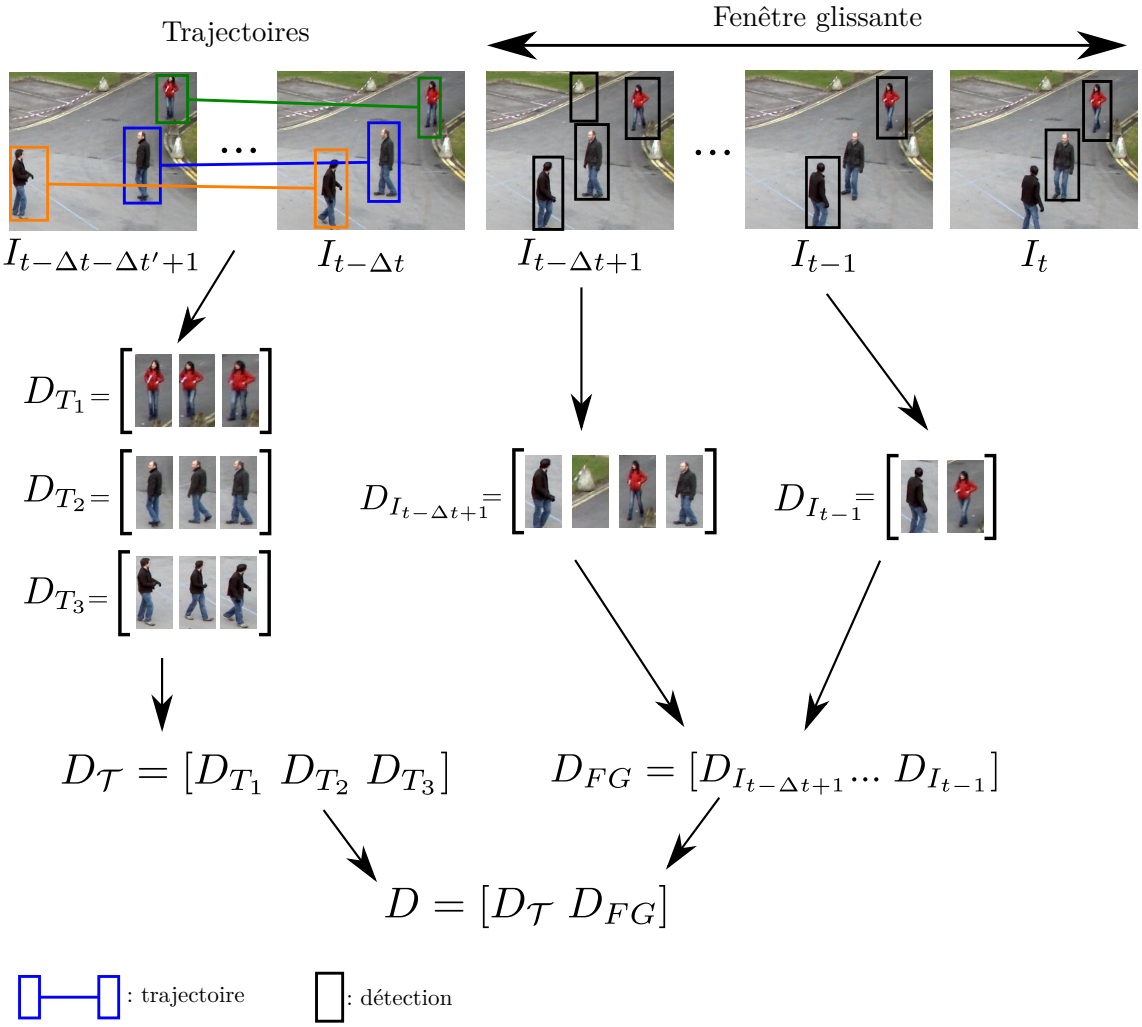


FIGURE IV.2 – Exemple de dictionnaire D considéré, en précisant les sous-dictionnaires qui le composent propres aux trajectoires et aux images de la fenêtre glissante.

donc éventuellement $D = D_t$, $D_{\mathcal{T}} = D_{\mathcal{T}_{t-\Delta t}}$ et $D_{FG} = D_{FG_t}$. Un exemple d'un tel dictionnaire est donné en figure IV.2.

On associe alors à chaque détection $d \in Det_{I_t}$ de l'image I_t une représentation parcimonieuse α_{y_d} solution du problème :

$$\min_{\alpha} \frac{1}{2}||y_d - D\alpha||_2^2 + \lambda\Omega(\alpha), \tag{IV.2}$$

où Ω est une norme favorisant une certaine parcimonie des solutions. Nous verrons dans ce chapitre que la norme couramment employée pour définir des représentations parcimonieuses, à savoir la norme l_1 que nous avons utilisée au chapitre III, n'est pas la plus adaptée dans le contexte d'un problème d'association multi-images. De ce fait, nous étudierons en section IV.3 d'autres nomes, moins usuelles dans ce contexte, et montrerons leur pertinence dans notre contexte.

Association de données multi-images

Le problème d'association de données multi-images prend en compte une fenêtre glissante de plusieurs images consécutives, $\{I_{t-\Delta t+1}, ..., I_t\}$, et un ensemble de trajec-

toires $\mathcal{T}$ estimées avant ces images (i.e. à l'instant $t - \Delta t$). Chaque image I fournit un ensemble de détections Det_I, et le problème d'association multi-images consiste à associer toutes les détections de la fenêtre glissante, c'est-à-dire des ensembles $Det_{I_{t-\Delta t+1}}, ..., Det_{I_t}$, aux trajectoires de $\mathcal{T}$. Chaque détection n'est néanmoins pas nécessairement associée à une trajectoire de $\mathcal{T}$ afin de prendre en compte les éventuelles fausses détections et les trajectoires démarrant au sein de la fenêtre glissante.

Durant cette étape d'association de données, on cherche à trouver un ensemble de pistes $C = \{\tau_1, \tau_2, ..., \tau_M\}$ où chaque piste τ est composée de détections de la fenêtre glissante (i.e. des ensembles $Det_{I_{t-\Delta t+1}}, ..., Det_{I_t}$) et, éventuellement, d'une unique trajectoire de $\mathcal{T}$. Dans le cas où une piste τ inclut une trajectoire T de $\mathcal{T}$, toute détection d incluse dans cette piste τ est supposée associée à T. Dans le cas où une piste τ n'inclut que des détections de la fenêtre glissante, cette piste constitue une hypothèse de nouvelle trajectoire démarrant au sein de la fenêtre glissante. Ainsi, l'ensemble $\{\tau_1, \tau_2, ..., \tau_M\}$ associe bien les détections de la fenêtre glissante aux trajectoires de $\mathcal{T}$. Certaines détections peuvent n'être incluses dans aucune piste τ, ces détections étant considérées comme des fausses détections.

Une configuration C est alors un ensemble de pistes $C = \{\tau_1, \tau_2, ..., \tau_M\}$ vérifiant certaines contraintes, à savoir :

(i) Toute détection d et toute trajectoire T est incluse dans au plus une piste τ.

(ii) Toute piste τ inclut au moins deux éléments dont au plus une seule trajectoire T.

(iii) Toute piste τ inclut au plus une seule détection par image (donc par pas de temps) de la fenêtre glissante.

Toute détection d est associée à une boîte x_d, de hauteur h_d, de largeur w_d et de score de détection s_d. Deux détections d et d' d'une piste τ d'une configuration C doivent alors satisfaire les contraintes supplémentaires suivantes :

(iv) $\delta t \leq \delta t_l$,

(v) $dist(x_d, x_{d'}) \leq (1 + \delta t)\, \frac{w_d + w_{d'}}{2}\, d_l$,

(vi) $|h_d - h_{d'}| \leq (1 + \delta t)\, \frac{h_d + h_{d'}}{2}\, h_l$,

où $dist(x_d, x_{d'})$ est la distance Euclidienne entre les deux centres des boîtes associées à d et d', $\delta t = |d_t - d'_t|$ correspond à l'espacement temporel entre d et d', et où δt_l, d_l, h_l sont des paramètres fixes. Ces contraintes supplémentaires ont pour effet de réduire le nombre de configurations possibles en imposant que deux détections consécutives dans une même piste soient suffisamment proches, à la fois (iv) dans le temps, (v) dans le domaine image et (vi) au niveau de leurs tailles respectives.

L'association de données est alors formulée, de manière usuelle vis-à-vis de la littérature récente, comme un problème de minimisation d'énergie qui peut s'apparenter à une approche de type *MAP* (*Maximum A-Posteriori*). On cherche une configuration C^* qui minimise une énergie E, définie à partir des représentations parcimonieuses des détections. Les associations des détections aux trajectoires sont alors déterminées à partir de cette configuration optimale C^*. L'optimisation de cette énergie est réalisée par échantillonnage de type MCMC en suivant l'approche proposée dans [99]. La formulation de cette énergie E et son optimisation sont détaillées à la suite de cette sous-section.

Gestion et prolongement des trajectoires

Les trajectoires de $\mathcal{T}_{t-\Delta t}$ sont ensuite prolongées à l'instant $t-\Delta t+1$ en suivant simplement les pistes de la configuration C^* déterminée par l'étape d'association de données. Le procédé utilisé pour prolonger les trajectoires à partir de C^* est alors le suivant.

Tout d'abord, toute trajectoire de $\mathcal{T}_{t-\Delta t}$ non incluse dans une piste τ de la configuration C^* est déclarée terminée, tandis que toute piste τ de C^* non associée à une trajectoire et démarrant par une détection à l'image $I_{t-\Delta t+1}$ entraîne éventuellement la création d'une nouvelle trajectoire. Une trajectoire est en effet créée si la piste τ est suffisamment fiable, c'est-à-dire si τ comporte au moins N_c détections avec un score de détection moyen au-dessus de la valeur s_c, avec N_c et s_c deux paramètres fixes.

Pour toute trajectoire T incluse dans une piste τ, cette trajectoire est prolongée à l'instant $t-\Delta t+1$ via le procédé suivant. Tout d'abord, on désigne par x_τ l'ensemble des boîtes estimant la position de la piste τ sur l'ensemble de la fenêtre glissante ($x_\tau(t)$ correspond ainsi à la position de la piste τ estimée pour l'image I_t). Les boîtes x_τ permettent notamment de gérer les images pour lesquelles aucune détection n'est présente dans la piste τ, ce qui est par exemple le cas lorsqu'une cible est parfois non détectée. x_τ est ici déterminé par une simple interpolation linéaire entre les détections de τ, mais des techniques plus complexes de lissage peuvent être envisagées comme effectué au chapitre V. La trajectoire $T \in \tau$ est alors étendue à l'instant $t-\Delta t+1$ en considérant la position $x_\tau(t-\Delta t+1)$.

IV.2.2 Energie globale proposée

Nous précisons ici l'énergie E utilisée pour réaliser l'association de données au sein de la fenêtre glissante. Cette énergie attribue des valeurs faibles aux configurations C les plus pertinentes vis-à-vis de modèles d'observation, d'apparence, de mouvement et d'interaction. Cette énergie est ainsi formulée comme une combinaison linéaire de quatre termes :

$$E(C) = \theta_{Ob} Ob(C) + \theta_{App} App(C) + \theta_{Mot} Mot(C) + \theta_{Int} Int(C)\,. \tag{IV.3}$$

Chacun de ces termes modélise un aspect particulier du suivi multi-objets et les valeurs θ permettent de pondérer chaque terme dans l'énergie globale. Cette forme d'énergie et ces différents termes se retrouvent, de manière plus ou moins similaire, dans plusieurs approches de suivi hors ligne comme par exemple [88, 89]. La principale différence entre l'énergie proposée ici et ces précédents travaux se situe au niveau du modèle d'apparence $App(C)$ qui vise à favoriser les configurations de pistes les plus cohérentes vis-à-vis des représentations parcimonieuses des détections. Les différents termes intervenant dans l'écriture de l'énergie E sont décrits dans cette sous-section, le modèle d'apparence $App(C)$ faisant l'objet de la sous-section IV.3.1.

Pour rappel, $x_\tau(t)$ correspond à la boîte estimée pour la piste τ à l'instant t, boîte estimée ici à partir d'une interpolation linéaire des détections de τ. Pour toute piste τ, b_τ et e_τ correspondent respectivement à l'instant du premier et dernier élément inclus dans la piste τ.

Le but du *modèle d'observation* $Ob(C)$ est de favoriser les configurations C dont les pistes τ font intervenir les détections de score élevé ainsi que les trajectoires de

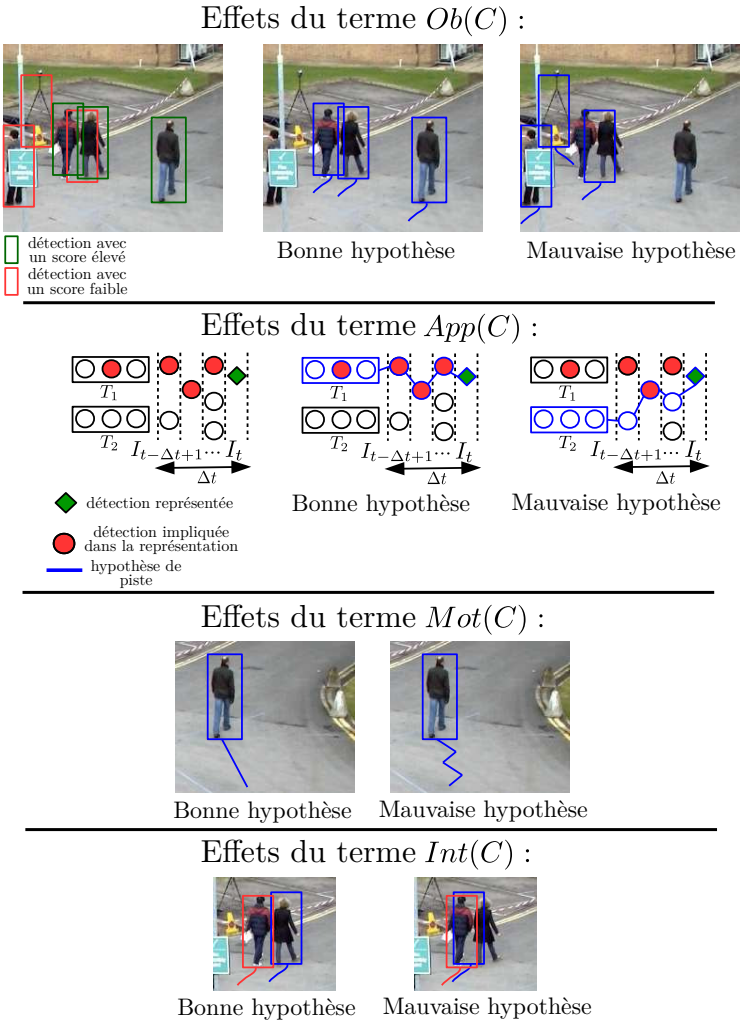


FIGURE IV.3 – Situations favorisées et pénalisées par les différents termes de l'énergie E. Le terme d'observation $Ob(C)$ favorise les configurations impliquant des détections de score élevé et le terme $App(C)$ les configurations cohérentes vis-à-vis des représentations parcimonieuses de toutes les détections. Le terme $Mot(C)$ favorise un mouvement des cibles à vélocité constante et le terme $Int(C)$ pénalise les pistes proches.

$\mathcal{T}$ avant la fenêtre glissante. Pour ce faire, ce modèle s'écrit :

$$Ob(C) = -\sum_{\tau \in C} \sum_{d \in \tau} [\alpha_{Ob} + \beta_{Ob} s_d] - \sum_{\tau \in C} \sum_{T \in \tau} \gamma_{Ob}\,, \tag{IV.4}$$

où α_{Ob}, β_{Ob} et γ_{Ob} sont des paramètres fixes positifs. Le premier terme de l'équation (IV.4) récompense la participation des détections de score s_d élevé tandis que le second terme favorise la prolongation des trajectoires auparavant estimées.

Le *modèle d'apparence* $App(C)$ exploite les représentations parcimonieuses de toutes les détections de la fenêtre glissante. Ce terme favorise les configurations où chaque détection obtient une erreur de reconstruction résiduelle faible, cette reconstruction faisant intervenir uniquement les détections appartenant à la même piste. Ce modèle tend à rassembler au sein d'une même piste les détections qui se représentent mutuellement. Plus de détails sur ce terme sont donnés à la sous-section IV.3.1.

Le *modèle de mouvement* $Mot(C)$ suppose un mouvement à vélocité constante des cibles, et est de ce fait formulé de la façon suivante :

$$Mot(C) = \sum_{\tau \in C} \sum_{t=b_\tau+1}^{e_\tau-1} \frac{1}{(x_\tau(t)_w)^2} ||x_\tau(t+1) + x_\tau(t-1) - 2x_\tau(t)||_2^2 \,, \tag{IV.5}$$

où $x_\tau(t)_w$ correspond à la largeur de la boîte $x_\tau(t)$ et est employé pour normaliser les accélérations par rapport à la taille des cibles. Ce terme $Mot(C)$ favorise un mouvement de vélocité constante en pénalisant les accélérations et décélérations des pistes. Malgré son apparente simplicité, un modèle de vélocité constante permet déjà d'aider à limiter les changements d'identité des cibles, en particulier lors de croisements entre cibles ou des situations d'occultations.

Pour finir, le *modèle d'interaction* s'écrit :

$$Int(C) = \sum_{\tau_1 \in C} \sum_{\tau_2 \in C \setminus \{\tau_1\}} \sum_{t=max(b_{\tau_1}, b_{\tau_2})}^{min(e_{\tau_1}, e_{\tau_2})} IOU(x_{\tau_1}(t), x_{\tau_2}(t))^2 \,. \tag{IV.6}$$

Ce terme permet de pénaliser les recouvrements entre deux cibles à l'aide d'un critère d'*Intersection-Over-Union* (IOU). Ce modèle est particulièrement utile pour éviter d'estimer plusieurs trajectoires pour un même objet dans le cas de détections multiples associées à une même cible.

L'effet de ces différents termes est illustré en figure IV.3, où des exemples de configurations favorisées et pénalisées par ces modèles sont précisés.

IV.2.3 Optimisation avec méthode de Monte-Carlo par chaînes de Markov

Le problème d'association de données sur une fenêtre glissante, c'est-à-dire avec plusieurs images, peut être résolu par des méthodes d'échantillonnage de Monte-Carlo par Chaînes de Markov (MCMC) basées notamment sur l'algorithme Metropolis-Hastings. Ce genre de technique est alors dénommée MCMCDA (Markov Chain Monte-Carlo Data Association) et a été employée par plusieurs méthodes de suivi multi-objets [13, 43, 77, 99, 110, 132]. Nous nous inspirons fortement des travaux initiaux proposés par l'article [99] avec néanmoins quelques adaptations. Nous détaillons brièvement ici le principe général des méthodes d'association de données de type MCMCDA et précisons les modifications faites par rapport à l'approche initiale de l'article [99], notamment en ce qui concerne les mouvements possibles (pour la distribution de proposition).

Principe général des méthodes d'échantillonnage par MCMC

Les méthodes d'échantillonnage visent à estimer une loi de probabilité π, définie sur un espace $\mathcal{C}$ supposé ici fini, à partir d'un ensemble de N éléments (ou états) $(x_i)_{i=0\ldots N-1}$. Ces N éléments $(x_i)_{i=0\ldots N-1}$ correspondent à des tirages de variables aléatoires $(X_i)_{i=0\ldots N-1}$ [1]. On cherche à déterminer ainsi un ensemble d'états

1. La notation X indique ici une variable aléatoire, qui correspond donc à une fonction mesurable de $\mathcal{C} \to \mathbb{R}$. La notation x correspond à un élément de l'ensemble des éventualités $\mathcal{C}$. Tout état $x \in \mathcal{C}$ est ainsi une valeur que peut prendre toute variable aléatoire X, avec une certaine probabilité notée $P(X = x)$.

$(x_i)_{i=0...N-1}$ dont la distribution converge vers celle de π lorsque $N \to \infty$. Les méthodes d'échantillonnage de Monte-Carlo par chaîne de Markov génèrent ces états (x_i) à partir d'une chaîne de Markov homogène $(X_i)_{i\geq 0}$. Cela signifie que l'état de la variable aléatoire X_{i+1} est déterminé uniquement à partir de l'état de la variable aléatoire précédente X_i en suivant la probabilité conditionnelle $P(X_{i+1} = x|X_i = x') = p_{x,x'}$ appelée probabilité de transition de l'état x vers l'état x' (probabilité indépendante de l'instant i). Les chaînes de Markov seront toutes ici supposées homogènes et définies sur un ensemble fini d'états. La chaîne de Markov définie sur l'espace $\mathcal{C}$ à partir des probabilités de transitions $p_{x,x'}$ et de la loi μ_0 de l'état initial X_0 est alors notée $\mathcal{M}$. L'idée générale des méthodes d'échantillonnage de type MCMC est de définir une chaîne de Markov $\mathcal{M}$, en choisissant judicieusement des probabilités de transitions $p_{x,x'}$, dont les variables aléatoires $(X_i)_{i\geq 0}$ vont converger vers la distribution π lorsque $i \to \infty$ et cela quelque soit la loi de probabilité initiale μ_0.

Pour qu'une chaîne de Markov converge vers une distribution de probabilités μ, il est nécessaire que μ soit une distribution stationnaire. Cela signifie que μ est invariante pour les probabilités de transition $p_{x,x'}$, et plus particulièrement que si l'état X_n suit la distribution μ, alors X_{n+1} suit aussi cette distribution μ. Une distribution est ainsi stationnaire si et seulement si :

$$\forall x \in \mathcal{C}, \mu(x) = \sum_{x' \in \mathcal{C}} \mu(x') p_{x',x}. \tag{IV.7}$$

Il existe alors plusieurs hypothèses qui permettent de garantir l'unicité de la distribution stationnaire μ pour une chaîne de Markov homogène $\mathcal{M}$, et d'assurer la convergence de $\mathcal{M}$ vers cette unique distribution stationnaire [21]. Dans le cas où l'espace $\mathcal{C}$ est fini, on a en particulier :

(i) $\mathcal{M}$ est dite irréductible si tout état x' est accessible à partir de n'importe quel état x, c'est-à-dire :

$$\forall x, x' \in \mathcal{C}, \exists n \geq 0 \; tel\, que \; P(X_n = x'|X_0 = x) > 0\,. \tag{IV.8}$$

(ii) Si $\mathcal{M}$ est irréductible alors $\mathcal{M}$ admet au plus une distribution stationnaire.

(iii) $\mathcal{M}$ est dite apériodique si et seulement si :

$$\forall x \in \mathcal{C}, PGCD\{n > 0 :\; P(X_n = x|X_0 = x) > 0\} = 1\,. \tag{IV.9}$$

(iv) Si $\mathcal{M}$ est irréductible, apériodique et admet une distribution stationnaire μ alors $\mathcal{M}$ converge vers μ.

Pour appliquer des méthodes MCMC pour estimer la loi de probabilité π, il va donc falloir définir une chaîne de Markov $\mathcal{M}$ telle que :

(i) π est une loi stationnaire pour $\mathcal{M}$.

(ii) $\mathcal{M}$ admet au plus une loi stationnaire.

(iii) $\mathcal{M}$ converge vers sa loi stationnaire.

Nous décrivons maintenant comment vérifier en partie ces critères avec l'algorithme de Metropolis-Hastings.

Algorithme de Metropolis-Hastings

Étant donné une loi de probabilité π définie sur $\mathcal{C}$, l'algorithme de Metropolis-Hastings [21] va permettre de définir une chaîne de Markov admettant π comme distribution stationnaire.

Plutôt que de définir directement les probabilités de transition $p_{x,x'}$, l'idée est ici de considérer des probabilités de propositions $q_{x,x'}$. À partir du dernier état x de la chaîne, on propose un nouvel état x' en suivant la loi de proposition $q(x,.)$. On considère alors une probabilité d'acceptation du nouvel état x' définie par :

$$A(x,x') = min(1, \frac{\pi(x')q_{x',x}}{\pi(x)q_{x,x'}}) \,, \tag{IV.10}$$

et le nouvel état x' est uniquement accepté avec cette probabilité $A(x,x')$ (en cas de non-acceptation l'ancien état x est conservé). $A(x,x')$ est ici uniquement défini lorsque $q_{x,x'} > 0$ (autrement x' ne peut pas être proposé), et on considère $A(x,x') = 0$ si $q_{x,x'} = 0$ dans les équations qui suivent. L'intérêt est que les probabilités de transition s'écrivent alors $p_{x,x'} = q_{x,x'}A(x,x')$ et vérifient :

$$\pi(x)p_{x,x'} = \pi(x)q_{x,x'}A(x,x') = min(\pi(x)q_{x,x'}, \pi(x')q_{x',x}) = \pi(x')p_{x',x}. \tag{IV.11}$$

Cette équation d'équilibre $\pi(x)p_{x,x'} = \pi(x')p_{x',x}$ permet de montrer que π est bien une distribution stationnaire qui vérifie l'équation (IV.7). En effet :

$$\forall x, x' \in \mathcal{C},\ \sum_{x' \in \mathcal{C}} \mu(x')p_{x',x} = \sum_{x' \in \mathcal{C}} \mu(x)p_{x,x'} = \mu(x) \sum_{x' \in \mathcal{C}} p_{x,x'} = \mu(x) \,. \tag{IV.12}$$

Pour pouvoir appliquer la méthode de Metropolis-Hastings, il faut de plus vérifier que $\mathcal{M}$ admet π comme unique distribution stationnaire et que $\mathcal{M}$ converge bien vers sa loi de distribution stationnaire. Comme vu précédemment, cela peut être le cas si les probabilités de proposition $q_{x,x'}$ sont choisies de manière à ce que $\mathcal{M}$ soit irréductible et apériodique. Bien qu'une chaîne de Markov qui vérifie ces critères converge bien vers la distribution π, les premiers éléments de la chaîne peuvent suivre une distribution très différentes en fonction de l'état initial x_0 (en particulier si x_0 n'est pas dans une zone de forte probabilité de la loi π). En pratique, les k premiers éléments de la chaîne sont généralement mis de côté et la loi de probabilité est estimée à partir des éléments $(x_i)_{i=k...N-1}$ (supprimer les premiers éléments de la chaîne constitue une étape dite de *burn-in*). Cette suppression des premiers éléments de la chaîne n'est cependant pas nécessaire si le premier élément x_0 de la chaîne est déjà un élément de forte probabilité pour la loi π.

Association de données par MCMC

Nous détaillons maintenant comment l'algorithme de Metropolis-Hastings peut être employé pour résoudre l'association de données multi-images, en nous inspirant de l'approche proposée dans [99]. L'idée est de définir une loi de probabilité π à partir de l'énergie globale E et de maximiser cette probabilité de manière similaire à une approche de type MAP (*Maximum A-Posteriori*). La configuration C^* maximisant la probabilité π est alors estimée à partir des configurations de la chaîne de Markov produite. La loi de probabilité π que nous considérons s'écrit :

$$\pi(C) = \frac{1}{Z} e^{-E(C)/\sigma^2} \,, \tag{IV.13}$$

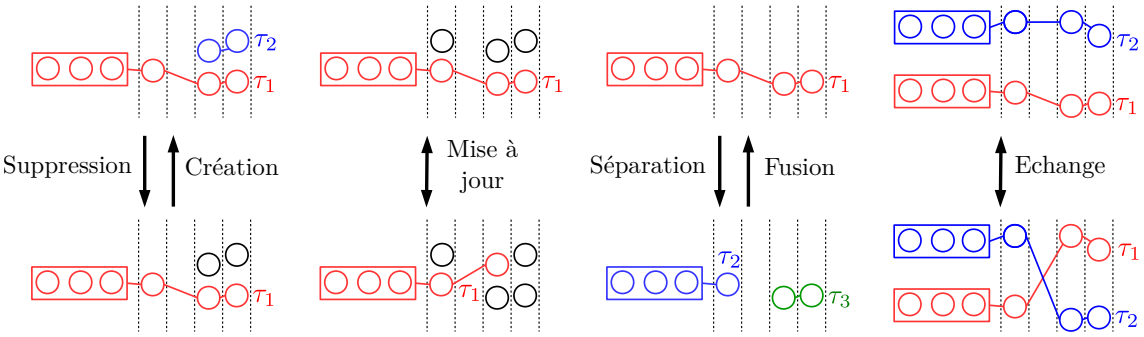


FIGURE IV.4 – Illustrations des différents types de mouvements considérés pour les propositions de configurations dans l'association de données par échantillonnage MCMC utilisée.

avec Z une constante de normalisation et où σ peut être choisi de manière à rendre la distribution plus ou moins piquée. En pratique, un paramètre σ approprié permettra de faire un compromis idéal entre l'exploitation de l'état courant, en rejetant majoritairement les configurations peu pertinentes, tout en garantissant une certaine exploration de l'espace des configurations possibles $\mathcal{C}$ en s'autorisant à accepter quelques configurations légèrement moins pertinentes dans la chaîne de Markov. Une telle stratégie d'optimisation permet ainsi d'éviter de rester bloquer sur un minimum local de E (respectivement un maximum local de π). Enfin, il est important de remarquer que le paramètre de normalisation Z, qui est complexe à évaluer en pratique, n'est pas utilisé dans l'algorithme de Metropolis-Hastings puisque seuls des ratios $\frac{\pi(C')}{\pi(C)}$ sont évalués pour les probabilités d'acceptation.

De la même manière que proposé dans l'article [99], une nouvelle configuration C' est proposée à partir de la configuration courante C, avec une probabilité de proposition $q_{C,C'}$, en envisageant plusieurs types de mouvements. Nous considérons six types de mouvements, à savoir des mouvements de création, de suppression, de mise à jour, de séparation, de fusion et d'échange. Par rapport à l'article [99], qui propose huit types de mouvements, nous utilisons moins de mouvements afin de simplifier les calculs des probabilités de proposition $q_{C,C'}$ et $q_{C',C}$. Nos mouvements sont néanmoins légèrement différents afin de permettre d'explorer plus efficacement l'espace des configurations $\mathcal{C}$. Les mouvements envisagés par notre approche, qui sont choisis aléatoirement pour chaque nouvelle proposition de configuration C', fonctionnent de la manière suivante :

(i) **Création** : Pour ce mouvement, les étapes suivantes sont effectuées :
- Un sens temporel de création $\epsilon \in \{-1, 1\}$, soit chronologique ($\epsilon = 1$) ou anti-chronologique ($\epsilon = -1$) est sélectionné aléatoirement, avec dans les deux cas une probabilité $\frac{1}{2}$.
- Une détection d^1 est alors choisie aléatoirement (uniformément) parmi les détections de la fenêtre glissante non associées à une piste τ de la configuration C pour lesquelles il existe au moins une seconde détection d' à laquelle d^1 peut être reliée en respectant les contraintes indiquées à la sous-section IV.2.1 (c'est-à-dire en étant suffisamment proche temporellement, spatialement et au niveau de leurs tailles respectives). Cette détection d' doit pouvoir se situer après ou avant la détection d^1 d'un point de vue temporel de manière cohérente avec le sens temporel de création, c'est-à-dire

$sign(d'_t - d^1_t) = \epsilon$.

— À partir de la détection d^1, les images $I_{d^1_t+\epsilon}, I_{d^1_t+2\epsilon}, ..., I_{d^1_t+\delta t_l \epsilon}$ sont examinées, où δt_l est l'écartement temporel maximal entre deux détections consécutives au sein d'une même piste, défini en sous-section IV.2.1. Parmi ces images, seules sont conservées celles contenant effectivement des détections pouvant être associées à d^1. Une de ces images I est sélectionnée aléatoirement parmi ces images, et une détection d^2 est ensuite choisie aléatoirement (uniformément) parmi les détections de I pouvant être associées à d^1. Ce processus est répété à partir de la détection d^2 et s'arrête avec une probabilité $prob_{arrêt}$ à chaque détection d^i sélectionnée ou lorsque aucune détection supplémentaire n'est disponible.

— Une nouvelle configuration C' est alors proposée, en ajoutant à la configuration C une piste τ constituée des détections $(d^i)_{i\geq 1}$ sélectionnées, soit $C' = C \cup \{\tau\}$. On peut remarquer que le procédé décrit ici oblige à considérer des pistes τ comportant au moins deux éléments afin de respecter les contraintes sur les configurations énoncées en sous-section IV.2.1.

(ii) **Suppression** : À partir de la configuration $C = \{\tau_1, ..., \tau_M\}$, une piste τ_i est choisie aléatoirement (uniformément) parmi $\{\tau_1, ..., \tau_M\}$ et la configuration C' proposée est obtenue à partir de C en supprimant cette piste τ_i, c'est-à-dire $C' = C \backslash \{\tau_i\}$.

(iii) **Mise à jour** : À partir de la configuration $C = \{\tau_1, ..., \tau_M\}$, une piste τ_i est choisie aléatoirement (uniformément) parmi $\{\tau_1, ..., \tau_M\}$ ainsi qu'un sens de mise à jour $\epsilon \in \{-1, 1\}$, soit chronologique ($\epsilon = 1$) ou anti-chronologique ($\epsilon = -1$). Une détection $d \in \tau_i$ est choisie aléatoirement (uniformément) parmi les détections de τ_i, et toutes les détections de τ_i avant ou après d en suivant le sens de mise à jour sont supprimées de τ_i. La trajectoire τ_i est ensuite prolongée au-delà de la détection d, en suivant le sens de mise à jour, via le même processus que décrit pour la création des pistes, ce qui donne une nouvelle piste τ'_i. On considère alors $C' = C \backslash \{\tau_i \cup \tau'_i\}$.

(iv) **Séparation** : À partir de la configuration $C = \{\tau_1, ..., \tau_M\}$, une piste τ_i est choisie aléatoirement (uniformément) parmi les pistes de $\{\tau_1, ..., \tau_M\}$ qui comportent au moins quatre éléments (les pistes de moins de quatre éléments ne pouvant être séparées en deux pistes admissibles contenant chacune au moins deux éléments). On examine alors toutes les possibilités de partition de la piste τ_i en deux pistes τ'_i et τ''_i, $\tau_i = \tau'_i \cup \tau''_i$ avec τ'_i et τ''_i formant deux pistes admissibles et consécutives dans le temps. Une telle partition $\tau_i = \tau'_i \cup \tau''_i$ est alors choisie aléatoirement (uniformément) et on considère $C' = C \backslash \{\tau_i\} \cup \{\tau'_i \cup \tau''_i\}$.

(v) **Fusion** : À partir de la configuration $C = \{\tau_1, ..., \tau_M\}$, deux pistes τ_i et τ_j consécutives temporellement, c'est-à-dire si le dernier élément de τ_i précède le premier de τ_j, peuvent être fusionnées si leur réunion $\tau_i \cup \tau_j$ forme une piste admissible (qui vérifie les contraintes énoncées en sous-section IV.2.1). On considère aléatoire (uniformément) un couple de pistes (τ_i, τ_j) parmi l'ensemble des couples de pistes qui peuvent être fusionnées. Une nouvelle piste $\tau = \tau_i \cup \tau_j$ est alors créée, et on considère la configuration $C' = C \backslash \{\tau_i, \tau_j\} \cup \{\tau\}$.

(vi) **Echange** : Étant donné une configuration $C = \{\tau_1, ..., \tau_M\}$, deux pistes τ_i et τ_j peuvent réaliser un échange si elles peuvent s'écrire sous la forme de partitions $\tau_i = \tau'_i \cup \tau''_i$ et $\tau_j = \tau'_j \cup \tau''_j$, avec τ'_i et τ''_i deux pistes admissibles

données : C_0, N_{mouv}
$C^* = C_0$;
pour $i = 1$ *à* N_{mouv} **faire**
  $C_i = C_{i-1}$;
  Choisir aléatoirement un mouvement (création, suppression, mise à jour, séparation, fusion ou échange);
  Proposer C' à partir de C_{i-1} en suivant le mouvement choisi, comme décrit en sous-section IV.2.3.;
  Calculer les probabilités de proposition $q_{C_{i-1},C'}$ et $q_{C',C_{i-1}}$;
  Calculer la probabilité d'acceptation $A(C_{i-1}, C') = min(1, \frac{\pi(C')q_{C',C_{i-1}}}{\pi(C_{i-1})q_{C_{i-1},C'}})$;
  $C_i = C'$ avec une probabilité $A(C_{i-1}, C')$;
  si $\pi(C_i)/\pi(C^*) > 1$ **alors**
    $C^* = C_i$;
  fin
fin
retourner C^*;

Algorithme IV.1 : Association de données multi-images par MCMCDA.

consécutives temporellement et de même pour τ'_j et τ''_j. On suppose de plus que les ensembles $\tau_1 = \tau'_i \cup \tau''_j$ et $\tau_2 = \tau'_j \cup \tau''_i$ forment deux pistes admissibles. On choisit alors aléatoirement (uniformément) un échange parmi l'ensemble des échanges possibles pour la configuration C, et la configuration C' proposée est $C' = C \backslash \{\tau_i, \tau_j\} \cup \{\tau_1, \tau_2\}$.

Ces mouvements sont illustrés en figure IV.4. Les principales différences concernant ces mouvements, par rapport à l'article [99], sont les suivantes. Les créations et mises à jour des pistes peuvent être faites à la fois dans le sens chronologique et dans le sens anti-chronologique, alors que seul le sens chronologique est envisagé dans [99]. Pour la création et la mise à jour à partir d'une détection d, on ne considère que les images contenant des détections pouvant être associées à d. Cela permet de créer et de prolonger plus facilement des pistes lorsque les détections disponibles sont toutes distantes de plusieurs images. Ces modifications permettent d'explorer plus aisément l'ensemble des configurations possibles $\mathcal{C}$.

Afin d'appliquer l'algorithme de Metropolis-Hastings avec ces mouvements, il reste à vérifier que la chaîne de Markov engendrée $\mathcal{M}$ converge effectivement vers la distribution stationnaire π. La preuve donnée par l'article [99] reste valide, malgré nos modifications sur les mouvements, et nécessite de vérifier que $\mathcal{M}$ est irréductible et apériodique. $\mathcal{M}$ est effectivement irréductible car toute configuration C est accessible à partir de toute autre configuration C' en considérant un enchaînement de mouvements de création et de suppression. $\mathcal{M}$ est de plus apériodique car toute configuration C a une probabilité non nulle d'être elle-même prise pour l'état suivant de la chaîne grâce au mouvement de mise à jour. L'algorithme de Metropolis-Hastings peut donc effectivement être employé, et nous détaillons l'algorithme utilisé pour résoudre l'association de données dans l'algorithme IV.1. En pratique, la configuration initiale C_0 est choisie à partir de la solution C^* déterminée pour la fenêtre glissante précédente, afin de permettre de démarrer l'échantillonnage dans un endroit approprié de l'espace des configurations $\mathcal{C}$.

IV.3 Représentations structurées en norme $l_{\infty,1}$

Dans cette section nous détaillons comment le terme d'apparence $App(C)$ peut être formulé afin d'exploiter des représentations parcimonieuses des détections de la fenêtre glissante. Nous proposons de plus des représentations parcimonieuses structurées, définies avec une norme $l_{\infty,1}$ pondérée, qui sont plus pertinentes vis-à-vis de notre contexte spécifique d'association de données multi-images. Enfin, nous montrons comment les méthodes d'optimisation de gradient proximal avec ensembles actifs, déjà employées au chapitre précédent dans le cas de représentations parcimonieuses en norme l_1, peuvent être adaptées pour calculer efficacement les représentations en norme $l_{\infty,1}$ pondérée proposées.

Cette section présente les spécificités de notre approche eu égard aux méthodes de suivi à fenêtre glissante déjà existantes. En effet, et dans la limite de nos connaissances actuelles, l'emploi de représentations parcimonieuses structurées n'a pas été auparavant proposé pour traiter le problème de suivi multi-objets hors ligne.

IV.3.1 Modèle d'apparence à base de représentations parcimonieuses

Nous définissons ici le modèle d'apparence $App(C)$ employé au sein de l'énergie E décrite par l'équation (IV.3). Ce terme exploite des représentations parcimonieuses des détections en s'inspirant des modèles d'apparence à base de représentations parcimonieuses proposés pour le suivi mono-objet [135] ainsi que des méthodes de classification multi-classes de type SRC [124].

Comme expliqué au sein des chapitres précédents, en particulier en section II.3, lorsqu'un dictionnaire collaboratif D inclut des éléments de k classes distinctes $L_1, ..., L_k$, il est possible de classifier un nouvel élément y parmi ces classes en utilisant sa représentation parcimonieuse α_y vis-à-vis du dictionnaire D. Le dictionnaire peut en effet s'écrire $D = [D_{L_1} \; ... \; D_{L_k}]$, où D_{L_i} est un dictionnaire spécifique propre à la classe L_i, et la classe L^* attribuée au nouvel élément y peut être estimée par :

$$L^* = \arg\min_L ||y - D_L \alpha_y^L||_2 \,, \tag{IV.14}$$

où α_y^L est la restriction de la représentation parcimonieuse α_y aux seules dimensions liées à des éléments de la classe L. La classe L^* est ainsi déterminée en examinant les erreurs de reconstruction résiduelle propres à chaque classe.

Au chapitre III précédent, ce principe était employé pour définir les valeurs d'affinité entre les trajectoires et les détections. Une telle stratégie pourrait être employée ici en considérant un dictionnaire composé d'éléments récents des trajectoires de $\mathcal{T}$, mais cette approche ne serait pas la plus adaptée dans le cas d'une fenêtre glissante. En effet, si une telle technique permet d'évaluer l'affinité entre les détections de la fenêtre glissante et les trajectoires de $\mathcal{T}$, elle ne peut néanmoins pas évaluer la cohérence en apparence entre deux détections de la fenêtre glissante. Dans le cas de pistes τ n'incluant pas de trajectoires de $\mathcal{T}$, c'est-à-dire des pistes qui correspondent à une hypothèse de création de trajectoire au sein de la fenêtre glissante, comment évaluer la cohérence en apparence de ces pistes ?

Pour ces raisons, le dictionnaire D décrit en sous-section IV.2.1 ne va pas se limiter aux seules détections des trajectoires de $\mathcal{T}$. Pour rappel, ce dictionnaire inclut

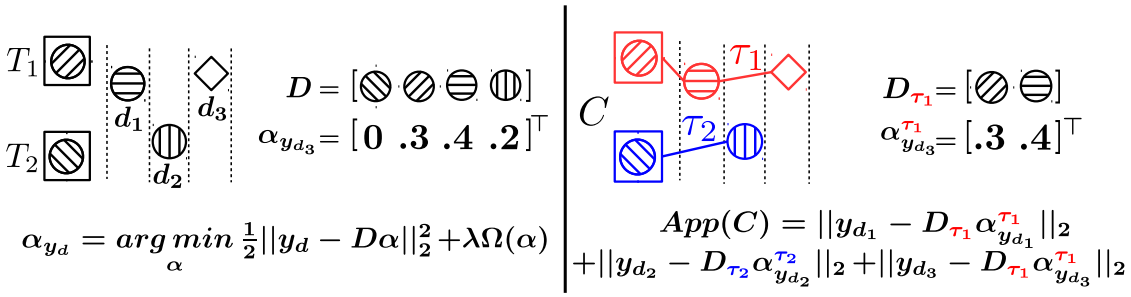


FIGURE IV.5 – Modèle d'apparence proposé à partir de représentations parcimonieuses. À gauche : fenêtre glissante actuelle et représentations parcimonieuses calculées pour les détections de la nouvelle image. À droite : configuration C considérée et valeur associée du terme $App(C)$.

l'ensemble des caractéristiques des détections de la fenêtre glissante ainsi que celles des N_{tr} dernières détections associées à toute trajectoire de $\mathcal{T}$. Une représentation parcimonieuse pour une détection d est alors définie par :

$$\alpha_{y_d} = \arg\min_{\alpha} \frac{1}{2}||y_d - D\alpha||_2^2 + \lambda\Omega(\alpha), \tag{IV.15}$$

où Ω est une norme favorisant la parcimonie des solutions α, c'est-à-dire les solutions α avec un faible nombre de coefficients non nuls.

L'intérêt de considérer, en plus des détections des trajectoires, l'ensemble des détections de la fenêtre glissante au sein du dictionnaire D est de permettre d'évaluer la pertinence en apparence d'une détection d par rapport à une piste τ. Pour toute piste τ, on considère le dictionnaire spécifique D_τ qui inclut les caractéristiques des détections de la piste τ. Dans le cas où une trajectoire T est de plus incluse dans cette piste τ, le dictionnaire D_τ inclut aussi les caractéristiques des N_{tr} dernières détections associées à T. On peut alors considérer l'erreur de reconstruction résiduelle de d par rapport à la piste τ pour évaluer la pertinence de l'appartenance de la détection d à la piste τ. Cela mène à considérer le terme $App(C)$ suivant :

$$App(C) = \sum_{\tau \in C} \sum_{d \in \tau} ||y_d - D_\tau \alpha^\tau_{y_d}||_2\,, \tag{IV.16}$$

où $||y_d - D_\tau\alpha^\tau_{y_d}||_2$ est l'erreur de reconstruction résiduelle de d par rapport à la piste τ. Ce modèle favorise ainsi les configurations C au sein desquelles toute détection d présente une faible erreur de reconstruction résiduelle par rapport à sa piste τ, comme illustré à la figure IV.5. Un tel terme tend ainsi à regrouper les détections qui se représentent mutuellement au sein d'une même piste, formant des pistes partageant une même apparence.

En pratique, évaluer le terme $App(C)$ s'avère coûteux dans le cas d'une optimisation par échantillonnage de type MCMC qui nécessite d'évaluer un très grand nombre de configurations C et donc de calculer un nombre important d'erreurs de reconstruction résiduelle. Au lieu d'utiliser les erreurs de reconstruction résiduelle, certaines approches en classification et en suivi mono-objet, comme par exemple [55], utilisent directement comme approximation les coefficients des représentations parcimonieuses :

$$L^* = \arg\max_{L} \sum_i |\alpha^L_{y_d}(i)|\,, \tag{IV.17}$$

où la somme impliquée dans cette équation prend en compte tous les coefficients $\alpha^L_{y_d}(i)$ du vecteur $\alpha^L_{y_d}$. Afin d'accélérer l'association de données de type MCMCDA que nous employons, nous utilisons cette approche et considérons au final comme modèle d'apparence :

$$App(C) = \sum_{\tau \in C} \sum_{d \in \tau} [1 - \sum_i |\alpha^\tau_{y_d}(i)|] \,. \tag{IV.18}$$

On peut noter que la valeur $1 - \sum_i |\alpha^\tau_{y_d}(i)|$ peut être vue comme une heuristique sur l'erreur de reconstruction résiduelle $||y_d - D_\tau \alpha^\tau_{y_d}||_2$. Plus précisément, la valeur $1 - \sum_i |\alpha^\tau_{y_d}(i)|$ est en réalité une borne inférieure de cette erreur de reconstruction résiduelle. En effet, on a :

$$||y_d - D_\tau \alpha^\tau_{y_d}||_2 \geq ||y_d||_2 - ||D_\tau \alpha^\tau_{y_d}||_2, \tag{IV.19}$$

par inégalité triangulaire inversée. De plus,

$$||D_\tau \alpha^\tau_{y_d}||_2 = ||\sum_i \alpha^\tau_{y_d}(i)\ y_{d^i_\tau}||_2 \leq \sum_i ||\alpha^\tau_{y_d}(i)\ y_{d^i_\tau}||_2 = \sum_i |\alpha^\tau_{y_d}(i)|\ ||y_{d^i_\tau}||_2, \tag{IV.20}$$

avec $(y_{d^i_\tau})$ les caractéristiques des détections considérées dans D_τ. On aboutit ainsi à :

$$||y_d - D_\tau \alpha^\tau_{y_d}||_2 \geq ||y_d||_2 - ||D_\tau \alpha^\tau_{y_d}||_2 \geq ||y_d||_2 - \sum_i |\alpha^\tau_{y_d}(i)|\ ||y_{d^i_\tau}||_2. \tag{IV.21}$$

Or, toutes les caractéristiques y_d et $y_{d^i_\tau}$ sont supposées normalisées, ce qui amène finalement à :

$$||y_d - D_\tau \alpha^\tau_{y_d}||_2 \geq 1 - \sum_i |\alpha^\tau_{y_d}(i)|. \tag{IV.22}$$

IV.3.2 Pénalisation en norme $l_{\infty,1}$ pondérée proposée

Nous venons de détailler comment des représentations parcimonieuses des détections pouvaient être exploitées au sein du terme $App(C)$. Néanmoins, la norme Ω employée dans l'équation (IV.15) n'a pas été explicitée. En pratique, plusieurs normes Ω peuvent être envisagées pour induire une parcimonie au niveau des représentations. Nous examinons ici différentes possibilités de normes Ω et proposons une norme spécifiquement adaptée à notre problème de suivi à fenêtre glissante.

Représentations parcimonieuses structurées

Suivant la norme Ω employée, les représentations obtenues peuvent présenter un support qui s'approche d'une structure particulière. Par exemple, une représentation parcimonieuse induite par une norme l_1, comme effectué au cours du chapitre III, favorise une simple parcimonie des représentations qui tend seulement à limiter le nombre de coefficients non nuls des solutions. La principale limitation d'une pénalisation en norme l_1, pour le calcul des représentations parcimonieuses, est que cette norme fait jouer un rôle complètement symétrique aux variables des représentations ou de façon similaire aux éléments du dictionnaire D. Dans certains contextes d'applications on peut en effet disposer d'un a-priori plus précis sur la structure du support idéal des représentations.

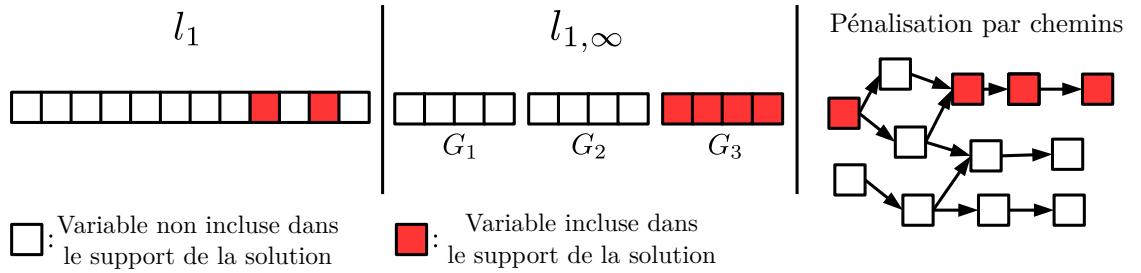


FIGURE IV.6 – Supports favorisés par différentes normes. À gauche, la norme l_1 favorise un support restreint à quelques variables. Au centre, une norme de groupe de type $l_{1,\infty}$ favorise un support incluant peu de groupes G_i mais faisant participer tous les éléments de ces groupes. À gauche, la pénalisation par chemins [83] favorise un support qui peut être recouvert par un faible nombre de chemins dans un graphe dirigé acyclique.

Des normes plus élaborées ont de ce fait été proposées de façon à promouvoir des structures plus spécifiques pour le support des représentations. En particulier, des normes de groupes comme les normes $l_{1,2}$ ou $l_{1,\infty}$ (ou de façon plus générale les normes $l_{1,q}$ avec $q > 1$ [2]) vont rassembler les éléments du dictionnaire D au sein d'une partition, constituée de groupes disjoints. Ces normes vont alors favoriser les représentations où seul un faible nombre de groupes sont actifs (i.e. avec au moins un élément du groupe associé à un coefficient non nul) tout en favorisant la participation de tous les éléments de chaque groupe actif. Ainsi, les représentations obtenues présentent une parcimonie au niveau des groupes (peu de groupes étant actifs) et une participation uniforme des éléments au sein de chaque groupe. Les représentations parcimonieuses de groupe ont été notamment employées dans certaines approches de suivi mono-objet afin d'exploiter des descriptions de cibles avec des caractéristiques visuelles multiples ou de façon à reconstruire conjointement plusieurs hypothèses pour la cible [50, 137, 138].

Des structures encore plus particulières peuvent être favorisées par certaines normes. On peut par exemple citer l'article [83] où les éléments d'un dictionnaire sont organisés en tant que noeuds d'un graphe dirigé acyclique (DAG) et où la norme proposée tend à favoriser les solutions qui font intervenir des éléments d'un faible nombre de chemins du graphe considéré.

Ces types de représentations sont illustrées en figure IV.6. On peut noter que nous nous limitons principalement aux représentations parcimonieuses pénalisées par une norme Ω, alors qu'il est possible de définir des représentations induites par des pénalisations Ω qui ne sont pas nécessairement des normes. Les avantages liés à l'usage d'une norme sont principalement les suivants :

(i) La pénalisation Ω étant convexe, le problème (IV.15) est alors lui aussi convexe et cela permet d'envisager des méthodes d'optimisation exactes, notamment par les méthodes de gradient proximal explicitées à la sous-section III.3.2.

(ii) Des résultats théoriques, en particulier des critères d'optimalité généraux et spécifiques aux stratégies d'optimisation avec ensembles actifs, existent pour le cas de normes Ω quelconques [5].

2. On trouve dans la littérature les deux notations $l_{1,q}$ et $l_{q,1}$ pour décrire la même norme. Pour éviter toute confusion, nous gardons la même convention dans ce manuscrit et utilisons la convention de [103] en notant $l_{1,q}(u) = \sum_i ||u|_{G_i}||_q$.

(iii) Des formulations de l'opérateur proximal de Ω, en fonction de projections sur une norme dite duale, permettent généralement de déterminer efficacement ces opérateurs proximaux [5, 101].

Ainsi, se limiter à une pénalisation sous la forme d'une norme Ω permet d'adapter facilement les considérations d'optimisation développées au chapitre précédent.

Structure désirée pour les représentations parcimonieuses

Du fait des différentes possibilités de représentations parcimonieuses structurées envisageables, une question naturelle est de déterminer la structure de parcimonie qui serait la plus adaptée dans notre contexte d'association de données multi-images. Il restera alors à trouver une norme Ω qui permet d'induire une telle structure, ou, tout du moins, une structure proche.

Idéalement, toute détection d devrait être représentée uniquement par des détections associées à la même cible ainsi qu'éventuellement par les détections de la trajectoire T de cette cible. De cette manière, la piste τ dans la fenêtre glissante qui inclurait la trajectoire T et les détections de la cible liée à la détection d permettrait de représenter au mieux cette détection. La structure de parcimonie idéale prendrait donc la forme d'une unique piste au sein de la fenêtre glissante.

La norme proposée dans l'article [83], favorisant une structure comportant peu de chemins dans un graphe dirigé acyclique, parait dans un premier temps tout à fait appropriée. Il est en effet possible d'organiser les détections de la fenêtre glissante au sein d'un graphe de manière à ce qu'une piste τ, qui respecte les contraintes (i)-(vi) définies en sous-section IV.2.1, corresponde exactement à un chemin de ce graphe. Cependant, cette norme s'avère très coûteuse en temps de calcul. Les opérateurs proximaux associés doivent être évalués à partir de solutions de problèmes de flots avec des coûts quadratiques des arcs, ce qui nécessite l'emploi d'algorithmes de complexité trop importante pour notre approche de suivi. Il faut donc chercher une alternative permettant de se rapprocher au mieux de la structure de parcimonie envisagée tout en gardant une complexité raisonnable pour l'optimisation des représentations.

De toute évidence une simple norme l_1 n'est pas non plus idéale. Une telle norme aura tendance à représenter chaque détection par un très faible nombre de détections proches. Le problème majeur ici est que chaque détection aura tendance à être représentée exclusivement par la détection la plus proche en apparence dans la fenêtre glissante. Les représentations de chaque détection d ne feront pas intervenir toutes les détections liées à la même cible mais seulement un nombre très réduit de ces détections (les plus semblables en apparence à la détection d). On peut alors s'attendre à ce que ces représentations ne constituent pas assez de liens entre les détections d'une même cible pour les forcer à être rassemblées dans une unique piste, produisant plutôt un nombre important de pistes courtes très cohérentes en apparence.

Les normes de groupes discutées précédemment, de type $l_{1,q}$ avec $q > 1$, permettent de promouvoir des structures plus complexes. Ces normes nécessitent cependant de considérer un ensemble de groupes disjoints qui constituent une partition des éléments du dictionnaire D. Dans notre cas de figure, D est défini (voir le détail en sous-section IV.2.1) à partir de détections des trajectoires de $\mathcal{T}$ et de celles de la

fenêtre glissante :

$$D = [D_{\mathcal{T}}\ D_{I_{t-\Delta t+1}}\ ...\ D_{I_{t-1}}], \tag{IV.23}$$

où $D_{\mathcal{T}}$ est défini à partir des N_{tr} dernières détections de chaque trajectoire, et D_{I_i} à partir des détections de l'image I_i. Il est alors possible de définir naturellement Δt groupes $G_1, ..., G_{\Delta t}$, chaque groupe G_i pour $i = 1...\Delta t - 1$ étant associé aux détections de l'image I_{t-i} (liées à D_{t-i}) et le groupe $G_{\Delta t}$ étant associé aux détections des trajectoires (liées à $D_{\mathcal{T}}$). Cependant, les normes de groupes de type $l_{1,q}$ avec $q > 1$ (par exemple $l_{1,2}$ ou $l_{1,\infty}$) définies avec les groupes $(G_i)_{i=1...\Delta t}$ ne produisent pas une structure de parcimonie adaptée comme illustré en figure IV.7. Ces normes favorisant une participation de peu de groupes avec une participation de tous les éléments des groupes actifs, cela se traduit notamment par faire participer toutes les détections de seulement quelques images de la fenêtre glissante.

Bien que les normes de groupes précédemment évoquées favorisent des structures de parcimonie peu adaptées à notre problème de suivi, il est cependant intéressant de regarder comment les groupes $(G_i)_{i=1...\Delta t}$ peuvent être utilisés pour définir une norme induisant une structure plus appropriée. Pour ce faire, il est judicieux d'observer comment se comporte une piste τ par rapport à ces groupes $(G_i)_{i=1...\Delta t}$. Une piste τ correcte va typiquement :

(i) inclure au plus une détection de chaque groupe G_i associé à l'image I_{t-i},

(ii) inclure au plus une trajectoire de $\mathcal{T}$, trajectoire associée à un faible nombre de détections du groupe $G_{\Delta t}$,

(iii) inclure une détection de la plupart des groupes $(G_i)_{i=1...\Delta t}$ (la cible associée à τ étant supposée détectée sur la plupart des images de la fenêtre glissante).

Une structure appropriée vis-à-vis de ces observations aura alors tendance à faire participer le plus de groupes tout en cherchant à utiliser le moins d'éléments par groupes. Cette structure aurait de plus tendance à promouvoir une participation uniforme des groupes tout en favorisant une parcimonie au sein de chaque groupe. Ce genre de structure est à l'opposée de celle induite par les normes de groupes $l_{1,q}$ proposées ici. Est-il alors possible de formuler une norme qui induit une telle structure ? Pour trouver une telle norme, il est judicieux d'étudier l'emploi des normes induisant une structure de parcimonie dans le cadre plus général de la sélection de variables.

Régularisations pour la sélection de variables

Les pénalisations vu précédemment n'ont pas été seulement employées dans le cas de représentations structurées mais ont aussi été utilisées en particulier pour le problème plus général de sélection de variables ou sélection de caractéristiques (*feature selection*). La norme l_1 a été particulièrement étudiée (amenant à une régularisation de type LASSO) ainsi que les normes de groupes $l_{1,q}$ avec $q > 1$ (amenant à une régularisation de type Group-LASSO). Ces normes de groupes ont été notamment proposées pour aborder la sélection de caractéristiques multi-tâches (*multi-task feature selection*) [139]. Ce problème survient lorsque l'on cherche à apprendre plusieurs modèles pour différentes tâches à partir d'un même ensemble de caractéristiques. Les normes de groupes $l_{1,q}$ permettent alors de sélectionner un faible nombre de caractéristiques qui seront utilisées pour l'ensemble des modèles. Chaque groupe fait intervenir l'ensemble des variables associées à une même caractéristique pour les différents modèles. Les normes de groupes favoriseront l'emploi d'un faible nombre de

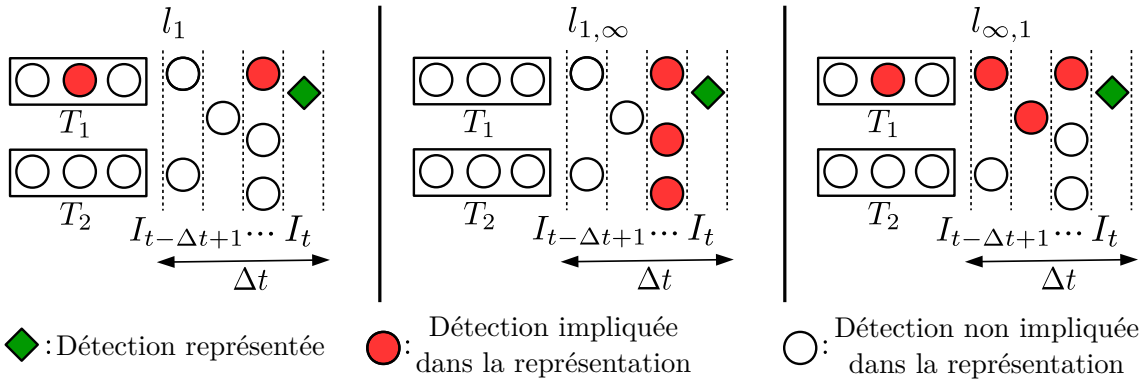


FIGURE IV.7 – Structures de parcimonie induites par différentes pénalisations au sein de la fenêtre glissante.

groupes, c'est-à-dire d'un faible nombre de caractéristiques, tout en favorisant l'utilisation de toutes les variables des groupes actifs, c'est-à-dire à promouvoir l'emploi par tous les modèles des caractéristiques sélectionnées.

Néanmoins, il existe certaines situations de sélection de caractéristiques multi-tâches où l'objectif n'est pas de trouver un faible nombre de caractéristiques employées pour toutes les tâches mais au contraire de trouver des caractéristiques employées chacune pour un faible nombre de tâches. Cela s'avère judicieux lorsque l'on cherche à différencier les différentes tâches les unes des autres en cherchant des caractéristiques spécifiques à chaque tâche. Pour résoudre un tel problème, une régularisation de type Exclusive Lasso a été proposée dans [141]. Cette régularisation fait intervenir une norme de groupe, élevée au carré, de type $l_{2,1}$:

$$||u||_{2,1} = \sqrt{\sum_i ||u_{|_{G_i}}||_1{}^2}. \tag{IV.24}$$

Cette norme a pour effet de mettre en concurrence les variables de chaque groupe du fait de la norme l_1 qui leur est appliquée, et de promouvoir l'utilisation de l'ensemble des groupes du fait de la norme l_2. Cela se traduit donc effectivement, lorsque chaque groupe correspond aux variables des différentes tâches liées à une même caractéristique, à favoriser l'exploitation de chaque caractéristique par un faible nombre de tâches.

On peut remarquer que la régularisation de type Exclusive Lasso se rapproche de la structure de groupe que nous cherchons à promouvoir. En effet, cette régularisation favorise la parcimonie au sein des groupes tout en favorisant l'activation de l'ensemble des groupes. Est-il de ce fait possible de s'inspirer de cette pénalisation pour définir une norme adaptée à notre problème de suivi ?

Représentations en norme $l_{\infty,1}$ pondérée

En nous inspirant de la pénalisation de type Exclusive Lasso, nous proposons d'employer une norme de groupe $l_{\infty,1}$ pondérée définie par :

$$||\alpha||^w_{\infty,1} = \max_{i=1..\Delta t} w_i ||\alpha_{|_{G_i}}||_1 \,, \tag{IV.25}$$

où $\alpha_{|_{G_i}}$ est la restriction de α aux variables du groupe G_i et où les valeurs $(w_i)_{i=1...\Delta t}$ sont des poids strictement positifs. Les groupes $(G_i)_{i=1...\Delta t}$ sont ceux précédemment

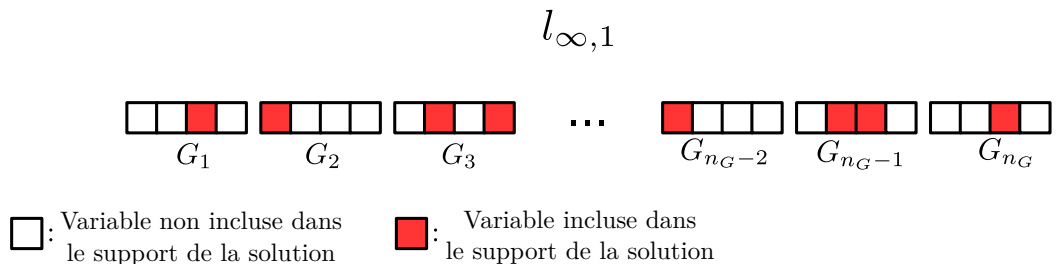


FIGURE IV.8 – Structure favorisée par une norme $l_{\infty,1}$. Cette norme favorise tous les groupes G_i à participer au support de la solution, mais en n'incluant qu'un faible nombre de variables de chaque groupe.

définis, à savoir, pour $i = 1...\Delta t - 1$, G_i est constitué des indices des colonnes du dictionnaire D correspondant aux détections de l'image I_{t-i} tandis que $G_{\Delta t}$ correspond aux indices liés aux détections des trajectoires de $\mathcal{T}$. Cette norme favorise effectivement la structure recherchée puisque la norme l_1 appliquée au sein de chaque groupe favorise la sélection de peu d'éléments tandis que la norme l_∞ appliquée entre les groupes favorise une participation de l'ensemble des groupes.

D'un point de vue plus concret, cette norme a les effets suivants :

(i) Toute détection est représentée par un faible nombre de détections par image et au sein des trajectoires $\mathcal{T}$, ce qui traduit une mise en compétition des détections de chaque image et une mise en compétition des trajectoires.

(ii) Toutes les images de la fenêtre glissante participent au sein de chaque représentation, ainsi que certaines détections des trajectoires de $\mathcal{T}$.

Cela signifie que les représentations parcimonieuses α_{y_d} ont effectivement tendance à impliquer les seules détections correspondant à la même cible que la détection d, comme illustré en figure IV.7 et en figure IV.8. Ainsi, le terme d'apparence $App(C)$ de l'énergie proposée en sous-section IV.3.1 cherchera effectivement à rassembler toutes les détections d'une même cible au sein d'une même piste, de manière à réduire au mieux l'ensemble des erreurs de reconstruction résiduelle des détections.

Les poids $(w_i)_{i=1...\Delta t}$ permettent ici de favoriser la participation de certains groupes, un poids w_i faible favorisant une participation plus importante des éléments du groupe G_i. Nous utilisons en pratique $w_{\Delta t} = \frac{1}{\Delta t - 1}$ et $w_i = 1$ pour $i < \Delta t$ afin de permettre aux détections liées aux trajectoires, c'est-à-dire du groupe $G_{\Delta t}$, de participer au sein des représentations avec une importance égale à celle de l'ensemble des détections de la fenêtre glissante.

L'emploi d'une norme de groupes de type $l_{\infty,1}$ à la place de la norme $l_{2,1}$ de la régularisation de type Exclusive Lasso est motivée par les points suivants. Tout d'abord, la norme l_2 dans la régularisation Exclusive Lasso tend à favoriser la participation de la majorité des groupes mais de manière assez souple. Cela est pertinent dans le contexte multi-tâches étudié par l'article [141] puisque l'on ne cherche pas nécessairement à faire participer l'ensemble des caractéristiques, le point important étant de limiter le nombre de tâches qui utilisent chaque caractéristique individuelle. Dans notre contexte de suivi à fenêtre glissante les cibles seront détectées en chaque image dans la majorité des cas, et on dispose donc d'un a-priori plus fort qui indique qu'une détection est supposée représentée par tous les groupes G_i. L'emploi d'une norme l_∞ favorise alors de manière bien plus forte une participation uniforme des groupes $(G_i)_{i=1...\Delta t}$, ce qui est nécessaire pour garantir que les représentations

feront intervenir des éléments de toutes les images de la fenêtre glissante. Une seconde raison est que les opérateurs proximaux des normes $l_{\infty,1}$ peuvent être évalués efficacement, comme précisé dans la prochaine sous-section. Les opérateurs proximaux des normes $l_{2,1}$ sont plus complexes à déterminer et les méthodes nécessitent de recourir à des algorithmes itératifs [22, 63]. L'emploi d'une norme $l_{\infty,1}$ est ainsi préférable pour l'optimisation par méthodes de gradient proximal.

IV.3.3 Optimisation par méthode de gradient proximal

Le calcul des représentations parcimonieuses en norme $l_{\infty,1}$ pondérée nécessite de résoudre le problème suivant :

$$\min_{\alpha} \frac{1}{2}||y - D\alpha||_2^2 + \lambda||\alpha||_{\infty,1}^w. \tag{IV.26}$$

Ce problème peut être vu comme un cas spécifique du problème :

$$\min_{\alpha} \frac{1}{2}||y - D\alpha||_2^2 + \lambda\Omega(\alpha), \tag{IV.27}$$

où Ω est une norme et λ un paramètre strictement positif. Au chapitre III, ce problème a été particulièrement approfondi dans le cas où Ω était une norme l_1. Ici, nous détaillons comment les méthodes d'optimisation proposées dans le cas de la norme l_1 (méthodes de gradient proximal, stratégie avec ensembles actifs...) s'étendent dans le cas d'une norme Ω quelconque. Enfin, nous précisons comment ces résultats généraux peuvent être utilisés pour calculer efficacement les représentations en norme $l_{\infty,1}$ pondérée définies par le problème (IV.26).

Normes, normes duales et méthodes proximales

Tout d'abord, le problème (IV.27) vérifie bien les conditions d'applications des méthodes proximales et peut s'écrire sous la forme du problème (III.13). En effet, la fonction objectif considérée ici peut s'écrire sous la forme de deux fonctions f et g, avec :

$$f(\alpha) = \frac{1}{2}||y - D\alpha||_2^2, \tag{IV.28}$$

$$g(\alpha) = \lambda\Omega(\alpha). \tag{IV.29}$$

Dans toute la suite de ce chapitre, les fonctions f et g feront références à ces formulations particulières. f et g sont toutes les deux convexes propres [3] fermées [4] avec f différentiable. g est en effet propre (Ω étant une norme, g est définie sur $\mathbb{R}^n$ sans prendre de valeurs infinies), convexe (toute norme étant convexe et λ étant strictement positif) et fermée (car g est continue par continuité de la norme). Il est donc possible d'employer directement les méthodes de gradient proximal, accélérées ou non, décrites par l'algorithme III.1 et l'algorithme III.2 en sous-section III.3.2.

Néanmoins, ces méthodes de gradient proximal (ISTA ou FISTA) nécessitent d'être capable d'évaluer les opérateurs proximaux de $prox_{\gamma g}$, avec $\gamma > 0$, c'est-à-dire

3. Une fonction $f : \mathbb{R}^n \to \bar{\mathbb{R}}$ est propre si $\forall x \in \mathbb{R}^n,\ f(x) > -\infty$ et $\exists x \in \mathbb{R}^n : f(x) < \infty$.

4. Une fonction $f : \mathbb{R}^n \to \bar{\mathbb{R}}$ est fermée si $\forall \alpha \in \mathbb{R}$, $\{x \in \mathbb{R}^n,\ f(x) \leq \alpha\}$ est un ensemble fermé.

les opérateur proximaux $prox_{\gamma\Omega}$ pour tout $\gamma > 0$. Lorsque Ω est une norme, il est possible de montrer [5, 101] que les opérateurs proximaux vérifient :

$$prox_{\gamma\Omega}(u) = u - \gamma\Pi_{\Omega^* \leq 1}(u/\gamma), \tag{IV.30}$$

où Ω^* est une norme spécifique associée à Ω, appelée norme duale, et où $\Pi_{\Omega^* \leq 1}$ correspond à la projection, pour la distance Euclidienne, sur la boule unité de cette norme. La norme duale Ω^* est définie par :

$$\Omega^*(u) = \max_{v/\Omega(v) \leq 1} u^\mathsf{T} v, \tag{IV.31}$$

et pour toute norme Ω l'application Ω^* est effectivement elle aussi une norme. La propriété décrite par l'équation (IV.30) montre qu'il est possible de déduire l'opérateur proximal d'une norme Ω à partir de la projection sur la boule unité de sa norme duale $\Pi_{\Omega^* \leq 1}$ (et réciproquement).

Dans certains cas, il est plus facile de déterminer la projection $\Pi_{\Omega^* \leq 1}$ que de déterminer l'opérateur proximal de Ω directement à partir de sa formulation initiale donnée par l'équation (III.16). C'est par exemple le cas de la norme l_1, dont la norme duale est la norme l_∞ et dont la projection sur la boule unité est simple à déterminer. Ainsi, la propriété mise en avant dans l'équation (IV.30) peut aider à déterminer les opérateurs proximaux pour des normes Ω, sous réserve de savoir déterminer leur norme duale Ω^* et la projection sur la boule unité associée.

De plus, la méthode de gradient proximal accélérée (FISTA ou APG) peut toujours gagner significativement en temps de calcul en pré-calculant la matrice de Gram $D^\mathsf{T}D$ du dictionnaire D, comme détaillé en sous-section III.3.2 dans le cas de la norme l_1. En effet, si le dictionnaire est composé de n éléments de dimension m et si $n \ll m$, alors ce pré-calcul de la matrice de Gram en $O(n^2)$ est toujours bénéfique puisque les évaluations de $f(\alpha)$ et $\nabla f(\alpha)$ à chaque itération sont ensuite grandement accélérées (passant d'un coût en $O(nm)$ à un coût en $O(n^2)$). Ce gain n'est néanmoins significatif en pratique que si les évaluations des opérateurs proximaux $prox_{\gamma\Omega}$ ne sont pas les plus limitantes en temps de calcul par rapport aux évaluations de $f(\alpha)$ et $\nabla f(\alpha)$.

Ensembles actifs

La norme duale Ω^* intervient aussi dans des conditions d'optimalité pour le problème (IV.27), conditions qui vont permettre d'adapter les stratégies à base d'ensembles actifs vu en sous-section III.3.2 dans le cas de la norme l_1. L'objectif est de résoudre le problème (IV.27) en considérant uniquement un sous-ensemble d'éléments du dictionnaire D, et de prendre progressivement davantage d'éléments en compte jusqu'à atteindre une solution globale.

Étant donné un sous-ensemble d'indices $\mathcal{A}$, on définit le sous-problème associé au problème (IV.27) restreint à $\mathcal{A}$:

$$\min_\alpha \ \frac{1}{2}||y - D_\mathcal{A}\alpha||_2^2 + \lambda\Omega(\widetilde{\alpha}). \tag{IV.32}$$

Le dictionnaire $D_\mathcal{A}$ est obtenu en se restreignant aux éléments $\{e_i, i \in \mathcal{A}\}$, c'est-à-dire en ne considérant que les colonnes de D dont l'indice est inclus dans $\mathcal{A}$. $\widetilde{\alpha}$ est alors le vecteur déduit de $\alpha_\mathcal{A}$ en considérant des coefficients nuls pour les dimensions du problème (IV.27) d'origine non considérées dans $\mathcal{A}$.

données : D, y
$\mathcal{A} = \varnothing$, $\alpha_{\mathcal{A}} = 0$;
répéter
  $\mathcal{S} = \{$au plus n_{sel} indices $i \notin \mathcal{A}$ permettant de réduire $\Omega^*(D^\mathsf{T}(D\widetilde{\alpha_{\mathcal{A}}} - y))\}$;
  Utilisant $\alpha_{\mathcal{A}}$ comme position initiale, trouver la solution optimale $\alpha_{\mathcal{A}\cup\mathcal{S}}$ du problème $\min_\alpha \frac{1}{2}||y - D_{\mathcal{A}\cup\mathcal{S}}\alpha||_2^2 + \lambda\Omega(\widetilde{\alpha})$;
  $\mathcal{A} = \mathcal{A} \cup \mathcal{S}$;
jusqu'à $\Omega^*(D^\mathsf{T}(D\widetilde{\alpha_{\mathcal{A}}} - y)) \leq \lambda$;
retourner $\widetilde{\alpha_{\mathcal{A}}}$;

Algorithme IV.2 : Méta-algorithme avec ensembles actifs pour le calcul de représentations parcimonieuses induites par une norme Ω.

Un vecteur α est alors une solution globale du problème (IV.27) si et seulement si [5] :

(i) $\Omega^*(\nabla f(\alpha)) \leq \lambda$,

(ii) $-\frac{1}{\lambda}(\nabla f(\alpha))^\mathsf{T}\alpha = \Omega(\alpha)$.

Néanmoins, lorsque le vecteur $\alpha_{\mathcal{A}}$ est une solution optimale du sous-problème (IV.32), $\widetilde{\alpha_{\mathcal{A}}}$ est une solution globale du problème général (IV.27) si et seulement si la condition (i) est satisfaite [5].

L'algorithme III.3 peut ainsi être adapté au cas de normes Ω quelconques. Les seules différences sont que l'on cherche ici à respecter la condition $\Omega^*(\nabla f(\alpha_{\mathcal{A}})) \leq \lambda$, et l'ensemble actif sera donc augmenté avec certains indices i dont on estime qu'ils permettront de réduire $\Omega^*(\nabla f(\alpha_{\mathcal{A}}))$ (différentes stratégies sont envisageables à ce niveau). L'algorithme s'arrête lorsque la condition (i) est vérifiée, et le vecteur $\widetilde{\alpha_{\mathcal{A}}}$ est alors une solution globale du problème (IV.27). Cette approche par ensembles actifs est décrite dans l'algorithme IV.2.

Optimisation des représentations en norme $l_{\infty,1}$ pondérée

Nous pouvons maintenant expliquer comment les représentations parcimonieuses en norme $l_{\infty,1}$ pondérée peuvent être efficacement calculées en utilisant les résultats généraux précédents. Nous considérons ici le problème (IV.26) :

$$\min_\alpha \frac{1}{2}||y - D\alpha||_2^2 + \lambda||\alpha||_{\infty,1}^w, \tag{IV.33}$$

qui correspond au problème plus général (IV.27) en choisissant une norme Ω définie par :

$$\Omega(\alpha) = ||\alpha||_{\infty,1}^w = \max_{i=1..\Delta t} w_i||\alpha_{|_{G_i}}||_1. \tag{IV.34}$$

Afin de résoudre le problème (IV.26) par une méthode de gradient proximal accélérée (APG) en exploitant des ensembles actifs, il est nécessaire de déterminer la norme duale de la norme $l_{\infty,1}$ pondérée ainsi que les opérateurs proximaux associés à cette norme. Nous montrons en annexe B que l'application $\Omega(\alpha) = ||\alpha||_{\infty,1}^w$ proposée ici est effectivement une norme, et que sa norme duale vérifie :

$$||\alpha||_{\infty,1}^{w\,*} = ||\alpha||_{1,\infty}^{1/w} = \sum_{i=1..\Delta t} \frac{1}{w_i}||\alpha_{|_{G_i}}||_\infty\,, \tag{IV.35}$$

ce qui signifie que la norme duale associée à la norme $l_{\infty,1}$ pondérée est en fait une norme $l_{1,\infty}$ pondérée, notée $l_{1,\infty}^{1/w}$. L'avantage est ici que la norme $l_{1,\infty}$ est une norme de groupe plus classique (menant à des régularisations du type Group-Lasso) et a donc été étudiée dans plusieurs autres contextes. En particulier, il a été proposé dans l'article [103] un algorithme efficace pour calculer la projection $\Pi_{l_{1,\infty}\leq 1}$ en $O(nlog(n))$.

Pour déterminer les opérateurs proximaux de la norme $l_{\infty,1}$ pondérée, nous utilisons donc la relation :

$$prox_{\gamma l_{\infty,1}^w}(u) = u - \gamma\Pi_{(l_{\infty,1}^w)^*\leq 1}(u/\gamma) = u - \gamma\Pi_{l_{1,\infty}^{1/w}\leq 1}(u/\gamma), \tag{IV.36}$$

et les projections $\Pi_{l_{1,\infty}^{1/w}\leq 1}$ sont calculées en utilisant une version légèrement modifiée de l'algorithme de l'article [103] (une modification basique de l'algorithme est nécessaire pour prendre en compte la pondération des groupes). Ainsi, les opérateurs proximaux $prox_{\gamma\Omega}$ peuvent être évalués efficacement en $O(nlog(n))$.

La norme duale étant déterminée et les opérateurs proximaux pouvant être efficacement calculés, il est possible de déterminer les représentations parcimonieuses en norme $l_{\infty,1}$ pondérée avec l'optimisation par ensembles actifs décrite par l'algorithme IV.2. L'optimisation du sous-problème (IV.32) sur l'ensemble actif $\mathcal{A}$ est alors effectuée en employant une méthode de gradient proximal accélérée, comme décrite par l'algorithme III.2, en pré-calculant la matrice de Gram du dictionnaire D. Le dernier élément à définir est la stratégie d'agrandissement de l'ensemble actif $\mathcal{A}$, à savoir comment déterminer à chaque itération de l'algorithme IV.2 les indices $\mathcal{S}$ qui vont permettre de diminuer la valeur :

$$\Omega^*(D^\mathsf{T}(D\widetilde{\alpha_\mathcal{A}} - y)) = \sum_{i=1..\Delta t} \frac{1}{w_i} |||[D^\mathsf{T}(D\widetilde{\alpha_\mathcal{A}} - y)]_{|G_i}||_\infty \,. \tag{IV.37}$$

Une stratégie basique, mais suffisante en pratique, est de sélectionner au plus n_{sel} indices i non inclus dans $\mathcal{A}$ qui maximisent $|e_i^\mathsf{T}(D\widetilde{\alpha_\mathcal{A}} - y)|$, tout en se limitant à sélectionner au plus un indice par groupe G_k afin d'éviter de se focaliser sur un faible nombre de groupes.

IV.4 Évaluations et analyse des résultats

IV.4.1 Protocole d'évaluation et implémentation

Bases de données et métriques employées

Les bases de données du *MOTChallenge* sont utilisées afin d'évaluer notre approche et de la comparer à d'autres méthodes récentes, en considérant à la fois la version 2015, intitulée *2DMOT2015*, et 2016, intitulée *MOT16*. Ces bases de données ont été présentées en sous-section II.4.1, nous donnons donc seulement ici une description plus succincte. Ces deux bases de données sont constituées de plusieurs vidéos qui se répartissent dans un ensemble d'entraînement et un ensemble de test. Ces vidéos ne concernent que le suivi de personnes, et des détections publiques sont disponibles pour permettre de comparer les méthodes de suivi avec le même détecteur de personnes en entrée (le détecteur étant de type ACF (*Aggregate Channel Features*) [33] pour la version *2DMOT2015* et un DPM (*Deformable Part Model*) [39]

pour la version *MOT16*). L'ensemble de vidéos d'entraînement est supposé être employé pour régler les paramètres libres utilisés sur l'ensemble de test, et les résultats de suivi sont évalués en ligne.

Nous utilisons les métriques données par le *MOTChallenge*, qui incluent notamment les métriques CLEARMOT [14] comme le MOTA, MOTP, le nombre de changements d'identié (IDS), le nombre de faux positifs (FP) et de faux négatifs (FN). D'autres métriques sont fournies, comme le nombre de fragmentations (FM), le nombre de fausses alarmes par image (FAF) et le pourcentage de trajectoires majoritairement suivies et majoritairement perdues (respectivement MT et ML). Nous indiquons de plus le ratio de changements d'identité (IR) pour analyser plus précisément ces changements d'identité et de manière plus indépendante du taux de rappel. Ces différentes métriques ont été discutées de façon plus approfondie, en détaillant plus formellement leurs définitions, en sous-section II.4.2.

Implémentation

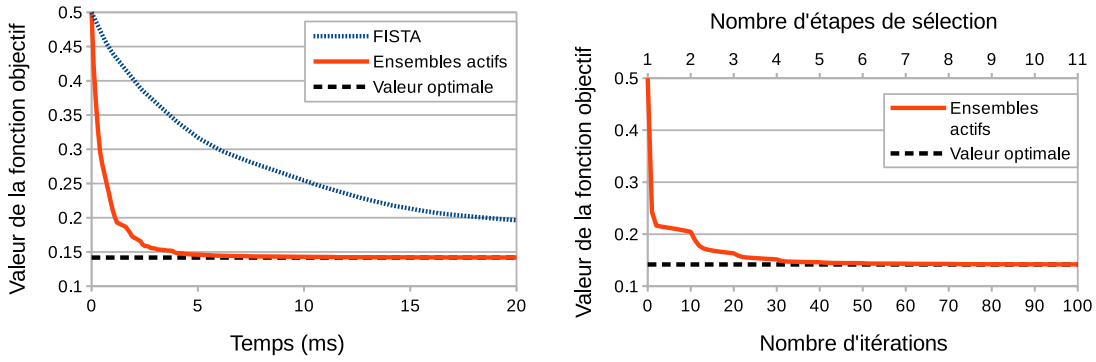
Notre approche est implémentée en C++ et est testée sur une machine exploitant un CPU multi-coeurs (4 coeurs, 8 threads) à 2.7 GHz. Seul le calcul des représentations parcimonieuses des détections peut tirer avantage d'une parallélisation naïve, en calculant séparément les représentations des détections d'une même image. L'optimisation de l'énergie E, par échantillonnage de type MCMC, est plus délicate à paralléliser et est donc réalisée sans parallélisation sur un seul coeur.

La bibliothèque OpenCV [5] est utilisée, principalement pour la gestion des images et l'extraction des caractéristiques, ainsi que la bibliothèque Eigen de calcul matriciel qui est employée pour notre implémentation des méthodes d'optimisation de gradient proximal. Nous nous sommes de plus fortement inspirés de l'implémentation en C donnée par les auteurs de l'article [103] qui réalise les projections sur la boule unité de la norme $l_{1,\infty}$. Ce code a été adapté afin de calculer les projections pour une norme $l_{1,\infty}$ pondérée en C++, nécessaire pour le calcul des opérateurs proximaux de la norme $l_{\infty,1}$ pondérée comme indiqué en sous-section IV.3.3.

L'optimisation des représentations parcimonieuses est effectuée, avec une méthode de gradient proximal accélérée avec ensembles actifs et pré-calcul de la matrice de Gram, comme décrit par l'algorithme III.3. En pratique, de la même façon qu'au chapitre III, l'algorithme III.3 est utilisé avec au maximum dix étapes d'agrandissement, ou d'étapes de sélection, de l'ensemble actif $\mathcal{A}$ en sélectionnant au plus dix nouveaux indices ($n_{sel} = 10$). L'optimisation des représentations parcimonieuses sur l'ensemble actif $\mathcal{A}$ est réalisée avec une méthode FISTA limitée à dix itérations. L'étude de la convergence effective de cette optimisation avec ce choix de paramètres est indiquée en figure IV.9, et justifie le gain en temps de calcul par rapport à une optimisation sans ensembles actifs. Le nombre de mouvements utilisé pour l'association de données par MCMCDA est fixé à 10000. Ce nombre de mouvements permet un compromis raisonnable entre la qualité de l'optimisation de l'énergie E et le temps de calcul de notre approche. Nous étudions en figure IV.10 l'impact de ce nombre de mouvements sur les performances en MOTA sur l'ensemble d'entraînement de la base de données *2DMOT2015*.

Avec ce choix de paramètres pour le calcul des représentations parcimonieuses et l'optimisation de l'énergie E, notre approche traite l'ensemble des vidéos de test de

5. URL : *opencv.org*



(a) (b)

FIGURE IV.9 – Évaluation expérimentale de la convergence de l'optimisation des représentations et du gain de temps avec l'emploi des ensembles actifs. La valeur moyenne de la fonction objectif est affichée pour la reconstruction de dix détections choisies aléatoirement sur la vidéo PETS S2L2 avec les détections publiques de la version 2015 du *MOTChallenge*. (a) : Comparaison de la vitesse de convergence entre l'algorithme FISTA et la méthode avec ensembles actifs. (b) : Vitesse de convergence de l'approche avec ensembles actifs en fonction du nombre d'étapes d'agrandissement de l'ensemble actif $\mathcal{A}$ et du nombre d'itérations utilisées pour résoudre le sous-problème sur ces ensembles actifs.

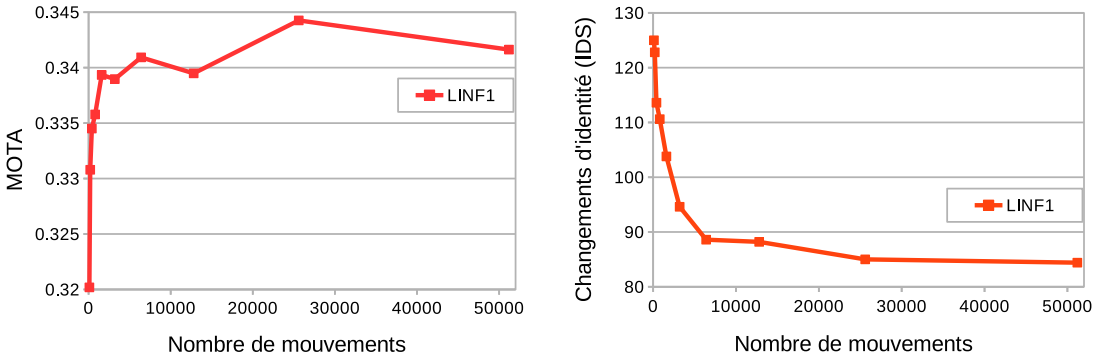


FIGURE IV.10 – Évaluation de l'impact du nombre de mouvements pour notre approche **LINF1**. Performance en MOTA et IDS évaluée sur l'ensemble d'entraînement de la base de données *2DMOT2015*, en considérant la moyenne des résultats pour cinq fonctionnements de notre approche en utilisant des graines différentes. Les performances atteignent un plateau peu avant 10000 mouvements, ce qui permet de justifier ce choix de nombre de mouvements comme compromis entre les performances et la vitesse de fonctionnement de notre méthode.

la base de données *2DMOT2015* à 7.5 images par seconde, ce qui permet un fonctionnement proche du temps réel. Il est cependant possible de dégrader la qualité de l'optimisation des représentations parcimonieuses ou de l'énergie E pour gagner en temps d'exécution au prix d'une perte en performances. Un avantage non négligeable de notre approche, du fait de l'utilisation d'une méthode d'échantillonnage par MCMC, est que nous pouvons terminer prématurément l'optimisation de l'énergie E sans réaliser le nombre limite de mouvements. Cette possibilité peut s'avérer

particulièrement utile dans des applications où le temps de traitement autorisé pour chaque image est variable.

Procédure d'hyper-optimisation

L'approche de suivi proposée ici dépend de nombreux paramètres, en particulier ceux intervenant dans la formulation de l'énergie E. Ce nombre important de paramètres pose deux problèmes principaux. Tout d'abord, déterminer manuellement un jeu correct de paramètres devient une tâche très laborieuse et coûteuse en temps. Secondement, cela complique la comparaison entre des variantes de notre approche, comparaisons nécessaires pour évaluer la pertinence des choix effectués. En effet, comparer deux variantes de notre approche sur un même jeu de paramètres par défaut n'est pas réellement satisfaisant car ce jeu de paramètres peut fortement biaiser la comparaison en étant favorable à l'une des variantes. Une technique courante consiste à déterminer manuellement le meilleur jeu de paramètres pour la méthode proposée et d'utiliser ce jeu de paramètres pour justifier la pertinence des différents modules en les remplaçant par des modules plus usuels. Dans ce cas de figure, le jeu de paramètre employé a fortement tendance à favoriser la méthode initiale puisqu'il a été déterminé de manière à être adapté pour cette méthode. Il est possible de chercher à déterminer manuellement un jeu de paramètres pour chaque variante évaluée, mais cela nécessite un temps très conséquent et il n'est pas possible de garantir que cette recherche manuelle soit réellement équitable pour toutes les variantes.

Dans le chapitre III, ce problème se posait pour un paramètre en particulier, le seuil d'affinité θ, dont la valeur optimale dépendait fortement des variantes. La solution retenue était de trouver manuellement les autres paramètres, qui ne semblaient pas favoriser de manière notable certaines variantes, et de déterminer pour chaque variante la meilleure valeur du seuil θ avec une recherche exhaustive. Cette stratégie n'est pas réellement applicable ici car plusieurs paramètres, en particulier ceux intervenant dans l'énergie E, dépendent fortement les uns des autres et nécessitent d'être déterminés conjointement. L'espace des paramètres à estimer est alors trop grand pour qu'une recherche exhaustive, sur une grille, soit efficace.

Cette problématique est traitée par les méthodes d'hyper-optimisation qui visent à estimer les hyper-paramètres d'un algorithme en testant plusieurs jeux d'hyper-paramètres et en exploitant les résultats des tests précédents pour orienter la recherche de ces paramètres vers des configurations plus propices. De telles méthodes d'hyper-optimisation ne nécessitent pas d'a-priori sur l'algorithme utilisant les paramètres recherchés et permettent de les déterminer plus efficacement comparés à des recherches exhaustives ou aléatoires. Nous employons donc une méthode d'hyper-optimisation pour déterminer automatiquement les paramètres de notre approche et ceux des variantes étudiées. Le programme SMAC est utilisé pour évaluer 1000 jeux de paramètres pour chaque variante, ce programme étant l'implémentation de l'approche [52] qui utilise des méthodes d'apprentissage statistique à base d'arbres aléatoires pour orienter la recherche des configurations.

IV.4.2 Évaluation de l'apport des représentations en norme $l_{\infty,1}$ et impact de la taille de la fenêtre glissante

Pour valider l'emploi de représentations parcimonieuses induites par une norme $l_{\infty,1}$, l'approche proposée est comparée ici à trois variantes. Ces trois variantes, dénommées **L1**, **MEAN** et **NN**, se différencient uniquement de notre approche générale, dénommée **LINF1**, au niveau du terme d'apparence $App(C)$ employé dans l'énergie E. Tout d'abord, nous comparons notre approche à une première variante **L1** qui emploie des représentations parcimonieuses plus usuelles en norme l_1. Nous comparons aussi notre approche avec deux variantes **MEAN** et **NN** qui emploient des modèles d'apparences $App(C)$ plus basiques, sans représentations parcimonieuses. Les termes d'apparence de ces variantes sont appelés App_{L1}, App_{MEAN} et App_{NN} et sont définis de la manière suivante :

(i) Pour la variante **L1**, $App_{L1}(C)$ est défini de la même manière que le terme $App(C)$ l'approche proposée $LINF1$ en équation (IV.18) :

$$App_{L1}(C) = \sum_{\tau \in C} \sum_{d \in \tau} [1 - \sum_{i} |\alpha^{\tau}_{y_d}(i)|] \,, \tag{IV.38}$$

avec néanmoins cette fois les représentations parcimonieuses α_{y_d} définies à partir d'une norme l_1 usuelle, en minimisant le problème :

$$\min_{\alpha} \ \frac{1}{2}||y_d - D\alpha||_2^2 + \lambda||(\alpha)||_1 \,. \tag{IV.39}$$

(ii) Pour la variante **MEAN**, le terme $App_{MEAN}(C)$ est défini par :

$$App_{MEAN}(C) = \sum_{\tau \in C} \sum_{d \in \tau} ||y_d - y_\tau||_2 \,, \tag{IV.40}$$

où y_τ correspond à la moyenne des caractéristiques $(y_d)_{d \in \tau}$ de la piste τ. Ce terme favorise ainsi les pistes où les détections sont d'apparence similaire.

(iii) Pour la variante **NN**, le terme $App_{NN}(C)$ est défini par :

$$App_{NN}(C) = \sum_{\tau \in C} \sum_{d \in \tau} ||y_d - NN_\tau(y_d)||_2 \,, \tag{IV.41}$$

où $NN_\tau(y_d)$ correspond au plus proche voisin de y_d parmi les caractéristiques des autres détections de τ, c'est-à-dire $NN_\tau(y_d) = \arg\min_{y_{d'}, d' \in \tau, d' \neq d} ||y_d - y_{d'}||_2$. Ce terme favorise ainsi les pistes τ où toute détection est proche en apparence d'au moins une autre détection de τ. Ce terme permet alors d'accepter plus de variabilité en apparence comparé au terme $App_{MEAN}(C)$.

L'approche proposée, **LINF1**, est comparée aux trois variantes **L1**, **MEAN** et **NN**. Plusieurs largeurs Δt pour la fenêtre glissante sont envisagées pour chacune des méthodes, avec $\Delta t \in \{5, 10, 15, 20\}$. Afin de permettre une comparaison la plus pertinente possible, les jeux de paramètres pour chaque méthode et pour chaque largeur Δt de la fenêtre glissante sont déterminés par la procédure d'hyper-optimisation discutée précédemment.

Les valeurs du MOTA et du nombre de changements d'identité (IDS) sont indiquées en figure IV.11 pour toutes les variantes expérimentées en fonction de la largeur Δt de la fenêtre glissante. Nous nous limitons à ces deux métriques car ces dernières

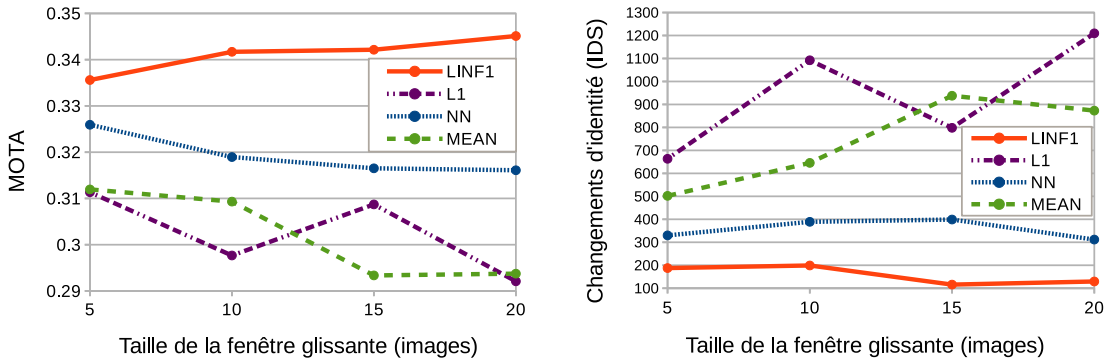


FIGURE IV.11 – Résultats en MOTA et en changements d'identité (IDS) de notre approche **LINF1** et de variantes avec d'autres modèles d'apparence pour des fenêtres glissantes incluant différents nombres Δt d'images. Résultats obtenus en s'évaluant sur l'ensemble d'entraînement de la base de donnée *2DMOT2015* après une procédure d'hyper-optimisation spécifique à chaque couple de variante et de taille de fenêtre glissante.

2DMOT2015 - Base d'entraînement - Détections publiques											
Méthode	Δt	**MOTA** ↑	**IDS** ↓	**IR** ↓	**FM** ↓	**FAF** ↓	**MOTP** ↑	**FP** ↓	**FN** ↓	**MT** (%) ↑	**ML** (%) ↓
LINF1	**20**	**0.345**	**129**	**2.9**	**385**	**0.7**	**0.728**	3931	**22073**	**18.0**	57.1
NN	**5**	0.326	330	7.7	533	**0.7**	0.727	**3779**	22790	16.4	54.0
MEAN	**5**	0.312	502	11	525	0.8	0.727	4379	22576	16.2	54.4
L1	**5**	0.311	663	15	663	**0.7**	0.726	4000	22818	15.7	**53.5**

Tableau IV.1 – Résultats pour les différentes variantes **LINF1**, **NN**, **MEAN** et **L1** testées sur l'ensemble d'entraînement de la base de données *2DMOT2015*. La largeur de fenêtre glissante $\Delta t \in \{5, 10, 15, 20\}$ maximisant les performances en termes de MOTA est utlisée pour chaque méthode (meilleures valeurs en gras et rouge, secondes meilleures soulignées en bleu).

sont parmi les plus considérées pour comparer les approches de suivi. Le MOTA est en effet la métrique principalement considérée pour comparer les approches de suivi multi-objets, comme discuté à la sous-section II.4.3, tandis que le nombre de changements d'identité (IDS) permet de juger plus spécifiquement la qualité des trajectoires et de l'association de données. Tout d'abord, l'approche proposée **LINF1** surpasse les autres variantes évaluées en termes de MOTA et de changements d'identité (IDS). De plus, seule cette approche arrive à gagner en performances lorsque davantage d'images sont considérées au sein de la fenêtre glissante, à la fois au niveau du MOTA et des changements d'identité. En effet, toutes les autres variantes, **L1**, **MEAN** et **NN**, voient globalement leur performance en MOTA diminuer lorsque la largeur de la fenêtre glissante Δt augmente. C'est aussi le cas pour les changements d'identité qui augmentent avec une plus grande fenêtre glissante pour ces variantes, à l'exception de la variante **NN** dont le nombre de changements d'identité reste plus ou moins stable.

L'ensemble des métriques pour chacune des approches **LINF1**, **L1**, **MEAN** et **NN** est indiqué en tableau IV.1, en sélectionnant pour chaque variante la largeur

2DMOT2015 - Base d'entraînement - Détections publiques											
Méthode	Δt	**MOTA** ↑	**IDS** ↓	**IR** ↓	**FM** ↓	**FAF** ↓	**MOTP** ↑	**FP** ↓	**FN** ↓	**MT** (%) ↑	**ML** (%) ↓
LINF1	**5**	0.336	188	4.4	413	**0.6**	**0.728**	**3346**	22980	16.5	57.8
	10	0.342	199	4.5	444	0.7	0.726	3740	22330	17.8	59.7
	15	0.342	**116**	**2.6**	410	0.7	0.727	3829	22307	18.4	58.8
	20	**0.345**	129	2.9	**385**	0.7	**0.728**	3931	22073	18.0	57.1
	25	0.341	163	3.3	470	1.1	0.724	6200	**19942**	**22.2**	**50.6**
	30	0.338	155	3.3	446	1.0	0.726	5260	20997	20.8	53.2
	35	0.336	141	3.0	446	1.0	0.727	5455	20909	20.4	51.4

Tableau IV.2 – Résultats de l'approche proposée **LINF1**, avec des largeurs de fenêtre glissante $\Delta t \in \{5, 10, 15, 20, 25, 30, 35\}$, sur l'ensemble d'entraînement de la base de données *2DMOT2015* (meilleures valeurs en gras et rouge, secondes meilleures soulignées en bleu).

de la fenêtre glissante Δt qui donne la meilleure valeur en MOTA. L'approche employant des représentations parcimonieuses usuelles en norme l_1, **L1**, est celle qui présente les moins bonnes performances avec en particulier un nombre élevé de changements d'identité (IDS) et de fragments (FM). Ces résultats sont cohérents avec la structure des représentations impliquées, qui a tendance à promouvoir un grand nombre de trajectoires courtes de détections similaires. Les deux autres variantes, **MEAN** et **NN**, donnent de meilleurs résultats tout en restant significativement inférieures en termes de MOTA, changements d'identité (IDS), de fragmentation (FM) et de trajectoires majoritairement suivies (MT) à ceux de l'approche proposée **LINF1**.

Les gains observés pour notre approche **LINF1** par rapport aux autres variantes évaluées, en particulier en changements d'identité (IDS) et en fragmentation (FM), peuvent s'expliquer pour les raisons suivantes. Tout d'abord, les représentations parcimonieuses en norme $l_{\infty,1}$ évitent la fragmentation des trajectoires en favorisant des trajectoires longues. En effet, toute détection est représentée au sein de sa représentation par des détections de l'ensemble des images de la fenêtre glissante. Cela signifie que pour réduire les erreurs de reconstruction résiduelle, les pistes estimées doivent elles-aussi s'étendre tout au long de la fenêtre glissante. De plus, le caractère parcimonieux des représentations en norme $l_{\infty,1}$ au sein de chaque groupe signifie que ces représentations font intervenir un faible nombre de détections par image. Cela met les détections d'une même image en compétition les unes avec les autres dans les représentations et permet de mieux discriminer les détections entre elles. La baisse significative des changements d'identité par rapport aux autres variantes peut s'expliquer par cette propriété des représentations en norme $l_{\infty,1}$. Comparé aux autres variantes, l'approche **LINF1** présente un taux de trajectoires majoritairement perdues (ML) supérieur et est moins performante que toutes les autres méthodes pour cette métrique spécifique. Puisque le taux de trajectoire majoritairement suivi (MT) est néanmoins supérieur aux autres variantes, cela indique que notre approche estime un nombre moins important de trajectoires mais avec des trajectoires estimées sur une période plus longue.

Nous avons de plus évalué notre approche pour des largeurs Δt de fenêtre glissante allant au-delà de 20 images (jusqu'à 35 images), ces résultats étant présentés

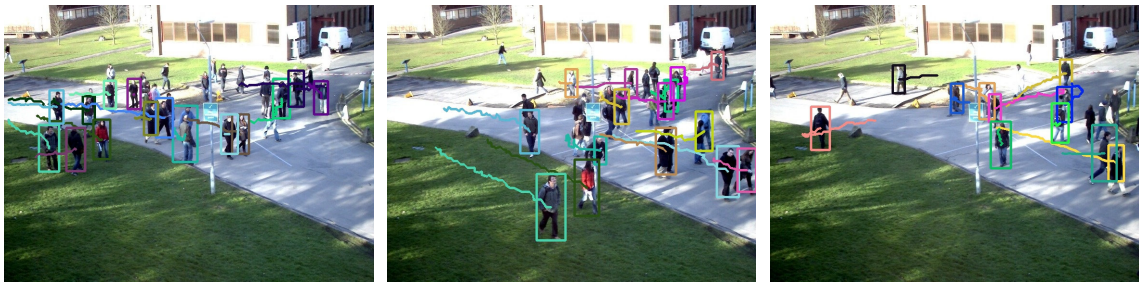


Image n°50 **Image n°100** **Image n°150**
PETS S2L2 (LINF1)

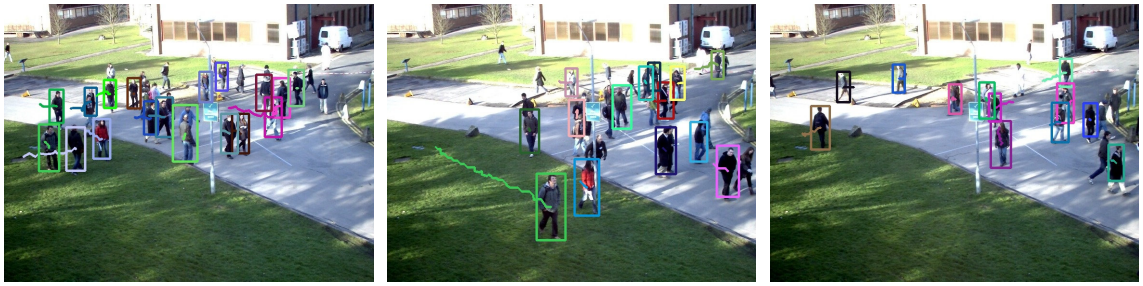


Image n°50 **Image n°100** **Image n°150**
PETS S2L2 (L1)

FIGURE IV.12 – Comparaison de certains résultats donnés par l'approche **LINF1** à ceux donnés par la variante **L1** sur l'ensemble de test de la base de données *2DMOT2015* en utilisant les détections publiques fournies. L'approche **L1** produit alors des trajectoires bien plus courtes que l'approche **LINF1**.

en tableau IV.2. La largeur de fenêtre Δt idéale se situe aux alentours de 15-20 images pour l'ensemble des métriques, et dans la plage de 10-25 images en ne considérant que le MOTA. L'approche proposée gagne en performances lorsque davantage d'images sont considérées dans la fenêtre glissante car les représentations en norme $l_{\infty,1}$ favorisent l'implication de l'ensemble des images de la fenêtre glissante. Les performances générales se dégradent néanmoins pour des tailles de fenêtres importantes, à partir de 30 images. L'espace de recherche du MCMCDA, c'est-à-dire l'ensemble des configurations possibles, croit significativement avec la largeur de la fenêtre glissante et rend l'optimisation plus difficile. Cet aspect permet de donner une explication possible à cette perte de performances.

Des trajectoires estimées par les approches **LINF1** et **L1** sont données en figure IV.12. On peut voir ainsi que la méthode **LINF1** génère effectivement des trajectoires plus longues que l'approche **L1**, qui produit des trajectoires bien plus courtes, en accord avec les résultats quantitatifs obtenus pour ces deux approches. Ces observations confirment la pertinence de la norme $l_{\infty,1}$ proposée pour traiter correctement le problème d'association de données multi-images, notamment comparée à l'emploi d'une norme l_1 plus usuelle. Ainsi, le choix d'une structure de parcimonie adaptée à notre problème particulier s'avère crucial pour obtenir de bons résultats de suivi.

2DMOT2015 - Base de test - Détections publiques												
Méthode	**Réf.**	**T.**	**MOTA** ↑	**IDS** ↓	**IR** ↓	**FM** ↓	**FAF** ↓	**MOTP** ↑	**FP** ↓	**FN** ↓	**MT** (%)↑	**ML** (%)↓
NOMT	**[24]**	**H**	**0.337**	442	9.4	823	1.3	**0.719**	7762	32547	12.2	44.0
MHT_DAM	**[62]**	**H**	0.324	435	9.1	826	1.6	0.718	9064	**32060**	**16.0**	43.8
MDP	**[127]**	**E**	0.303	680	14	1500	1.7	0.713	9717	32422	13.0	**38.4**
LP_SSVM	**[122]**	**H**	0.252	646	16	849	1.4	0.717	8369	36932	5.8	53.0
ELP	**[84]**	**H**	0.250	1396	36	1804	1.3	0.712	7345	37344	7.5	43.8
LINF1	**-**	**H**	0.245	**298**	**8.6**	744	**1.0**	0.713	**5864**	40207	5.5	64.6
JPDA_m	**[108]**	**H**	0.238	365	11	869	1.1	0.682	6373	40084	5.0	58.1
MotiCon	**[69]**	**H**	0.231	1018	24	1061	1.8	0.709	10404	35844	4.7	52.0
SegTrack	**[90]**	**H**	0.225	697	19	**737**	1.4	0.717	7890	39020	5.8	63.9
DCO_X	**[89]**	**H**	0.196	521	14	819	1.8	0.714	10652	38232	5.1	54.9
CEM	**[88]**	**H**	0.193	813	19	1023	2.5	0.707	14180	34591	8.5	46.5
RMOT	**[130]**	**E**	0.186	684	17	1282	2.2	0.696	12473	36835	5.3	53.3
SMOT	**[32]**	**H**	0.182	1148	33	2132	1.5	0.712	8780	40310	2.8	54.8
ALExTR.	**[15]**	**H**	0.170	1859	53	1872	1.6	0.712	9233	39933	3.9	52.4
TBD	**[44]**	**H**	0.159	1939	45	1963	2.6	0.709	14943	34777	6.4	47.9
GSCR	**[36]**	**E**	0.158	514	18	1010	1.3	0.694	7597	43633	1.8	61.0
TC_ODAL	**[6]**	**E**	0.151	637	17	1716	2.2	0.705	12970	38538	3.2	55.8
DP_NMS	**[102]**	**H**	0.145	4537	105	3090	2.3	0.708	13171	34814	6.0	40.8

Tableau IV.3 – Résultats de l'approche proposée **LINF1** sur l'ensemble de test de la base de données *2DMOT2015*. Résultats comparés à ceux des autres approches de l'état de l'art publiées sur cette base de données au 14/03/2016 (meilleures valeurs en gras et rouge, secondes meilleures soulignées en bleu). Troisième colonne : type d'approche avec E pour en ligne et H pour hors ligne.

MOT16 - Base de test - Détections publiques												
Méthode	**Réf.**	**T.**	**MOTA** ↑	**IDS** ↓	**IR** ↓	**FM** ↓	**FAF** ↓	**MOTP** ↑	**FP** ↓	**FN** ↓	**MT** (%)↑	**ML** (%)↓
LINF1	**-**	**H**	**0.405**	**426**	**9.4**	953	1.4	0.749	8401	**99715**	**10.7**	56.1
DP_NMS	**[102]**	**H**	0.319	969	29	**941**	**0.2**	**0.764**	**1343**	121813	4.8	65.2
SMOT	**[32]**	**H**	0.292	3072	75	4437	3.0	0.752	17929	108041	4.9	**53.3**

Tableau IV.4 – Résultats de l'approche proposée **LINF1** sur l'ensemble de test de la base de données *MOT16*. Résultats comparés à ceux des autres approches de l'état de l'art publiées sur cette base de données au 14/03/2016 (meilleures valeurs en gras et rouge, secondes meilleures soulignées en bleu). Troisième colonne : type d'approche avec E pour en ligne et H pour hors ligne.

Base de données	θ_{Ob}	θ_{App}	θ_{Mot}	θ_{Int}	α_{Ob}	β_{Ob}	γ_{Ob}	σ	λ	N_c	s_c	δt_l	d_l	h_l	N_{tr}
2DMOT2015	0.50	0.39	0.77	0.08	0.60	0.004	0.99	0.14	3.1	5	29	6	0.28	0.24	11
MOT16	0.33	0.40	0.77	0.41	1.3	0.99	1.3	0.15	7.7	5	0.13	18	0.29	0.30	18

Tableau IV.5 – Meilleurs jeux de paramètres pour l'approche **LINF1**, avec une fenêtre glissante de 20 images, déterminés sur les ensembles d'entraînement des bases de données *2DMOT2015* et *MOT16* en employant une procédure d'hyper-optimisation.

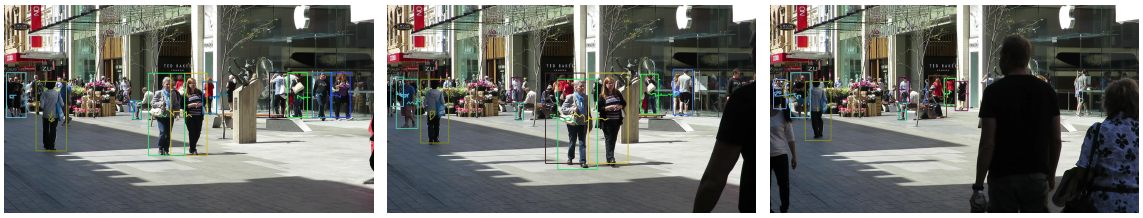


Image n°300 | **Image n°350** | **Image n°400**

ADL-Rundle-3

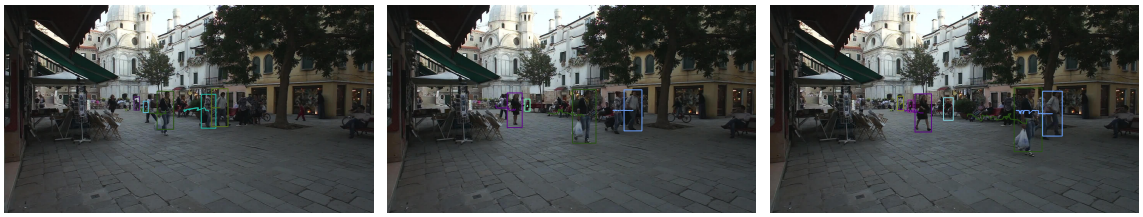


Image n°200 | **Image n°250** | **Image n°300**

Venice1

FIGURE IV.13 – Certaines trajectoires estimées par notre méthode **LINF1** sur l'ensemble de test de la base de données *2DMOT2015* avec les détections publiques.

IV.4.3 Comparaison aux méthodes récentes de l'état de l'art

Les résultats de notre approche **LINF1** sur l'ensemble de test de la base de données *2DMOT2015* sont indiqués dans le tableau IV.3. Certaines des trajectoires estimées sur ces vidéos sont de plus données en figure IV.13. Notre méthode est comparée aux autres approches publiées, au 14/03/2016, sur cette base de données en utilisant les détections publiques fournies. Afin de suivre les règles imposées par le *MOTChallenge*, nous utilisons le jeu de paramètres qui maximise le MOTA sur l'ensemble d'entraînement et qui est déterminé par la procédure d'hyper-optimisation décrite en sous-section IV.4.1. Le jeu de paramètre ainsi trouvé est indiqué au tableau IV.5.

En termes de MOTA, notre approche est supérieure ou comparable à la majorité des autres approches, à l'exception de [24, 62, 127]. Notre méthode se distingue néanmoins en termes de changements d'identité (IDS), le nombre de changements d'identité délivrés par notre approche étant significativement inférieur à celui des autres approches. Cela indique une meilleure capacité à distinguer les cibles similaires en évitant de confondre au sein d'une même trajectoire deux cibles différentes. Ce nombre de changements d'identité est cependant corrélé au taux de rappel, et de manière similaire au nombre de faux négatif FN, une méthode estimant davantage de cibles ayant naturellement tendance à produire davantage de changements d'identité. Pour limiter en partie ce biais, il est possible d'examiner le taux de changements d'identité (IR) qui revient à considérer le nombre de changements d'identité divisé par le taux de rappel. Notre approche **LINF1** est alors toujours en tête pour ce critère. Notre méthode est aussi la meilleure en termes de faux positifs (FP) et la seconde en termes de fragmentation (FM). Les résultats en termes de précision (MOTP) et de trajectoires majoritairement suivies (MT) sont dans la moyenne de ceux des autres approches, mais notre approche est par contre moins performante en termes de faux négatifs (FN) et de trajectoires majoritairement perdues (ML).

Ces résultats confirment l'analyse réalisée précédemment lorsque notre approche

LINF1 a été comparée à d'autres variantes, **L1**, **NN** et **MEAN**. La méthode proposée donne des résultats très fiables, c'est-à-dire avec peu de changements d'identité (IDS), de fragmentation (FM) et de fausses alarmes (FAF). Cela est dû à l'emploi des représentations parcimonieuses en norme $l_{\infty,1}$ qui permettent de prendre en compte efficacement l'ensemble des images de la fenêtre glissante tout en restant discriminatives entre les différentes détections en chaque image, ce qui limite les changements d'identité (IDS). La structure de parcimonie induite par ces représentations permet notamment de favoriser des trajectoires longues, s'étendant tout au long de la fenêtre glissante, ce qui permet de mieux gérer les occultations ou les périodes de non-détection et de limiter ainsi le nombre de changements d'identité (IDS) et de fragmentation (FM). Néanmoins, notre approche semble se concentrer sur un nombre plus faible de trajectoires pertinentes ce qui se traduit par des performances inférieures en termes de faux négatifs (FN) ou, de manière similaire, de taux de rappel, et de trajectoires majoritairement perdues (ML). Cette méthode est ainsi appropriée pour des applications où la précision des résultats est plus importante que le taux de rappel et où le maintien des identités des cibles est crucial.

Les résultats obtenus sur la version 2016 du *MOTChallenge*, *MOT16*, sont indiqués au tableau IV.4 (au 14/03/2016). Cette base de données ayant été publiée peu de temps avant l'évaluation de notre approche, les résultats de seulement deux autres approches de suivi étaient disponibles. Notre méthode donne des résultats largement supérieurs à ces méthodes, en particulier en termes de MOTA et de changements d'identité (IDS). Le jeu de paramètres utilisé, déterminé par une procédure d'hyper-optimisation sur l'ensemble de vidéos d'entraînement, est indiqué au tableau IV.5. Davantage de résultats pour cette base de données, ainsi que pour la version 2015 *2DMOT2015*, sont indiqués au chapitre suivant avec les résultats des approches de suivi publiées plus récemment (après le 14/03/2016).

Conclusion

Dans ce chapitre, nous avons proposé une approche de suivi multi-objets qui exploite à la fois une fenêtre glissante et des représentations parcimonieuses structurées des détections. Une énergie globale E est utilisée pour déterminer la meilleure configuration de pistes au sein de la fenêtre glissante et la formulation de cette énergie se distingue principalement des formulations existantes du fait de son terme d'apparence, qui prend en considération les représentations parcimonieuses des détections. Afin de travailler avec des représentations parcimonieuses plus adaptées au cadre du suivi multi-objets avec fenêtre glissante, nous proposons de définir ces représentations à partir d'une norme plus spécifique à ce problème. Le choix d'une norme $l_{\infty,1}$ pondérée a été retenu afin d'inciter toute détection à n'être représentée que par les détections liées à la même cible. Nous avons de plus montré que ces représentations particulières pouvaient être efficacement calculées avec des méthodes d'optimisation de gradient proximal et des stratégies avec ensembles actifs, en adaptant l'approche utilisée en sous-section III.3.2 pour les représentations parcimonieuses en norme l_1.

Combiner des représentations parcimonieuses déduites d'une norme $l_{\infty,1}$ pondérée avec une approche de suivi à fenêtre glissante permet d'exploiter efficacement l'ensemble des informations présentes au sein de la fenêtre. En effet, ces représentations favorisent des représentations faisant à la fois participer l'ensemble des images considérées dans la fenêtre glissante tout en restant parcimonieuses au niveau de

chaque image. Cela permet à la fois de favoriser le prolongement des trajectoires, ce qui réduit le nombre de fragmentations (FM), tout en mettant en compétition les détections de chaque image ce qui tend à réduire le nombre de changements d'identité (IDS). L'approche décrite dans ce chapitre a été évaluée sur les bases de données du *MOTChallenge* et les expérimentations effectuées mettent en évidence les gains en performances obtenus en employant les représentations parcimonieuses structurées proposées pour un suivi multi-objets à fenêtre glissante. Cette méthode se compare correctement à la majorité des autres compétiteurs du *MOTChallenge* en termes de MOTA et donne de très bons résultats sur certaines métriques comme le nombre de changements d'identité (IDS) ou de fragmentation (FM).

Les travaux de ce chapitre ont été publiés dans la conférence ECCV 2016 [37].

Chapitre V

Représentations parcimonieuses avec dictionnaires denses pour le suivi multi-objets

Sommaire

Introduction

Au sein de ce chapitre, nous considérons une extension de l'approche proposée au chapitre précédent en employant un nouveau type de dictionnaire. L'approche précédemment proposée est très dépendante des performances du détecteur d'objets, en partie car le dictionnaire considéré pour les représentations parcimonieuses ne fait intervenir que des détections. Une cible non détectée sur certaines images, ou détectée de façon imprécise, impacte fortement la qualité des représentations parcimonieuses des détections associées à cette cible. De ce fait, nous proposons

d'employer des dictionnaires, dits denses, qui ne se limitent plus aux seules détections du détecteur d'objets et qui incluent aussi des éléments provenant de positions non détectées.

La section V.2 est dédiée à la définition des dictionnaires denses proposés et du modèle d'apparence $App(C)$ associé. La façon dont les représentations structurées en norme $l_{\infty,1}$ peuvent être efficacement calculées avec de tels dictionnaires est ensuite détaillée. Le système de notre approche de suivi est présenté en section V.3. Dans la section V.4, l'implémentation de notre approche est précisée et de nombreuses évaluations quantitatives sont effectuées pour étudier l'apport des dictionnaires proposés. Afin d'évaluer la robustesse de notre approche par rapport à la qualité du détecteur, des tests sont de plus effectués en simulant des détecteurs plus ou moins performants à partir des vérités terrains suivant la méthodologie proposée dans l'article [115]. Cette extension est de plus comparée aux autres méthodes récentes de l'état de l'art en étant évaluée sur les bases de données du *MOTChallenge*.

V.1 Motivations

Les principales motivations qui justifient les hypothèses étudiées dans ce chapitre sont expliquées ci-après.

V.1.1 Limitations des dictionnaires à base de détections

L'approche proposée au chapitre IV exploite des représentations parcimonieuses structurées, qui représentent chaque détection, afin d'améliorer l'association de données dans une approche de suivi à fenêtre glissante. Cependant, les dictionnaires employés pour calculer ces représentations parcimonieuses ne font intervenir que des détections issues des trajectoires estimées ou de la fenêtre glissante. Cela signifie que chaque détection est uniquement représentée par d'autres détections.

Ce constat montre que cette précédente approche est très dépendante de la qualité du détecteur d'objets employé. En effet, dans le cas où le détecteur d'objets ne détecte qu'occasionnellement une cible O, une détection d liée à la cible O ne sera jamais correctement représentée par la représentation parcimonieuse α_{y_d} en norme $l_{\infty,1}$. La norme $l_{\infty,1}$ favorisant les représentations faisant intervenir des détections de chaque image de la fenêtre glissante, la représentation α_{y_d} aura tendance à exploiter des détections d'une cible différente pour les images où la cible O n'est pas détectée. Les coefficients de la représentation α_{y_d} liés à ces détections seront néanmoins plus faibles, car leurs caractéristiques visuelles seront plus éloignées de α_{y_d}. Si l'on considère la piste τ qui n'inclut que les détections liées à la cible O, l'erreur de reconstruction résiduelle $||y_d - D_\tau \alpha^\tau_{y_d}||_2$, utilisée dans le modèle d'apparence $App(C)$, sera toujours importante du fait des non-détections. En réalité, le principal problème est que la structure de parcimonie favorisée par la norme $l_{\infty,1}$ est adaptée lorsque toutes les cibles sont correctement détectées à chaque image, mais est moins appropriée pour les cibles qui ne sont qu'occasionnellement détectées.

L'approche que nous avons proposée au chapitre précédent, **LINF1**, n'utilise que les caractéristiques visuelles issues des détections sans chercher à exploiter davantage d'information visuelle présente dans les images. Plusieurs approches multi-objets récentes [24, 117, 127] utilisent au contraire l'intégralité des informations visuelles disponibles au sein des images. Cela est en particulier fait pour calculer des valeurs

d'affinité pertinentes entre détections, dans les approches hors ligne, ou entre les trajectoires estimées et les détections pour les approches en ligne (et éventuellement pour prolonger les trajectoires). Par exemple, l'approche [24] exploite des trajectoires de points d'intérêt, en utilisant un flot optique, pour déterminer des valeurs d'affinité pour chaque paire de détections. Une autre approche récente [117] se base sur des résultats de DeepMatching pour ses valeurs d'affinité entre détections, tandis que l'approche proposée dans [127] raisonne sur du flot optique propagé entre plusieurs images. Toutes ces différentes méthodes s'aident au final d'informations visuelles qui ne se limitent pas à la description des seules détections données par le détecteur.

Deux critiques peuvent ainsi être observées à l'encontre de l'approche **LINF1** que nous avons proposée au chapitre IV. Tout d'abord, notre modèle d'apparence $App(C)$ dans l'énergie globale E est fortement dépendant de la performance du détecteur employé. Cela est dû principalement au fait que les représentations parcimonieuses employées se basent sur un dictionnaire n'incluant que des détections données par le détecteur d'objets utilisé. Une seconde critique est que cette approche se limite à considérer l'information visuelle des détections, sans chercher à exploiter davantage d'information visuelle au sein des images qui est pourtant disponible.

V.1.2 Représentations parcimonieuses à convolutions

Une extension des représentations parcimonieuses, qui exploite des convolutions, a été étudiée au cours des dernières années [20, 23, 48, 109, 136]. Pour rappel, une représentation parcimonieuse, usuelle en norme l_1, peut être définie comme solution du problème :

$$\min_{\alpha \in \mathbb{R}^m} \frac{1}{2}||y - D\alpha||_2^2 + \lambda||\alpha||_1 \,. \tag{V.1}$$

L'élément y est alors représenté à partir des éléments $\{e_1, ..., e_m\}$ du dictionnaire D. Les représentations parcimonieuses à convolutions sont définies à partir d'une autre formulation. On suppose alors que l'élément reconstruit y se met sous la forme d'une matrice de $\mathbb{R}^{n_y, n_x}$, ce qui peut correspondre aux valeurs d'intensité, en niveaux de gris, d'une boîte dans une image. On considère de nouveau m éléments de $\mathbb{R}^{n_y, n_x}$, $\{e_1, ..., e_m\}$, chacun de même dimension que y. On cherche alors une représentation parcimonieuse de y sous la forme d'un ensemble de filtres $\alpha = \{\alpha_1, ..., \alpha_m\}$, avec $\alpha_i \in \mathbb{R}^{n_y, n_x}$, qui sont solutions du problème :

$$\min_{\alpha_1, ..., \alpha_m} \frac{1}{2}||y - \sum_{i=1...m} e_i * \alpha_i||_2^2 + \lambda \sum_{i=1...m} ||\alpha_i||_1 \,, \tag{V.2}$$

où $*$ représente ici l'opérateur de convolution 2D.

Ce nouveau problème peut en réalité se ramener à une formulation similaire à celle du problème (V.1). Pour chaque élément e, et pour tout déplacement $(u, v) \in [\![0, n_y - 1]\!] \times [\![0, n_x - 1]\!]$, on considère l'élément $e^{u,v}$ obtenu en effectuant un décalage de u lignes et v colonnes sur la matrice e. On considère désormais l'élément reconstruit y, les éléments $\{e_1, ..., e_m\}$ et leurs décalages $({e_1}^{u,v})_{u,v}, ..., ({e_m}^{u,v})_{u,v}$ comme des vecteurs de $\mathbb{R}^{n_y n_x}$ (par exemple en concaténant les valeurs d'intensité précédentes par colonnes). Pour chaque élément e, le dictionnaire D_e est défini comme le dictionnaire qui rassemble tous les décalages de e, $D_e = [e^{0,0}\ e^{0,1}\ ...\ e^{n_y-1, n_x-1}]$. On peut alors considérer le dictionnaire D qui rassemble ces sous-dictionnaires D_e :

$$D = [D_{e_1}\ ...\ D_{e_m}] \,. \tag{V.3}$$

Dans ce cas, le problème (V.2) est alors équivalent au problème (V.1) avec ce dictionnaire D (chaque filtre α_i, pour le problème (V.2), correspondant en fait aux coefficients de α liés au dictionnaire D_{e_i} dans le problème (V.1)). Il faut néanmoins bien réaliser que ce dictionnaire D contient un nombre très important d'éléments, $m \times n_x n_y$ avec les notations précédentes.

Ainsi, les représentations parcimonieuses à convolutions peuvent être vues comme des représentations parcimonieuses classiques, mais en utilisant un dictionnaire qui inclut tous les décalages d'un ensemble d'éléments $\{e_1, ..., e_m\}$. L'avantage principal de telles représentations est qu'elles permettent d'être davantage invariantes vis-à-vis des décalages de l'objet représenté. En pratique, les méthodes utilisant ces représentations cherchent, pour la plupart, à apprendre les éléments $\{e_1, ..., e_m\}$ en formulant un problème de type apprentissage de dictionnaire. Bien que le dictionnaire D inclue un grand nombre d'éléments, la structure particulière de ce dictionnaire permet de déployer des méthodes d'optimisation efficaces à base de méthodes proximales et de transformée de Fourier rapide (*Fast Fourier Transform*, FFT).

Un exemple récent d'application de telles techniques, pour le suivi mono-objet, a été proposé dans [136]. Cette approche résout un problème du type (V.2), sans chercher à apprendre les éléments du dictionnaire. L'idée est alors d'utiliser un système de suivi mono-objet à représentations parcimonieuses classiques, avec un filtre à particules, mais de chercher à estimer le décalage des particules par rapport à la cible à l'aide de représentations parcimonieuses à convolutions. Cela permet notamment de réduire considérablement le nombre de particules à considérer.

Nous avons vu ici des approches qui utilisent des représentations parcimonieuses sur des dictionnaires contenant bien plus d'éléments. Est-il possible de s'inspirer de ces approches pour étendre la méthode de suivi proposée au chapitre IV, en considérant des dictionnaires incluant d'autres éléments que les détections du détecteur d'objets ?

V.1.3 Principe de l'approche proposée

Dans ce chapitre, nous proposons d'étendre l'approche de suivi **LINF1** du chapitre IV en employant des représentations définies à partir de dictionnaires denses. L'idée principale est de définir des dictionnaires qui ne se limitent pas aux seules détections données par le détecteur d'objets, mais qui incluent un grand nombre d'éléments provenant de positions non détectées dans les images de la fenêtre glissante. L'objectif est alors d'employer de tels dictionnaires pour compenser des erreurs du détecteur, notamment des cibles occasionnellement non détectées ou bien des détections imprécises. L'emploi de ces dictionnaires nécessite d'adapter le modèle d'apparence $App(C)$ de l'approche précédente afin de pouvoir prendre en compte dans ce terme les éléments du dictionnaire qui ne sont pas associés à une détection.

Considérer des dictionnaires avec un nombre très important d'éléments pose néanmoins certains problèmes d'optimisation, les approches d'optimisation proposées dans le chapitre précédent n'étant plus suffisantes pour maintenir une vitesse de fonctionnement acceptable. Une partie non négligeable de ce chapitre sera donc dédiée à l'adaptation des méthodes proximales avec ensembles actifs pour traiter plus spécifiquement les dictionnaires denses que nous considérons. Cette adaptation des méthodes d'optimisation fait notamment intervenir des considérations sur des calculs rapides de corrélations croisées normalisées.

Les représentations parcimonieuses proposées s'inspirent des représentations parcimonieuses à convolutions présentées précédemment sans pour autant être identiques. Tout comme ces méthodes, nous exploitons néanmoins la structure particulière des dictionnaires concernés pour utiliser des outils adaptés qui permettent d'accélérer grandement les optimisations de ces représentations, en particulier par le calcul de convolutions par transformée de Fourier rapide.

V.2 Représentations avec dictionnaires denses en norme $l_{\infty,1}$

Dans cette section, nous précisons les dictionnaires denses qui sont considérés dans ce chapitre. Nous abordons aussi les problèmes liés à l'optimisation de représentations parcimonieuses définies vis-à-vis de tels dictionnaires et proposons une méthode d'optimisation appropriée.

V.2.1 Dictionnaires denses

Rappels et principe

Au chapitre IV, toute détection d de la fenêtre glissante est associée à une représentation parcimonieuse α_{y_d}. Cette représentation est déterminée comme solution du problème :

$$\min_{\alpha} \frac{1}{2}||y_d - D\alpha||_2^2 + \lambda||\alpha||_{\infty,1}^w. \tag{V.4}$$

Le dictionnaire D est alors constitué de plusieurs sous-dictionnaires :

$$D = [D_{\mathcal{T}}\ D_{I_{t-\Delta t+1}}\ ...\ D_{I_{t-1}}]\,, \tag{V.5}$$

où $D_{\mathcal{T}}$ correspond aux détections des trajectoires estimées au-delà de la fenêtre glissante tandis que chaque sous-dictionnaire D_I correspond aux détections de l'image I. Ces dictionnaires sont donc uniquement composés de caractéristiques visuelles liées à des détections.

Nous proposons ici de redéfinir les dictionnaires $D_{\mathcal{T}}$ et D_I de façon à inclure des caractéristiques visuelles de positions non associées à des détections. Contrairement au chapitre précédent, où le dictionnaire D ne dépendait que de l'instant courant et était identique pour toutes les détections de la dernière image I_t considérée, le dictionnaire $D(d)$ sera ici spécifique à chaque détection d. Pour chaque détection d, sa représentation parcimonieuse α_{y_d} sera alors définie comme solution du problème :

$$\min_{\alpha} \frac{1}{2}||y_d - D(d)\alpha||_2^2 + \lambda||\alpha||_{\infty,1}^w. \tag{V.6}$$

De la même manière qu'au chapitre précédent, le dictionnaire $D(d)$ est constitué de plusieurs sous-dictionnaires :

$$D(d) = [D_{\mathcal{T}}\ D_{I_{t-\Delta t+1}}(d)\ ...\ D_{I_{t-1}}(d)]\,, \tag{V.7}$$

où $D_{\mathcal{T}}$ est un dictionnaire lié à des détections au-delà de la fenêtre glissante tandis que chaque sous-dictionnaire $D_I(d)$ est associé à une image I de la fenêtre glissante.

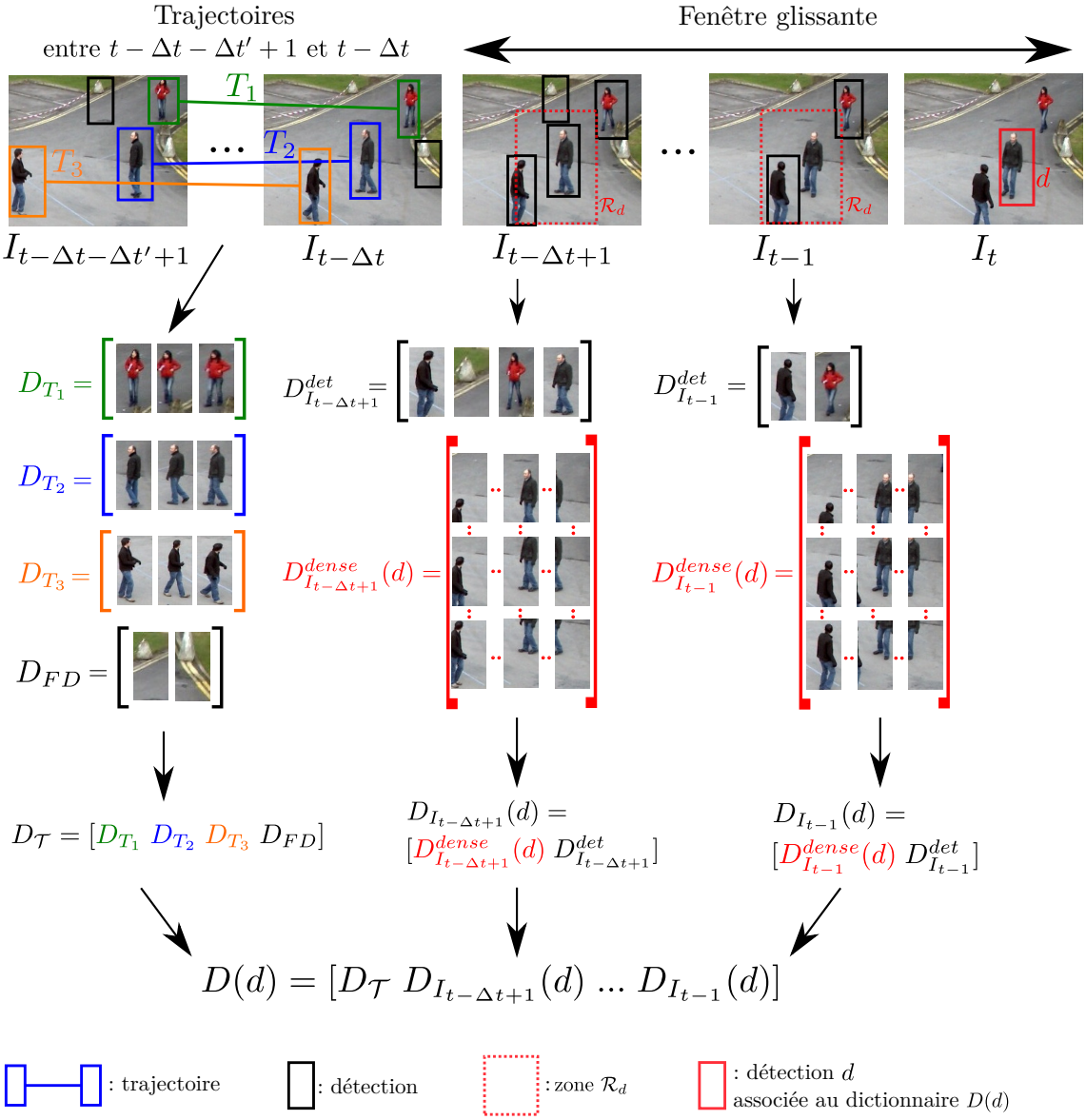


FIGURE V.1 – Exemple de dictionnaire $D(d)$ considéré, en détaillant les sous-dictionnaires pour chaque image de la fenêtre glissante et pour les trajectoires estimées au-delà de cette fenêtre.

Ces sous-dictionnaires sont alors utilisés pour définir les groupes liés à la norme $l_{\infty,1}$ pondérée, de manière similaire à ce qui est fait dans le chapitre précédent en sous-section IV.3.2. Un exemple de dictionnaire $D(d)$ est donné en figure V.1, en précisant les éléments inclus dans les sous-dictionnaires. Nous précisons plus formellement ces sous-dictionnaires dans ce qui suit.

Sous-dictionnaires utilisés

Nous considérons ici une détection d de la dernière image considérée I_t, dont la boîte associée est notée x_d. Une zone de recherche dans le repère image $\mathcal{R}_d$ est de plus considérée, qui peut être supposée quelconque à la seule condition d'être rectangulaire ($\mathcal{R}_d$ peut typiquement inclure toute l'image ou être une zone plus limitée autour de la position de x_d). Étant donnée une image I de la fenêtre glissante,

on considère la sous-image $\mathcal{R}_d(I)$ obtenue en limitant l'image I à la zone $\mathcal{R}_d$. On note alors $\mathcal{B}_d(\mathcal{R}_d(I))$ l'ensemble des boîtes x, avec une largeur et hauteur identique à la détection d, qui sont incluses dans la sous-image $\mathcal{R}_d(I)$. On définit le sous-dictionnaire dense associé à la détection d et l'image I, noté $D_I^{dense}(d)$, comme le dictionnaire formé à partir des caractéristiques visuelles des boîtes de l'ensemble $\mathcal{B}_d(\mathcal{R}_d(I))$. Ce dictionnaire $D_I^{dense}(d)$ inclut donc les caractéristiques visuelles y_x d'un grand nombre de boîtes de l'image I, de même taille que la détection d.

Pour chaque image I de $\{I_{t-\Delta t+1}, ..., I_{t-1}\}$ on considère aussi, de manière similaire au chapitre IV, un dictionnaire D_I^{det} formé à partir des caractéristiques visuelles de l'ensemble des détections de l'image I. Le sous-dictionnaire lié à l'image I, noté $D_I(d)$, est alors obtenu comme la réunion des dictionnaires $D_I^{dense}(d)$ et D_I^{det} :

$$D_I(d) = [D_I^{dense}(d)\ D_I^{det}]\,. \tag{V.8}$$

Le dictionnaire $D_\mathcal{T}$ est simplement défini à partir de l'ensemble des détections des images $\{I_{t-\Delta t-\Delta t'+1}, ..., I_{t-\Delta t}\}$, c'est-à-dire :

$$D_\mathcal{T} = [D^{det}_{I_{t-\Delta t-\Delta t'+1}}\ ...\ D^{det}_{I_{t-\Delta t}}]\,. \tag{V.9}$$

Le dictionnaire $D_\mathcal{T}$ est aussi lié aux trajectoires au delà de la fenêtre glissante, celui-ci pouvant être défini à partir des dictionnaires spécifiques à chaque trajectoire T, notés D_T. Contrairement au chapitre précédent, chaque dictionnaire D_T n'est pas constitué des N_{tr} dernières détections de la trajectoire T mais des détections de T présentes sur les $\Delta t'$ dernières images au-delà de la fenêtre glissante. On considère aussi un dictionnaire D_{FD} (FD pour fausses détections) qui rassemble les détections des images $\{I_{t-\Delta t-\Delta t'+1}, ..., I_{t-\Delta t}\}$ non associées à une trajectoire. Le dictionnaire $D_\mathcal{T}$ peut alors être défini comme :

$$D_\mathcal{T} = [D_{T_1}\ ...\ D_{T_{n_{traj}}}\ D_{FD}]\,. \tag{V.10}$$

Tous les sous-dictionnaires considérés, qui constituent le dictionnaire $D(d)$, sont illustrés en figure V.1.

Justification des choix effectués

Tout d'abord, prendre en compte les détections non associées à des trajectoires dans le dictionnaire $D_\mathcal{T}$ permet de limiter des erreurs d'association entre les trajectoires et les détections de la fenêtre glissante. En effet, toute détection d de la fenêtre glissante a tendance à être représentée, au sein de sa représentation parcimonieuse α_{y_d}, par des éléments de $D_\mathcal{T}$ du fait de la norme $l_{\infty,1}$. Cela est vrai même si la détection correspond à une fausse détection, et cela peut ainsi favoriser des appariements erronés entre les trajectoires et les détections. Inclure les détections non associées à des trajectoires, via le dictionnaire D_{FD}, permet alors de limiter cet effet en permettant aux fausses détections d'être représentées par des détections similaires qui n'appartiennent à aucune trajectoire. C'est notamment le cas des détections dans des pistes qui n'entraînent pas la création d'une trajectoire confirmée à cause de scores de détection trop faible. C'est pour cette raison qu'il est judicieux de considérer pour $D_\mathcal{T}$ l'ensemble des détections des images $\{I_{t-\Delta t-\Delta t'+1}, ..., I_{t-\Delta t}\}$ au lieu de se limiter aux seules détections des trajectoires.

Détection d à représenter
sans dictionnaires denses

Eléments de $[D_{I_{t-\Delta t+1}}(d)\ ...\ D_{I_{t-1}}(d)]$
participant à la représentation de d

Eléments de $D_{\mathcal{T}}$
participant à la représentation de d

FIGURE V.2 – Exemple de représentation parcimonieuse sans dictionnaires denses. Les positions des détections participant à la représentation de d sont affichées en bleu. Les images se lisent de gauche à droite et du haut vers le bas, en allant des images les plus récentes vers les plus anciennes. La cible n'est pas correctement représentée sur toutes les images de la fenêtre glissante, principalement à cause de non détections et de détections imprécises. Représentation calculée sur la vidéo ETH-Bahnhof du *MOTChallenge 2015* en utilisant les détections publiques fournies.

Détection d à représenter
avec dictionnaires denses

Eléments de $[D_{I_{t-\Delta t+1}}(d) \ ... \ D_{I_{t-1}}(d)]$
participant à la représentation de d

Eléments de $D_{\mathcal{T}}$
participant à la représentation de d

FIGURE V.3 – Exemple de représentation parcimonieuse avec dictionnaires denses. Les positions issues de détections sont affichées en bleu, tandis que celles provenant des dictionnaires denses sont affichées en vert. Les images se lisent de gauche à droite et du haut vers le bas, en allant des images les plus récentes vers les plus anciennes. L'emploi de dictionnaires denses permet ici de représenter correctement la cible sur toutes les images de la fenêtre glissante, malgré certaines non détections ou détections imprécises. Représentation calculée sur la vidéo ETH-Bahnhof du *MOTChallenge 2015* en utilisant les détections publiques fournies.

La prise en compte de dictionnaires denses, pour les sous-dictionnaires $D_I(d)$ spécifiques aux images de la fenêtre glissante, permet de compenser une éventuelle non-détection ou une détection imprécise sur cette image. En effet, même si la cible liée à une détection d n'est pas détectée au sein de l'image I, la détection d pourra toujours être représentée par une boîte de $\mathcal{B}_d(\mathcal{R}_d(I))$ liée à la cible.

Plusieurs raisons justifient, au contraire, de ne pas employer de dictionnaires denses au sein du dictionnaire $D_\mathcal{T}$. Tout d'abord, contrairement aux groupes (de la norme $l_{\infty,1}$) liés aux dictionnaires $D_I(d)$, le groupe lié à $D_\mathcal{T}$ fait intervenir des détections sur plusieurs images consécutives. Il est alors beaucoup moins probable qu'une cible, détectée occasionnellement, soit non détectée sur un grand nombre d'images consécutives. Une autre raison est que nous limitons les dictionnaires denses à n'inclure que des caractéristiques visuelles de boîtes de taille identique à la détection reconstruite. Si cela s'avère peu gênant pour les images de la fenêtre glissante qui sont proches temporellement de la détection d, cela devient problématique pour les images plus éloignées temporellement. Si la taille apparente de l'objet dans l'image change significativement, aucun élément des dictionnaires denses ne permet de décrire correctement la cible concernée. Il est tout à fait possible de contourner ce problème en considérant plusieurs échelles possibles pour les boîtes des dictionnaires denses, mais cela se traduit alors par un surcoût significatif en temps de calcul.

Il faut noter ici que les représentations parcimonieuses proposées sont proches des représentations parcimonieuses à convolutions discutées en section V.1, les éléments dans les dictionnaires denses étant obtenus en décalant une boîte de taille fixe dans une image. Néanmoins, les représentations proposées ne peuvent se mettre sous la forme du problème (V.2) pour plusieurs raisons. L'une d'entre elles est que les dictionnaires denses $D_I^{dense}(d)$ utilisés ne peuvent s'écrire comme la réunion de dictionnaires $D_{e_1}...D_{e_m}$, où chaque dictionnaire D_{e_i} inclut tous les décalages d'un élément e_i (il est possible de simplement observer qu'un dictionnaire D_{e_i} est composé d'exactement autant d'éléments que la dimension de e_i, ce qui n'est pas nécessairement le cas pour les dictionnaires $D_I^{dense}(d)$ dont le nombre d'éléments dépend de la zone $\mathcal{R}_d$ indépendamment de la dimension de ces éléments).

Un exemple de représentations parcimonieuses avec et sans emploi des dictionnaires denses est donné en figure V.2 et en figure V.3.

V.2.2 Modèle d'apparence proposé

Bien que le dictionnaire $D(d)$ proposé précédemment permette de reconstruire la détection d correctement, y compris pour les images sur lesquelles la cible associée est non détectée, le modèle d'apparence défini au chapitre IV ne peut être directement réutilisé. En effet, si tout élément e du dictionnaire était auparavant associé à une détection, les éléments du dictionnaire $D(d)$ sont maintenant associés à une position, plus précisément une boîte englobante, dans une image. Certains éléments sont toujours associés à des détections (les éléments des dictionnaires $D_\mathcal{T}$ et de D_I^{det} pour $I \in \{I_{t-\Delta t+1}, ..., I_{t-1}\}$). Néanmoins, on peut considérer que tout élément e du dictionnaire est associé à une boîte x_e. Il suffit, si besoin, de considérer la boîte de la détection éventuellement associée à l'élément e. Le modèle d'apparence $App(C)$ doit donc être adapté pour ne faire intervenir que les boîtes x_e des éléments participant au sein des représentations parcimonieuses.

Pour rappel, le modèle d'apparence employé au chapitre IV est défini, pour toute

configuration C, par :

$$App(C) = \sum_{\tau \in C} \sum_{d \in \tau} ||y_d - D_\tau \alpha^\tau_{y_d}||_2 \,, \tag{V.11}$$

où $||y_d - D_\tau \alpha^\tau_{y_d}||_2$ est l'erreur de reconstruction résiduelle de la détection d par rapport à sa piste τ. Cette erreur de reconstruction résiduelle est définie en considérant le vecteur $\alpha^\tau_{y_d}$ obtenu à partir de la représentation parcimonieuse α_{y_d} en se limitant aux coefficients liés à des éléments de D associés à la piste τ. Lorsque les éléments du dictionnaire sont associés à des détections, les éléments de D associés à la piste τ sont simplement ceux associés à des détections incluses dans cette piste τ. Comment déterminer ces éléments dans le cas actuel, où les éléments du dictionnaire ne sont pas associés à des détections mais à des positions dans les images ?

On note $x_\tau(t)$ la boîte englobante estimée pour la piste τ à l'instant t, boîte estimée à partir des détections constituant la piste τ. L'approche la plus simple est de considérer pour x_τ une interpolation linéaire à partir des détections de τ, ce qui est proposé au chapitre précédent. Les éléments du dictionnaire $D(d)$ associés à une piste τ sont alors les éléments e vérifiant :

$$IOU(x_\tau(t(x_e)), x_e) \geq 0.5 \,, \tag{V.12}$$

où $t(x_e)$ est l'instant correspondant à la boîte x_e, et $x_\tau(t(x_e))$ est donc la boîte estimée pour la piste τ au même instant que x_e. Les éléments associés à τ sont donc ceux dont la boîte est suffisamment proche de la boîte estimée pour τ au même instant, avec un critère de type IOU (*Intersection Over Union*).

Le dictionnaire D_τ associé à la piste τ peut alors être défini à partir des éléments e qui vérifient le critère précédent. On définit $\alpha^\tau_{y_d}$ comme la restriction de α_{y_d} aux coefficients liés à D_τ. Le modèle d'apparence suit alors la même formulation que proposé dans l'équation (V.11), et favorise les configurations C qui minimisent les erreurs de reconstruction résiduelle $||y_d - D_\tau \alpha^\tau_{y_d}||_2$ pour l'ensemble des détections incluses dans la configuration. Les configurations favorisées sont ainsi celles pour lesquelles, pour chaque piste τ, les positions estimées x_τ sont proches des positions des éléments qui reconstruisent les détections de la piste τ.

Le fonctionnement de ce nouveau modèle d'apparence est illustré en figure V.4. Sur cette figure, la piste τ considérée est pénalisée à l'image I_{t-1} car la position $x_\tau(t-1)$ est trop éloignée (en distance IOU) des positions (en noir) représentant la détection d sur cette image. Une piste τ', identique à τ mais qui inclurait la détection de la personne au centre de l'image I_{t-1} à la place de la détection de la personne en bas de cette image, serait alors mieux considérée dans le terme d'apparence App_{Dense}. L'erreur de reconstruction résiduelle $||y_d - D_\tau \alpha^\tau_{y_d}||_2$ serait en effet plus faible dans ce cas de figure.

En pratique, comme fait au chapitre précédent, nous considérons directement les coefficients de $\alpha^\tau_{y_d}$ afin d'accélérer l'association de données par MCMCDA. Le modèle d'apparence, noté $App_{Dense}(C)$, s'écrit donc au final :

$$App_{Dense}(C) = \sum_{\tau \in C} \sum_{d \in \tau} [1 - \sum_i |\alpha^\tau_{y_d}(i)|] \,. \tag{V.13}$$

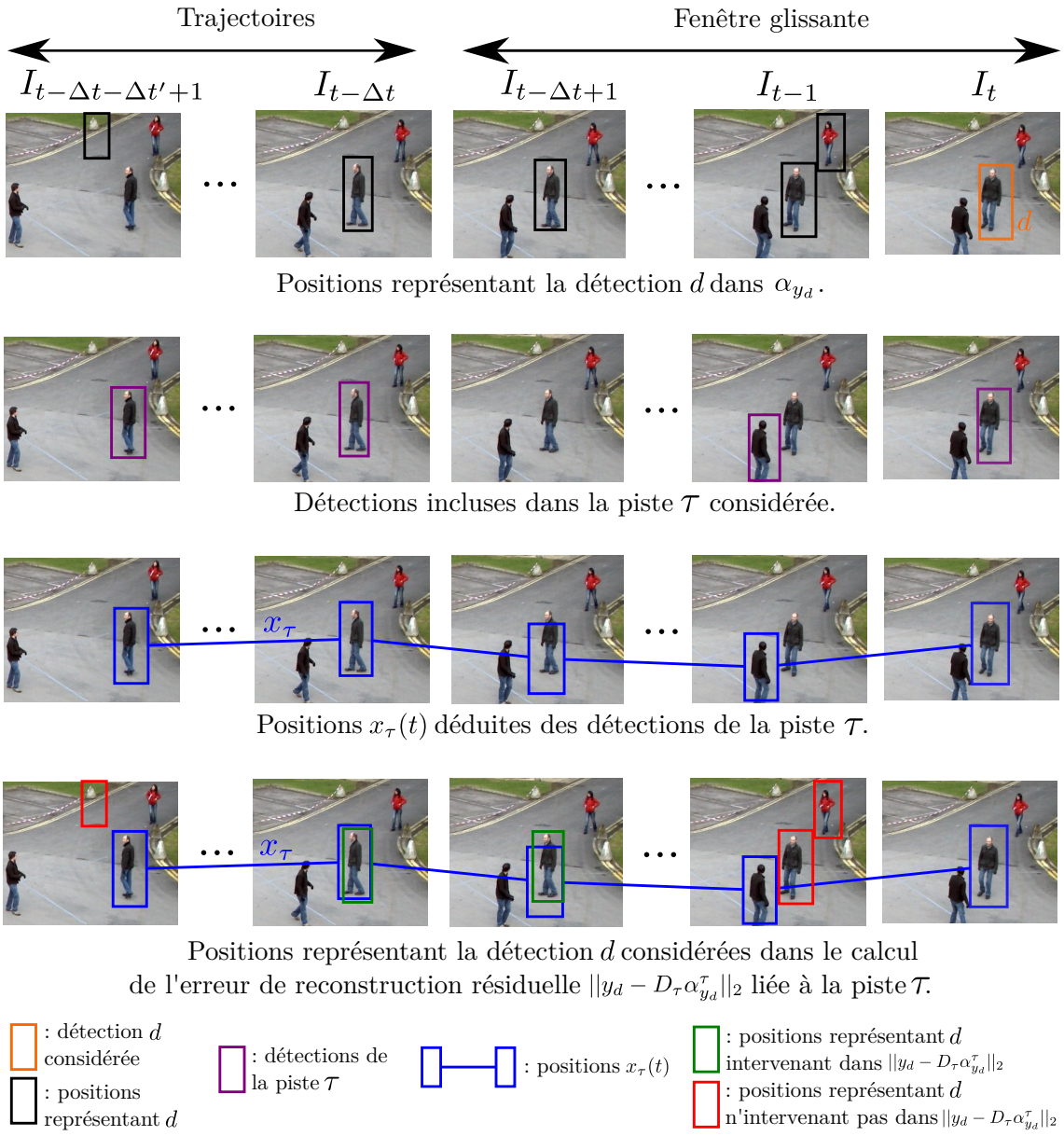


FIGURE V.4 – Fonctionnement du modèle d'apparence $App_{Dense}(C)$ proposé, qui prend en compte les positions déterminées par les représentations parcimonieuses α_{y_d} et les positions x_τ des pistes.

V.2.3 Adaptation des méthodes d'optimisation

Les dictionnaires employés dans ce chapitre ayant été précisés, ainsi que la façon dont les représentations parcimonieuses associées pouvaient être utilisées dans un nouveau modèle d'apparence, il reste à savoir comment ces nouvelles représentations parcimonieuses peuvent être efficacement calculées. On cherche donc ici à résoudre, pour chaque détection d, le problème :

$$\min_\alpha \frac{1}{2}||y_d - D(d)\alpha||_2^2 + \lambda||\alpha||_{\infty,1}^w \,, \tag{V.14}$$

où $D(d)$ est le dictionnaire défini en sous-section V.2.1. La norme $l_{\infty,1}$ pondérée s'écrit :

$$||\alpha||_{\infty,1}^w = \max_{i=1\ldots\Delta t} w_i||\alpha|_{G_i}||_1 \,, \tag{V.15}$$

données : D, y
$\mathcal{A} = \varnothing$, $\alpha_{\mathcal{A}} = 0$;
répéter
  $\mathcal{S} = \{$au plus n_{sel} indices i non inclus dans $\mathcal{A}$ qui maximisent $|e_i^{\mathsf{T}}(D\widetilde{\alpha_{\mathcal{A}}} - y)|$ en se limitant à sélectionner au plus un indice par groupe $G_k\}$;
  Utilisant $\alpha_{\mathcal{A}}$ comme position initiale, trouver la solution optimale $\alpha_{\mathcal{A}\cup\mathcal{S}}$ du problème $\min_\alpha \ \frac{1}{2}||y - D_{\mathcal{A}\cup\mathcal{S}}\alpha||_2^2 + \lambda||\widetilde{\alpha}||_{\infty,1}^w$;
  $\mathcal{A} = \mathcal{A} \cup \mathcal{S}$;
jusqu'à $||D^{\mathsf{T}}(D\widetilde{\alpha_{\mathcal{A}}} - y)||_{1,\infty}^{1/w} \leq \lambda$;
retourner $\widetilde{\alpha_{\mathcal{A}}}$;

Algorithme V.1 : Méta-algorithme avec ensembles actifs pour le calcul de représentations parcimonieuses induites par une norme $l_{\infty,1}$ pondérée.

et les groupes $(G_i)_{i=1...\Delta t}$ sont définis de façon similaire au choix du chapitre IV. Pour $i = 1...\Delta t - 1$, G_i est constitué des indices des éléments e du dictionnaire $D(d)$ correspondant aux éléments du sous-dictionnaire $D_{I_{t-i}}(d)$ tandis que $G_{\Delta t}$ correspond aux indices liés aux éléments e du sous-dictionnaire $D_{\mathcal{T}}$. Les poids $(w_i)_i$ sont identiques à ceux du chapitre IV, c'est-à-dire $w_{\Delta t} = \frac{1}{\Delta t - 1}$ et $w_i = 1$ pour $i < \Delta t$.

Limites des méthodes d'optimisation précédentes

La première approche qui peut être envisagée consiste à employer la méthode d'optimisation par ensembles actifs proposée au chapitre précédent, sans aucune modification, indiquée par l'algorithme V.1. Cependant, la taille des dictionnaires impliqués ici devient alors problématique. Pour des zones de recherches $\mathcal{R}_d$ de taille correcte, sans forcément considérer l'ensemble de l'image, le sous-dictionnaire dense $D_I^{dense}(d)$ pour l'image I est rapidement constitué de plusieurs dizaines de milliers d'éléments. Cela signifie que le dictionnaire complet $D(d)$ inclut plusieurs centaines de milliers d'éléments, un nombre bien plus important que ce qui est considéré dans les approches précédentes. Par exemple, si on considère pour caractéristiques visuelles les valeurs d'intensité des boîtes redimensionnées à 32×32 pixels, et une zone de recherche $\mathcal{R}_d$ avec une largeur et hauteur toutes deux cinq fois supérieures aux dimensions de la détection d, chaque sous-dictionnaire dense $D_I^{dense}(d)$ inclut dans ce cas $(5 \times 32 - 32 + 1)^2 = 16641$ éléments.

Le principal problème dans l'algorithme V.1 est l'étape de sélection des variables $\mathcal{S}$ à rajouter à l'ensemble actif $\mathcal{A}$. Cette étape nécessite de sélectionner au plus n_{sel} indices i non inclus dans $\mathcal{A}$ qui maximisent $|e_i^{\mathsf{T}}(D\widetilde{\alpha_{\mathcal{A}}} - y)|$, en se limitant à sélectionner au plus un indice par groupe G_k. En pratique, cela requiert d'évaluer la valeur $|e^{\mathsf{T}}(D\widetilde{\alpha_{\mathcal{A}}} - y)|$ pour tous les éléments de $D(d)$ qui ne sont pas déjà inclus dans l'ensemble actif $\mathcal{A}$. Evaluer toutes ces valeurs devient alors beaucoup trop coûteux en temps de calcul avec les dictionnaires denses employés. En réalité, la construction même du dictionnaire $D(d)$, en calculant toutes les caractéristiques visuelles des boîtes considérées, devient problématique.

Corrélation croisée normalisée

Il est néanmoins possible d'exploiter la structure particulière des éléments des dictionnaires denses $D_I^{dense}(d)$, qui correspondent aux caractéristiques visuelles d'une

boîte de taille fixe déplacée dans une image ou sous-image $\mathcal{R}_d(I)$. Les caractéristiques visuelles y_x considérées sont désormais limitées aux valeurs d'intensité des boîtes x, redimensionnées à une taille de $n_x \times n_y$ pixels. On suppose dans ce qui suit que ces intensités n'utilisent qu'un seul canal (en niveaux de gris), mais l'approche décrite s'étend sans difficulté au cas de canaux multiples et en particulier des valeurs d'intensité RGB. De plus, de manière classique pour les méthodes de représentation parcimonieuse, on suppose ces caractéristiques y_x normalisées (en norme l_2). On a donc pour tout élément e du dictionnaire $D(d)$, associé à une position x_e, $e = y_{x_e}$. Étant donné un élément e du dictionnaire $D(d)$, que représente concrètement la valeur $|e^\mathsf{T}(D\widetilde{\alpha_\mathcal{A}} - y)|$ qui est utilisée pour déterminer les éléments à ajouter à l'ensemble actif $\mathcal{A}$?

La valeur $e^\mathsf{T}(D\widetilde{\alpha_\mathcal{A}} - y)$ peut être vue comme une corrélation entre les valeurs d'intensité normalisées de la position x_e et un filtre ϕ dont les coefficients correspondent à $D\widetilde{\alpha_\mathcal{A}} - y$. On considère maintenant l'image $J_{\mathcal{R}_d(I)}$ obtenue en redimensionnant la sous-image $\mathcal{R}_d(I)$ de manière à ce que la détection d soit de taille $n_x \times n_y$ dans cette image. Appliquer une corrélation croisée normalisée (*Normalized Cross-Correlation*) en appliquant le filtre ϕ retourne alors la valeur :

$$\frac{i_x^\mathsf{T}\phi}{||i_x||_2||\phi||_2} = \frac{1}{||\phi||_2} y_x^\mathsf{T}(D\widetilde{\alpha_\mathcal{A}} - y)\,, \tag{V.16}$$

pour toute position x de l'image $J_{\mathcal{R}_d(I)}$, en notant i_x les valeurs d'intensité (non normalisées) associées à la position x en considérant une boîte de la taille du filtre ϕ. On remarque ainsi qu'évaluer toutes les valeurs $e^\mathsf{T}(D\widetilde{\alpha_\mathcal{A}} - y)$ pour les éléments e du dictionnaire dense $D_I^{dense}(d)$ se ramène au calcul d'une corrélation croisée normalisée sur l'image $J_{\mathcal{R}_d(I)}$ avec un filtre ϕ approprié (et sous réserve de corriger la normalisation $\frac{1}{||\phi||_2}$ dans les résultats finaux). Cette observation est particulièrement intéressante, car il existe des algorithmes efficaces pour calculer des corrélations croisées normalisées sur une image.

Algorithme rapide de corrélation croisée normalisée

Calculer des corrélations croisées normalisées dans une image, en déplaçant un filtre, est une opération très courante en Vision par Ordinateur et surtout en traitement d'image. De ce fait, des approches efficaces ont été proposées pour accélérer le temps de calcul d'une telle opération. Nous expliquons ici l'approche proposée dans [72], qui exploite des transformées de Fourier rapides et des images intégrales [27, 119].

On considère ici une image I, de dimension $N_x \times N_y$, et un filtre ϕ de dimension $M_x \times M_y$. On cherche à calculer, pour tout couple (x, y), la valeur :

$$NCC(x,y) = \frac{\sum_{x',y'} \phi(x',y') I(x+x', y+y')}{\sqrt{\sum_{x',y'} \phi(x',y')^2} \sqrt{\sum_{x',y'} I(x+x', y+y')^2}}\,. \tag{V.17}$$

Cette valeur peut s'écrire sous la forme :

$$NCC(x,y) = \frac{1}{||\phi||_2} \frac{CC(x,y)}{N(x,y)}\,, \tag{V.18}$$

où $CC(x,y) = \sum_{x',y'} \phi(x',y')I(x+x',y+y')$ est la corrélation non normalisée en (x,y) et

$$N(x,y) = \sqrt{\sum_{x'=0..M_x-1, y'=0..M_y-1} I(x+x',y+y')^2} \tag{V.19}$$

est la norme liée à la position (x,y). Ainsi, estimer les corrélations croisées normalisées $NCC(x,y)$ revient à estimer les corrélations croisées non normalisées $CC(x,y)$ et la norme $N(x,y)$.

Le calcul des corrélations $CC(x,y)$ peut alors s'effectuer efficacement dans le domaine fréquentiel avec des FFT [1]. Chaque corrélation $CC(x,y)$ peut en effet être vue comme une convolution avec un filtre ϕ' déduit du filtre ϕ [2]. On a alors :

$$CC(x,y) = \mathcal{F}^{-1}(\mathcal{F}(I)\mathcal{F}(\phi'))(x,y)\,, \tag{V.20}$$

où $\mathcal{F}(I)$ et $\mathcal{F}(\phi')$ sont les transformées de Fourier de l'image I et du filtre ϕ' (multipliées ici élément par élément), et $\mathcal{F}^{-1}$ est la transformée de Fourier inverse. Cette formulation est souvent plus rapide qu'un calcul naïf, car la complexité obtenue en passant dans le domaine fréquentiel est en $O(N_xN_ylog(N_xNy))$ tandis que le calcul naïf est en $O(N_xN_yM_xM_y)$.

Il reste à calculer la norme $N(x,y)$ liée à chaque position (x,y). Un calcul naïf mène ici aussi à une complexité en $O(N_xN_yM_xM_y)$. Il est néanmoins possible de réaliser cette étape en $O(N_xN_y)$ à l'aide d'images intégrales. Le principe est de calculer l'image intégrale $Int(I^2)$ du carré des intensités de l'image originale, définie par :

$$Int(I^2)(x,y) = \sum_{x'\leq x, y'\leq y} I(x,y)^2\,. \tag{V.21}$$

Cette image intégrale peut être calculée efficacement en $O(N_xN_y)$. Une fois le calcul de cette image réalisé, chaque valeur $N(x,y)$ se calcule en temps constant via l'expression [3] :

$$\begin{aligned} N(x,y)^2 = {} & Int(I^2)(x+M_x-1, y+M_y-1) + Int(I^2)(x-1,y-1) \\ & - Int(I^2)(x-1, y+M_y-1) - Int(I^2)(x+M_x-1, y-1)\,. \end{aligned} \tag{V.22}$$

Toutes les valeurs $N(x,y)$ peuvent ainsi être calculées en $O(N_xN_y)$, ce qui permet de déterminer toutes les valeurs $NCC(x,y)$ en $O(N_xN_ylog(N_xNy))$.

Nous expliquons maintenant comment cet algorithme rapide de corrélation croisée normalisée peut être exploité pour notre optimisation des représentations parcimonieuses.

Ensembles actifs et corrélations croisées normalisées

Nous avons vu précédemment que l'étape problématique dans l'algorithme V.1 concernait la sélection des variables $\mathcal{S}$ à ajouter à l'ensemble actif $\mathcal{A}$. Cette étape est coûteuse du fait du nombre très important d'éléments contenus dans le dictionnaire $D(d)$. Cependant, le calcul des valeurs $|e^\mathsf{T}(D\widetilde{\alpha_\mathcal{A}} - y)|$, pour tout élément e du

1. Plus précisément avec des DFT (*Discrete Fourier Transform*).

2. Le filtre ϕ' considéré est notamment déduit de ϕ en étant redimensionné aux mêmes dimensions que l'image I par ajout de valeurs nulles, ceci étant nécessaire pour obtenir des transformées de Fourier discrètes $\mathcal{F}(I)$ et $\mathcal{F}(\phi')$ de mêmes dimensions.

3. Avec pour convention $Int(I^2)(x,-1) = Int(I^2)(-1,y) = 0$ pour toutes valeurs de x et y.

données : y_d, $D(d)$, I_{t-1},..., $I_{t-\Delta t+1}$
$y = y_d$, $D = D(d)$, $\mathcal{A} = \varnothing$, $\alpha_{\mathcal{A}} = 0$;
pour $i = t - \Delta t + 1 \; ... \; t - 1$ **faire**
  Redimensionnement de l'image $\mathcal{R}_d(I_i)$ en $J_i = J_{\mathcal{R}_d(I_i)}$;
  Calcul de l'image intégrale $Int({J_i}^2)$;
  Calcul des valeurs $N_i(x, y)$ à partir de l'image intégrale $Int({J_i}^2)$ en suivant l'équation (V.22);
  Calcul de $\mathcal{F}(J_i)$;
fin
répéter
  Calculer le filtre ϕ correspondant à $(D\widetilde{\alpha_{\mathcal{A}}} - y)$, le filtre ϕ' associé et $\mathcal{F}(\phi')$;
  $\mathcal{S} = \; SELECTION(D, y, \mathcal{A}, \alpha_{\mathcal{A}}, \phi', \mathcal{F}(\phi'), (\mathcal{F}(J_i))_i, (N_i)_i)$ comme décrit par l'algorithme V.3;
  Utilisant $\alpha_{\mathcal{A}}$ comme position initiale, trouver la solution optimale $\alpha_{\mathcal{A}\cup\mathcal{S}}$ du problème $\min_\alpha \; \frac{1}{2}||y - D_{\mathcal{A}\cup\mathcal{S}}\alpha||_2^2 + \lambda||\widetilde{\alpha}||_{\infty,1}^w$;
  $\mathcal{A} = \mathcal{A} \cup \mathcal{S}$;
jusqu'à $||D^\mathsf{T}(D\widetilde{\alpha_{\mathcal{A}}} - y)||_{1,\infty}^{1/w} \leq \lambda$;
retourner $\widetilde{\alpha_{\mathcal{A}}}$;

Algorithme V.2 : Méta-algorithme avec ensembles actifs pour le calcul de représentations parcimonieuses en norme $l_{\infty,1}$ pondérée avec dictionnaires denses.

dictionnaire, se ramène pour les sous-dictionnaires denses $D_I^{dense}(d)$ au calcul d'une corrélation croisée normalisée avec une image et un filtre appropriés. Avec l'algorithme rapide de corrélation croisée normalisée, les valeurs $|e^\mathsf{T}(D\widetilde{\alpha_{\mathcal{A}}} - y)|$ peuvent alors être efficacement calculées pour les sous-dictionnaires denses $D_I^{dense}(d)$. Cela permet de déterminer rapidement l'ensemble $\mathcal{S}$ des variables à rajouter à l'ensemble actif $\mathcal{A}$.

Des considérations assez simples permettent encore de gagner en temps de calcul. En effet, déterminer les valeurs $|e^\mathsf{T}(D\widetilde{\alpha_{\mathcal{A}}} - y)|$ pour les éléments de chaque dictionnaire dense $D_I^{dense}(d)$ est réalisé en suivant les étapes suivantes :

(i) Redimensionnement de $\mathcal{R}_d(I)$ en l'image $J_{\mathcal{R}_d(I)}$.

(ii) Calcul de l'image intégrale $Int({J_{\mathcal{R}_d(I)}}^2)$ et des valeurs $N(x, y)$.

(iii) Calcul de $\mathcal{F}(J_{\mathcal{R}_d(I)})$.

(iv) Calcul de $\mathcal{F}(\phi')$, avec ϕ' filtre associé aux valeurs $D\widetilde{\alpha_{\mathcal{A}}} - y$.

(v) Calcul de $\mathcal{F}^{-1}(\mathcal{F}(J_{\mathcal{R}_d(I)})\mathcal{F}(\phi'))$, donnant les valeurs $CC(x, y)$.

(vi) Calcul des valeurs $NCC(x, y)$ à partir des valeurs $CC(x, y)$, $N(x, y)$ et $||\phi||_2$.

(vii) Correction de la normalisation non désirée sur ϕ, en multipliant les valeurs $NCC(x, y)$ par $||\phi||_2$.

Plusieurs de ces calculs peuvent être partagés entre plusieurs étapes de sélection des éléments $\mathcal{S}$. Les calculs de $J_{\mathcal{R}_d(I)}$, $Int({J_{\mathcal{R}_d(I)}}^2)$, $N(x, y)$ et $\mathcal{F}(J_{\mathcal{R}_d(I)})$ peuvent être réalisés une unique fois au début de l'algorithme V.1, pour toutes les images I de la fenêtre glissante. Le calcul de $\mathcal{F}(\phi')$ peut être effectué une unique fois dans chaque passage de la boucle principale de l'algorithme V.1. Le calcul de $||\phi||_2$ n'est pas nécessaire, il est possible de l'éviter en fusionnant les étapes (vi) et (vii). Le partage de ces différents calculs est précisé dans l'algorithme V.2 et l'algorithme V.3.

données : D, y, $\mathcal{A}$, $\alpha_{\mathcal{A}}$, ϕ', $\mathcal{F}(\phi')$, $(\mathcal{F}(J_i))_i$, $(N_i)_i$
pour $i = 1 \ldots \Delta t - 1$ **faire**
Calcul des valeurs $CC_i(x,y) = \mathcal{F}^{-1}(\mathcal{F}(J_i)\mathcal{F}(\phi'))(x,y)$;
Calcul des valeurs $NCC_i(x,y) = \frac{1}{||\phi'||_2}\frac{CC_i(x,y)}{N_i(x,y)}$;
Déterminer $s_i^{dense} = \arg\max_{s\in G_i,\, s\notin\mathcal{A},\, e_s\in D_{I_{t-i}}^{dense}(d)} |e_s^{\mathsf{T}}(D\widetilde{\alpha_{\mathcal{A}}} - y)|$ à partir des valeurs $NCC_i(x,y)$;
Déterminer $s_i^{det} = \arg\max_{s\in G_i,\, s\notin\mathcal{A},\, e_s\in D_{I_{t-i}}^{det}} |e_s^{\mathsf{T}}(D\widetilde{\alpha_{\mathcal{A}}} - y)|$;
$s_i = \arg\max_{s\in G_i,\, s\notin\mathcal{A}} |e_s^{\mathsf{T}}(D\widetilde{\alpha_{\mathcal{A}}} - y)| = \arg\max_{s\in\{s_i^{dense}, s_i^{det}\}} |e_s^{\mathsf{T}}(D\widetilde{\alpha_{\mathcal{A}}} - y)|$;
fin
Déterminer $s_{\Delta t} = \arg\max_{s\in G_{\Delta t},\, s\notin\mathcal{A}} |e_s^{\mathsf{T}}(D\widetilde{\alpha_{\mathcal{A}}} - y)|$;
$\mathcal{S}$ = au plus n_{sel} indices s parmi $(s_i)_{i=1..\Delta t}$ qui maximisent $|e_s^{\mathsf{T}}(D\widetilde{\alpha_{\mathcal{A}}} - y)|$;
retourner $\mathcal{S}$;

Algorithme V.3 : Algorithme *SELECTION* pour déterminer les indices à ajouter à l'ensemble actif $\mathcal{A}$ pour le calcul de représentations parcimonieuses en norme $l_{\infty,1}$ pondérée avec dictionnaires denses.

L'algorithme d'optimisation ainsi obtenu permet de calculer efficacement les représentations parcimonieuses sur des dictionnaires comportant un grand nombre d'éléments, car les dictionnaires complets $D(d)$ considérés ne sont alors jamais explicitement construits par l'algorithme. Seuls les dictionnaires $D_{\mathcal{A}}$ sont construits explicitement, mais comportent un nombre beaucoup plus raisonnable d'éléments.

V.3 Système de suivi employé

Cette section précise le système de suivi au sein duquel sont employées les représentations parcimonieuses avec dictionnaires denses. Nous utilisons le système de suivi proposé pour l'approche **LINF1** [37] dont nous rappelons rapidement le principe, et nous nous focalisons sur les quelques modifications apportées par rapport au chapitre précédent.

V.3.1 Principe général

Le principe général du système de suivi employé, similaire à celui du chapitre précédent, est rappelé très succinctement ici. Pour davantage de détails, le lecteur peut se référer à la section IV.2.

Le système de suivi utilisé se base sur une fenêtre glissante constituée des Δt dernières images $\{I_{t-\Delta t+1}, ..., I_t\}$. Les étapes suivantes sont alors effectuées à chaque nouvel instant t :

(i) Calcul des représentations parcimonieuses des détections Det_{I_t} de la dernière image I_t. Ces représentations parcimonieuses sont définies, pour chaque détection d, à partir du dictionnaire $D(d)$ proposé en sous-section V.2.1. Cette étape nécessite donc de résoudre pour chaque détection le problème associé à l'équation (V.6).

(ii) Association des données sur la fenêtre glissante en utilisant une approche de type MCMCDA. L'énergie employée est ici identique à celle proposée en sous-section IV.2.2, à l'exception du terme $App(C)$ qui est remplacé par un terme

$App_{Dense}(C)$ adapté aux dictionnaires denses en utilisant la formulation proposée en sous-section V.2.2.

(iii) Gestion et prolongement des trajectoires pour l'image $I_{t-\Delta t+1}$, en suivant la configuration optimale C^* déterminée par l'association de données.

Le système de suivi utilisé se différencie surtout de l'approche précédente, **LINF1**, du fait de l'emploi de dictionnaires $D(d)$ différents pour le calcul des représentations parcimonieuses et de l'utilisation d'un modèle d'apparence adapté $App_{Dense}(C)$. Les autres modifications apportées à ce système, moins significatives, sont décrites ci-après.

V.3.2 Lissage des pistes

L'énergie globale $E(C)$ considérée est constituée de quatre termes :

$$E(C) = \theta_{Ob}Ob(C) + \theta_{App}App_{Dense}(C) + \theta_{Mot}Mot(C) + \theta_{Int}Int(C) \qquad \text{(V.23)}$$

où $Ob(C)$, $Mot(C)$ et $Int(C)$ sont des termes définis en sous-section IV.2.2 tandis que le terme $App_{Dense}(C)$ est défini en sous-section V.2.2. Tous ces termes, à l'exception de $Ob(C)$, font intervenir, pour chaque piste τ de la configuration C, les boîtes $x_\tau(t)$ estimées pour chaque instant temporel de la piste τ. Contrairement à l'approche **LINF1**, où le terme $App(C)$ ne faisait pas intervenir ces estimations de boîtes $x_\tau(t)$, le terme $App_{Dense}(C)$ est concerné lui-aussi par ces estimations. Une estimation correcte des boîtes $x_\tau(t)$ pour chaque piste τ est donc cruciale ici.

Estimer les boîtes $x_\tau(t)$ à partir des boîtes initiales des détections de la piste τ revient à effectuer une opération de lissage. Cette étape est faite de façon très naïve dans l'approche **LINF1** où une simple interpolation linéaire entre les détections est effectuée. Est-il possible de réaliser un lissage plus pertinent ? Et dans quelle mesure le lissage de ces boîtes influence les performances de la méthode de suivi ?

Dans notre système de suivi, les boîtes $x_\tau(t)$ doivent être déterminées à chaque nouvelle proposition de piste τ dans l'optimisation par MCMCDA. Cela signifie que le nombre de lissages à effectuer est du même ordre de grandeur que le nombre de mouvements considérés dans l'échantillonnage de type MCMC, c'est-à-dire plusieurs milliers par image. L'approche de lissage utilisée pour déterminer les boîtes $x_\tau(t)$ doit donc avoir un coût CPU limité pour éviter que cette étape ne devienne trop pénalisante en temps de calcul.

Nous proposons ici une méthode de lissage assez basique, mais qui permet néanmoins d'estimer les boîtes $x_\tau(t)$ de manière plus précise qu'une interpolation linéaire tout en étant suffisamment rapide pour être appliquée au sein de l'optimisation par MCMCDA. L'approche proposée s'inspire de la méthode RDP (*Ramer-Douglas-Peucker*) qui est une méthode de simplification de polygones [34, 104]. La méthode RDP cherche à approximer une série de points $p_1, ..., p_{n_p}$ par un ensemble plus petit de segments. L'algorithme RDP consiste à considérer le segment $[p_1, p_{n_p}]$ et examiner si l'ensemble des points $p_1, ..., p_{n_p}$ sont correctement approchés par ce segment (en utilisant un critère basé sur une distance limite entre chaque point p et le segment). L'algorithme se termine si c'est le cas, sinon le point p_i le plus éloigné du segment $[p_1, p_{n_p}]$ est déterminé et l'algorithme est appelé récursivement sur les séries de points $p_1, ..., p_i$ et $p_i, ..., p_{n_p}$.

Nous adaptons ici cet algorithme pour lisser les boîtes $x_1, ..., x_{n_x}$ des détections d'une piste τ, en considérant comme distance une fonction IOU. De plus, pour

données : $(x_i)_{i=1..n_x}$, th_{RDP} (avec $th_{RDP} > 0$)
Déterminer la séquence de boîtes $(x_{reg}(t))$ par régression linéaire sur les boîtes $(x_i)_{i=1..n_x}$;
$k^* = \arg\max_{i=1..n_x} IOU(x_i, x_{reg}(t(x_i)))$;
si $IOU(x_{k^*}, x_{reg}(t(x_{k^*}))) \leq th_{RDP}$ **alors**
  retourner $(x_{reg}(t))$;
sinon
  Déterminer la séquence de boîtes $(x_{lin}(t))$ par interpolation linéaire entre les deux boîtes les plus distantes, x_1 et x_{n_x};
  $i^* = \arg\max_{i=1..n_x} IOU(x_i, x_{lin}(t(x_i)))$;
  retourner $RDP((x_i)_{i=1..i^*}) \cup RDP((x_i)_{i=i^*..n_x})$;
fin

Algorithme V.4 : Algorithme de type RDP pour lisser les boîtes des détections des pistes (appelé simplement RDP pour simplifier les notations).

le critère d'arrêt, plutôt que de considérer la distance maximale entre le segment $[x_1, x_{n_x}]$ et les boîtes $x_1, ..., x_{n_x}$, nous effectuons tout d'abord une régression linéaire sur toutes les boîtes entre x_1 et x_{n_x} et examinons la distance maximale entre le résultat de cette régression et les boîtes d'origine $x_1, ..., x_{n_x}$. L'algorithme V.4 permet alors d'estimer de façon plus précise les tailles des boîtes $x_\tau(t)$, du fait des régressions linéaires effectuées, tout en s'adaptant à d'éventuels mouvements brusques dans le repère image (provoqués par exemple par le mouvement de la caméra) à l'aide de la segmentation de type RDP effectuée récursivement.

On peut justifier assez facilement la terminaison de cet algorithme. En effet, on peut remarquer que, si le critère d'arrêt n'est pas vérifié, la boîte x_{i^*} choisie pour réaliser la segmentation et l'appel récursif ne pourra plus jamais être choisie comme point de segmentation dans le reste de l'algorithme. Il y a nécessairement un nombre fini d'appels récursifs, et l'algorithme se termine donc dans tous les cas.

Il est possible d'utiliser d'autres méthodes de lissage, en particulier à base de filtres de Kalman [105] ou de filtres à particules [54]. Une analyse détaillée des performances de ces différentes méthodes, en considérant aussi leur applicabilité en termes de temps de calcul, serait à envisager. Une telle étude s'écarte néanmoins de l'objet d'étude de ce chapitre, et nous considérons donc simplement la méthode de lissage proposée par l'algorithme V.4. Cela permet déjà d'étudier l'influence du lissage des pistes sur les résultats de suivi, ce qui est fait en section V.4.

V.3.3 Scores normalisés et endormissement des trajectoires

Les dernières modifications apportées au système de suivi du chapitre IV concernent la normalisation des scores de détection et l'endormissement des trajectoires.

Pour commencer, nous introduisons le calcul au fur et à mesure du suivi de la moyenne μ_s et de l'écart type σ_s des scores de détection donnés par le détecteur d'objets. Puis nous intégrons ces calculs dans le modèle d'observation $Ob(C)$ pour prendre en compte des valeurs normalisées des scores de détection :

$$Ob(C) = -\sum_{\tau \in C} \sum_{d \in \tau} [\alpha_{Ob} + \beta_{Ob} \bar{s_d}] - \sum_{\tau \in C} \sum_{T \in \tau} \gamma_{Ob}\,, \tag{V.24}$$

avec $\bar{s_d} = \frac{s_d - \mu_s}{\sigma_s}$. La moyenne μ_s et l'écart type σ_s sont également utilisés pour définir

le seuil limite s_c. Pour rappel, dans notre système de suivi si une piste τ issue de l'association de données n'est pas associée à une trajectoire et inclut une détection appartenant à la dernière image de la fenêtre glissante $I_{t-\Delta t+1}$, alors la piste τ peut mener à la création d'une nouvelle trajectoire si elle est composée d'au moins N_c détections avec un score de détection moyen supérieur à s_c. Ce seuil s_c est défini ici par :

$$s_c = \begin{cases} \mu_s + \delta_s \sigma_s & si\ \mu_s + \delta_s \sigma_s \in [s_{min}, s_{max}] \\ s_{min} & si\ \mu_s + \delta_s \sigma_s < s_{min} \\ s_{max} & si\ \mu_s + \delta_s \sigma_s > s_{max} \end{cases} . \tag{V.25}$$

Les paramètres s_{min}, s_{max} et δ_s sont alors fixés manuellement. s_{min} et s_{max} peuvent être vus comme des valeurs limites sur les scores de détections, valeurs au-delà desquelles on est certain d'avoir une fausse détection (score en dessous de s_{min}) ou une détection correcte (score au-dessus de s_{max}). Déterminer ainsi le seuil s_c permet de s'adapter à certaines scènes où le score des détections est globalement faible ou au contraire globalement élevé.

Pour finir, les trajectoires peuvent ici être endormies, comme fait auparavant au chapitre III, c'est-à-dire toujours considérées par l'algorithme de suivi mais sans participer aux résultats affichés. Toute trajectoire associée à la suite de l'association de données à une piste τ dont le score de détection moyen est inférieur à $th_{end.}\, s_c$ est considérée endormie. Toute trajectoire endormie est éveillée si elle se retrouve associée à une piste τ dont le score de détection moyen est au-dessus de s_c.

V.4 Évaluations et analyse des résultats

V.4.1 Implémentation et protocole d'évaluation

Implémentation

L'approche proposée dans ce chapitre, appelée **DSR** (*Dense Sparse Representation*) est implémentée en C++ et est exécutée en employant un CPU multi-coeurs (4 coeurs, 8 threads) à 2.7 GHz. Les calculs des représentations des détections de la dernière image considérée sont alors effectués en parallèle. L'algorithme V.2 de calcul d'une représentation parcimonieuse avec dictionnaires denses prend aussi avantage d'une possible parallélisation (en calculant en parallèle les corrélations croisées normalisées de chaque image de la fenêtre glissante). Les calculs des FFT, des IFFT et des images intégrales, utilisés par l'algorithme V.2, sont réalisés en employant la bibliothèque OpenCV [4].

Les représentations à base de dictionnaires denses sont calculées avec l'algorithme V.2. Comme décrit au chapitre III, cet algorithme à base d'ensembles actifs est en pratique utilisé avec un critère d'arrêt qui est ici un nombre limite d'itérations principales d'agrandissement de l'ensemble actif $\mathcal{A}$. L'algorithme V.2 est utilisé avec dix étapes de sélection de l'ensemble actif $\mathcal{A}$ en sélectionnant au plus dix nouveaux indices ($n_{sel} = 10$). Entre chacune de ces étapes, l'optimisation des représentations parcimonieuses sur l'ensemble actif $\mathcal{A}$ est réalisée par l'algorithme FISTA en se limitant à dix itérations. Ces choix se justifient à la suite de l'examen de la vitesse de convergence de cet algorithme pour certaines détections, comme indiqué en figure V.5.

4. URL : *opencv.org*

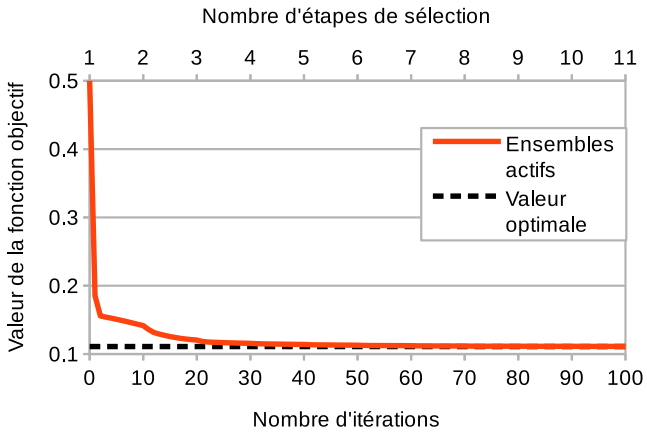


FIGURE V.5 – Évaluation de la convergence de l'optimisation des représentations parcimonieuses à base de dictionnaires denses. Les valeurs moyennes de la fonction objective du problème (V.14) sont indiquées en fonction du nombre d'itérations, pour dix détections prises aléatoirement sur la vidéo PETS S2L2 avec les détections publiques du *MOTChallenge 2015*. L'approche d'optimisation à base d'ensembles actifs proposée permet effectivement d'atteindre une solution globale avec une précision suffisante avec les choix faits en pratique pour appliquer cette méthode. Nous limitons en effet ici le nombre d'étapes d'agrandissement de l'ensemble actif à 10, avec 10 itérations de l'algorithme FISTA, entre chacune de ces étapes, pour résoudre le problème restreint à l'ensemble actif $\mathcal{A}$.

Les caractéristiques visuelles considérées pour décrire les détections, et les positions dans les images, sont simplement les valeurs d'intensité RGB des boîtes, redimensionnées à une taille de 32×32 pixels. Pour chaque détection d, la zone de recherche $\mathcal{R}_d$ considérée est un carré centré sur la détection d et dont la largeur est huit fois celle de d. Le nombre de mouvements utilisé pour l'association de données par MCMCDA est fixé à 10000, et nous considérons uniquement des fenêtres glissantes de dix images pour obtenir un bon compromis entre les performances et le temps de calcul de notre approche. Avec ce choix de paramètres, notre méthode fonctionne sur l'ensemble des vidéos de test de la base de données *2DMOT2015* à 2.6 images par seconde, cette vitesse de fonctionnement pouvant être facilement accélérée en limitant le nombre de mouvements du MCMCDA ou le nombre d'itérations pour l'optimisation des représentations parcimonieuses (mais au prix d'une légère dégradation des performances).

Protocole d'évaluation

Le protocole d'évaluation est proche de celui employé pour la méthode **LINF1** au chapitre précédent. Les bases de données du *MOTChallenge* sont utilisées, en considérant à la fois la version 2015, intitulée *2DMOT2015*, et la version 2016, intitulée *MOT16*. Nous utilisons toujours les métriques fournies par le *MOTChallenge*, consitutées notamment des métriques CLEARMOT [14] et de métriques de qualité des trajectoires comme le pourcentage de trajectoires majoritairement suivies et majoritairement perdues (respectivement MT et ML). Plus de détails sur ces métriques sont donnés en sous-section II.4.2.

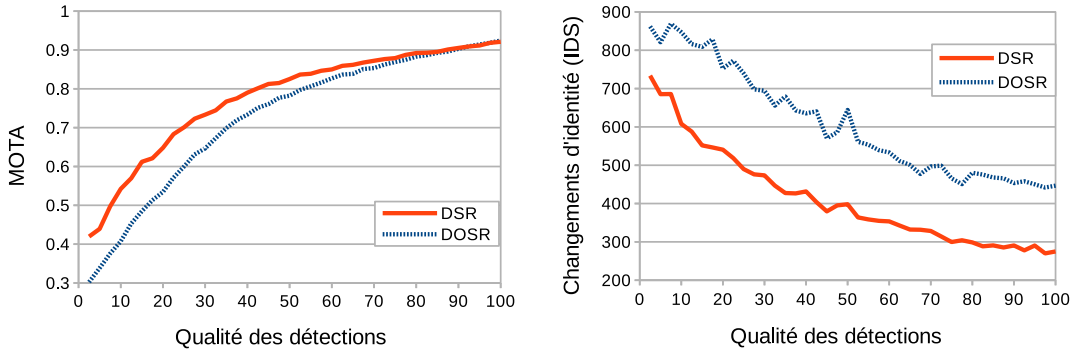


FIGURE V.6 – Résultats en termes de MOTA↑ et de changements d'identité (IDS↓) pour les approches **DSR**, avec dictionnaires denses, et **DOSR**, sans dictionnaires denses. Performances obtenues sur l'ensemble d'entraînement de la base de données *2DMOT2015* avec des jeux de détection simulés à partir des vérités terrain.

L'approche **DSR** proposée dans ce chapitre est comparée à certaines variantes pour évaluer les choix effectués, en particulier en ce qui concerne l'emploi des dictionnaires denses et le lissage proposé pour les pistes. Afin de comparer correctement les variantes proposées, nous utilisons de nouveau une procédure d'hyper-optimisation afin de déterminer le meilleur jeu de paramètres, en termes de MOTA, pour chacune des variantes. Cette procédure d'hyper-optimisation repose sur la méthode SMAC proposée par l'article [52], comme décrit précédemment en sous-section IV.4.1. En particulier, cette procédure d'optimisation teste chaque configuration de paramètres avec cinq graines différentes afin de limiter le caractère stochastique des méthodes proposées.

V.4.2 Comparaison des variantes étudiées

Tout au long de cette sous-section, nous considérons une variante, notée **DOSR** (*Detection Only Sparse Representations*), qui se différencie de l'approche proposée **DSR** en employant un dictionnaire qui n'inclut que des détections. Cette variante est donc très proche de la méthode **LINF1** du chapitre précédent, mais profite des améliorations apportées au système en ce qui concerne le lissage des pistes, la normalisation des scores de détection et l'endormissement des pistes. Comparer les approches **DSR** et **DOSR** permet ainsi d'étudier uniquement l'influence des dictionnaires denses sur les performances.

Expérimentation avec des détecteurs simulés

Nous évaluons ici les approches **DSR** et **DOSR** en simulant des détecteurs plus ou moins performants sur l'ensemble d'entraînement de la version 2015 du *MOTChallenge*. L'objectif est ici d'étudier la robustesse de ces deux approches vis-à-vis de la qualité du détecteur d'objets employé, afin de vérifier si l'emploi de dictionnaires denses permet effectivement de compenser certaines erreurs du détecteur. L'implémentation de l'article [115] est alors utilisée pour générer les jeux de détections. Les vérités terrains sont dégradées en supprimant un certain nombre de détections, en bruitant les détections restantes et en ajoutant des fausses détections près des détections de la vérité terrain. La dégradation des vérités terrains est alors contrôlée

2DMOT2015 - Base d'entraînement - Détections publiques										
Méthode	**MOTA** ↑	**IDS** ↓	**IR** ↓	**FM** ↓	**FAF** ↓	**MOTP** ↑	**FP** ↓	**FN** ↓	**MT** (%) ↑	**ML** (%) ↓
DOSR	0.366	**115**	**2.6**	**249**	**0.6**	**0.741**	**3174**	22009	19.5	55.2
DSR	**0.377**	128	2.7	267	0.7	0.740	3739	**20974**	**20.3**	**53.4**

Tableau V.1 – Résultats pour l'approche proposée, avec des dictionnaires denses (**DSR**) ou avec des dictionnaires utilisant uniquement des détections (**DOSR**), testées sur l'ensemble d'entraînement de la base de données *2DMOT2015*. Meilleures valeurs en gras et rouge.

de manière à obtenir un jeu de détections avec des valeurs de rappel et de précision pré-définies. Étant donné un pourcentage p_{qual}, qui traduit la qualité du détecteur simulé, on détermine ainsi un jeu de détections avec des taux de rappel et de précision valant p_{qual}. Si $p_{qual} = 100\%$, la qualité du détecteur est considérée parfaite et les vérités terrains ne sont pas dégradées.

Nos méthodes **DSR** et **DOSR** sont évaluées en faisant varier la qualité des détecteurs simulés en considérant quarante valeurs de p_{qual}, uniformément réparties entre 2.5% et 100%. Pour évaluer nos approches, employer un jeu de paramètres relativement indépendant de la qualité du détecteur simulé est souhaitable. Pour chaque approche (**DSR** ou **DOSR**), un unique jeu de paramètres est déterminé par une procédure d'hyper-optimisation en optimisant la moyenne des résultats en MOTA obtenus avec des détections de qualité $p_{qual} = 25\%$, $p_{qual} = 50\%$ et $p_{qual} = 75\%$. Ce jeu de paramètre est alors utilisé pour évaluer la méthode concernée sur toutes les valeurs de qualité p_{qual}.

Les résultats, en termes de MOTA et de changements d'identité (IDS), sont indiqués à la figure V.6. Les résultats pour l'ensemble des métriques usuelles sont de plus donnés en annexe C. On peut observer que l'approche **DSR**, qui emploie des dictionnaires denses, est plus robuste que l'approche **DOSR** vis-à-vis de la qualité du détecteur simulé. En effet, bien que les performances soient similaires pour des qualités p_{qual} élevées, l'approche **DSR** surpasse largement l'approche **DOSR** pour des qualités de détecteurs faibles pour presque toutes les métriques considérées. Il n'y a qu'en termes de MOTP que l'approche **DSR** est très légèrement inférieure. Cela peut se justifier si on suppose que les dictionnaires denses permettent d'associer correctement des détections davantage bruitées, ce qui a alors tendance à diminuer la précision de localisation des cibles indiquée par le MOTP. Même pour des qualités de détection élevées, l'approche **DSR** produit bien moins de changements d'identité (IDS) comparée à la variante **DOSR**. Cet écart reste ensuite plus ou moins constant lorsque la qualité du détecteur diminue. L'approche **DSR** ayant dans ce cas des valeurs en MOTA et MT plus élevées, ce qui indique que davantage de cibles sont estimées, celle-ci est cependant plus soumise à des changements d'identité que la méthode **DOSR**.

Comparaison sur les ensembles d'entraînement du *MOTChallenge*

Nous évaluons maintenant l'apport de l'approche avec des dictionnaires denses en situation réelle, en évaluant les méthodes **DSR** et **DOSR** sur les ensembles d'entraînement du *MOTChallenge*, en considérant les versions 2015 et 2016. Les résultats

MOT16 - Base d'entraînement											
Méthode	**Det.**	**MOTA** ↑	**IDS** ↓	**IR** ↓	**FM** ↓	**FAF** ↓	**MOTP** ↑	**FP** ↓	**FN** ↓	**MT** (%) ↑	**ML** (%) ↓
DOSR	**Pub.**	0.359	472	11.6	469	**0.9**	**0.792**	**4671**	65653	13.2	**47.9**
DSR	**Pub.**	**0.362**	**359**	**8.7**	**399**	**0.9**	0.789	5073	**65035**	**13.3**	48.8
DOSR	**[131]**	0.562	865	13.2	799	**1.7**	**0.821**	**9182**	38271	35.7	**17.0**
DSR	**[131]**	**0.567**	**747**	**11.4**	**738**	**1.7**	**0.821**	9210	**37798**	**36.6**	17.2

Tableau V.2 – Résultats pour l'approche proposée, avec des dictionnaires denses (**DSR**) ou avec des dictionnaires utilisant uniquement des détections (**DOSR**), testées sur l'ensemble d'entraînement de la base de données *MOT16*. Deux jeux de détections sont utilisés, à savoir les détections publiques de cette base de données et celles estimées par un détecteur de type Faster-RCNN (données par l'article [131]). Meilleures valeurs en gras et rouge, pour chaque jeu de détections considéré.

2DMOT2015 - Base d'entraînement - Détections publiques											
Méthode	**Lissage**	**MOTA** ↑	**IDS** ↓	**IR** ↓	**FM** ↓	**FAF** ↓	**MOTP** ↑	**FP** ↓	**FN** ↓	**MT** (%) ↑	**ML** (%) ↓
DOSR	**REG**	0.349	182	4.2	299	**0.5**	0.738	**2985**	22796	16.2	56.3
DOSR	**INT**	0.359	165	3.7	437	0.6	0.727	3482	21932	19.4	53.9
DOSR	**RDP**	0.366	**115**	**2.6**	**249**	0.6	**0.741**	3174	22008	19.5	55.2
DSR	**REG**	0.358	161	3.6	285	0.6	0.738	3535	21938	17.9	56.0
DSR	**INT**	0.360	143	3.1	417	0.7	0.726	3817	21584	20.0	53.5
DSR	**RDP**	**0.377**	128	2.7	267	0.7	0.740	3739	**20974**	**20.3**	**53.4**

Tableau V.3 – Résultats pour les différents lissages **REG** (régression linéaire), **INT** (intérpolation linéaire) et **RDP**, testés sur l'ensemble d'entraînement de la base de données *2DMOT2015*. Ces méthodes sont expérimentées avec des dictionnaires denses (**DSR**) ou avec des dictionnaires utilisant uniquement des détections (**DOSR**). Meilleures valeurs en gras et rouge, secondes meilleures soulignées en bleu.

obtenus, après une procédure d'hyper-optimisation, sont indiqués au tableau V.1 et au tableau V.2. Sur la version 2016 du *MOTChallenge*, nous avons aussi évalué ces deux variantes avec des détections données par un détecteur d'objets plus pertinent de type *Faster-RCNN* [131]. Nos deux approches **DSR** et **DOSR** sont alors testées en utilisant ces détections privées et en utilisant les même caractéristiques visuelles que pour les détections publiques (à savoir les intensité RGB des boîtes). On peut remarquer que l'emploi des dictionnaires denses donne toujours de meilleurs résultats en MOTA, en trajectoires majoritairement suivies (MT) et en faux négatifs (FN). Les gains sont moins significatifs en MOTA sur la version 2016 du *MOTChallenge*, mais plus importants en ce qui concerne les changements d'identité (IDS) et le nombre de fragments (FM). Sur la version 2015, le gain en MOTA est important, mais d'autres métriques secondaire (IDS, FM, FAF, MOTP) sont légèrement moins bonnes tout en restant proches des résultats de la variante **DOSR**.

Influence des méthodes de lissage

Nous évaluons maintenant trois variantes de lissage des pistes, utilisées à la fois au sein de l'approche avec dictionnaires denses **DOSR** et avec l'approche sans dictionnaires denses **DSR**. Ces méthodes consistent en une interpolation linéaire, une régression linéaire, ou bien la variante de la méthode RDP proposée. Ces six variantes sont évaluées sur la base d'entraînement de la version 2015 du *MOTChallenge*, et les résultats associés sont indiqués au tableau V.3.

Plusieurs remarques peuvent alors être observées. Tout d'abord, changer de méthode de lissage des pistes a un impact non négligeable sur les performances de suivi. Le tableau V.3 indique que l'approche de type RDP donne de meilleurs résultats, et que l'interpolation linéaire est à privilégier par rapport à une régression linéaire en termes de MOTA. Ce classement entre ces trois méthodes est assez indépendant du type de dictionnaire employé (**DSR** ou **DOSR**). En effet, bien que les résultats soient dans l'ensemble légèrement meilleurs avec les dictionnaires denses, les effets observés entre les différentes approches de lissage sont similaires. On remarque que le lissage proposé (**RDP**) fonctionne mieux que les deux autres sur une grande majorité de métriques. La comparaison entre la régression et l'interpolation linéaire est moins claire. Bien que l'interpolation linéaire donne de meilleures performances sur la plupart des métriques, comparée à la régression linéaire, ce n'est néanmoins pas le cas en ce qui concerne le nombre de fragments (FM) et la précision (MOTP).

Ces comparaisons permettent donc de confirmer le choix de la méthode de lissage proposée, décrite par l'algorithme V.4, par rapport à des approches basiques comme une interpolation ou régression linéaire.

Conclusion vis-à-vis des dictionnaires denses

Dans les différents tests effectués, que ce soit au tableau V.3 en jouant sur les méthodes de lissage utilisées, à la figure V.6 en simulant des détecteurs plus ou moins performants, ou bien encore au tableau V.1 et au tableau V.2 sur les bases de données du *MOTChallenge* avec différents détecteurs, l'emploi de dictionnaires denses améliore toujours les résultats en termes de MOTA. Sur l'ensemble des métriques, utiliser les dictionnaires denses permet d'avoir parfois des gains significatifs, comme c'est le cas sur la version 2016 du *MOTChallenge* avec les détections publiques ou privées de [131], ou bien des performances seulement légèrement dégradées pour certaines métriques par rapport à la variante sans dictionnaires denses.

Les dictionnaires denses renforcent donc, de façon modérée, les performances de l'approche de suivi. Néanmoins, les tests simulant des détecteurs plus ou moins fiables semblent indiquer que des gains significatifs en performance s'obtiennent pour des détecteurs de très mauvaise qualité.

V.4.3 Comparaison aux méthodes récentes de l'état de l'art

Nous évaluons maintenant notre approche **DSR** sur les ensembles de test du *MOTChallenge*, à la fois pour la version 2015 et 2016. Le jeu de paramètres employé pour chacune de ces versions est alors le jeu de paramètres maximisant le MOTA sur l'ensemble d'entraînement, qui est déterminé par hyper-optimisation. Les résultats sont donnés au tableau V.4 et au tableau V.5, où nous indiquons les résultats de toutes les approches publiées du *MOTChallenge*, en nous limitant à celles employant

2DMOT2015 - Base de test - Détections publiques												
Méthode	**Réf.**	**T.**	**MOTA** ↑	**IDS** ↓	**IR** ↓	**FM** ↓	**FAF** ↓	**MOTP** ↑	**FP** ↓	**FN** ↓	**MT** (%)↑	**ML** (%)↓
TSMLCDEn.	[120]	**H**	**0.343**	618	12	959	1.4	0.717	7869	31908	14.0	39.4
NOMT	[24]	**H**	0.337	442	9.4	823	1.3	0.719	7762	32547	12.2	44.0
TDAM	[128]	**E**	0.330	464	9.2	1506	1.7	**0.728**	10064	**30617**	13.3	39.1
MHT_DAM	[62]	**H**	0.324	435	9.1	826	1.6	0.718	9064	32060	**16.0**	43.8
MDP	[127]	**E**	0.303	680	14	1500	1.7	0.713	9717	32422	13.0	**38.4**
DSR	-	**H**	0.298	**269**	**6.8**	**688**	1.0	0.722	5692	37153	9.7	54.5
CNNTCM	[121]	**H**	0.296	712	16	943	1.3	0.718	7786	34733	11.2	44.0
SCEA	[51]	**E**	0.291	604	15	1182	1.0	0.711	6060	36912	8.9	47.3
SiameseCNN	[68]	**H**	0.290	639	16	1316	**0.9**	0.712	**5160**	37798	8.5	48.4
olCF	[61]	**E**	0.271	454	11	1660	1.3	0.700	7594	36757	6.4	48.7
LP_SSVM	[122]	**H**	0.252	646	16	849	1.4	0.717	8369	36932	5.8	53.0
ELP	[84]	**H**	0.250	1396	35	1804	1.3	0.712	7345	37344	7.5	43.8
LINF1	[37]	**H**	0.245	298	8.6	744	1.0	0.713	5864	40207	5.5	64.6
JPDA_m	[108]	**H**	0.238	365	10	869	1.1	0.682	6373	40084	5.0	58.1
MotiCon	[69]	**H**	0.231	1018	24	1061	1.8	0.709	10404	35844	4.7	52.0
SegTrack	[90]	**H**	0.225	697	19	737	1.4	0.717	7890	39020	5.8	63.9
EAMTTpub	[111]	**E**	0.223	833	22	1485	1.4	0.708	7924	38982	5.4	52.7
DCO_X	[89]	**H**	0.196	521	13	819	1.8	0.714	10652	38232	5.1	54.9
CEM	[88]	**H**	0.193	813	18	1023	2.5	0.707	14180	34591	8.5	46.5
RNN_LSTM	[87]	**E**	0.190	1490	37	2081	2.0	0.710	11578	36706	5.5	45.6
RMOT	[130]	**E**	0.186	684	17	1282	2.2	0.696	12473	36835	5.3	53.3
TSDA_OAL	[56]	**E**	0.186	806	17	1544	2.8	0.697	16350	32853	9.4	42.3
GMPHD_15	[116]	**E**	0.185	459	14	1266	1.4	0.709	7864	41766	3.9	55.3
SMOT	[32]	**H**	0.182	1148	33	2132	1.5	0.712	8780	40310	2.8	54.8
ALExTRAC	[15]	**H**	0.170	1859	53	1872	1.6	0.712	9233	39933	3.9	52.4
TBD	[44]	**H**	0.159	1939	44	1963	2.6	0.709	14943	34777	6.4	47.9
GSCR	[36]	**E**	0.158	514	17	1010	1.3	0.694	7597	43633	1.8	61.0
TC_ODAL	[6]	**E**	0.151	637	17	1716	2.2	0.705	12970	38538	3.2	55.8
DP_NMS	[102]	**H**	0.145	4537	104	3090	2.3	0.708	13171	34814	6.0	40.8

Tableau V.4 – Résultats de l'approche proposée **DSR** sur l'ensemble de test de la base de données *2DMOT2015*. Résultats comparés à ceux des autres approches de l'état de l'art publiées sur cette base de données au 15/12/2016 (meilleures valeurs en gras et rouge, secondes meilleures soulignées en bleu). Troisième colonne : type d'approche avec E pour en ligne et H pour hors ligne.

les détections publiques fournies. Les méthodes auxquelles nous nous comparons ici sont donc particulièrement récentes et rassemblent un grand nombre de méthodes de suivi multi-objets proposées au cours de ces trois dernières années. Certaines trajectoires estimées par notre approche sont de plus données en figure V.7.

Concernant la version 2015 du *MOTChallenge*, dont les résultats sont indiqués au tableau V.4, notre méthode **DSR** donne des performances satisfaisantes par rapport aux autres approches. Notre méthode se classe 6ème sur 29 en termes de MOTA, qui est la métrique principale du *MOTChallenge*. De plus, notre approche obtient de très bonnes performances sur des métriques secondaires. La méthode **DSR** est première en nombre de changements d'identité (IDS) (y compris en considérant le ratio de changements d'identités IR), et en termes de fragments (FM). De surcroît, notre

MOT16 - Base de test - Détections publiques												
Méthode	**Réf.**	**T.**	**MOTA** ↑	**IDS** ↓	**IR** ↓	**FM** ↓	**FAF** ↓	**MOTP** ↑	**FP** ↓	**FN** ↓	**MT** (%)↑	**ML** (%)↓
NOMT	[24]	**H**	**0.464**	**359**	**6.9**	**504**	1.6	0.766	9753	**87565**	**18.3**	41.4
JMC	[117]	**H**	0.463	657	13	1114	1.1	0.757	6373	90914	15.5	**39.7**
olCF	[61]	**E**	0.432	381	8.1	1404	1.1	0.743	6651	96515	11.3	48.5
MHT_DAM	[62]	**H**	0.429	499	10	659	1.0	0.766	5668	97919	13.6	46.9
DSR	-	**H**	0.428	688	14	756	1.1	**0.776**	6372	97214	12.8	45.8
LINF1	[37]	**H**	0.410	430	9.4	963	1.3	0.748	7896	99224	11.6	51.3
EAMTT_pub	[111]	**E**	0.388	965	22	1657	1.4	0.751	8114	102452	7.9	49.1
OVBT	[7]	**E**	0.384	1321	29	2140	1.9	0.754	11517	99463	7.5	47.3
LTTSC-CRF	[66]	**H**	0.376	481	10	1012	2.0	0.759	11969	101343	9.6	55.2
TBD	[44]	**H**	0.337	2418	63	2252	1.0	0.765	5804	112587	7.2	54.2
CEM	[88]	**H**	0.332	642	17	731	1.2	0.758	6837	114322	7.8	54.4
DP_NMS	[102]	**H**	0.322	972	29	944	**0.2**	0.764	**1123**	121579	5.4	62.1
GMPHD_H.	[116]	**E**	0.305	539	16	731	0.9	0.754	5169	120970	4.6	59.7
SMOT	[32]	**H**	0.297	3108	75	4483	2.9	0.752	17426	107552	5.3	47.7
JPDA_m	[108]	**H**	0.262	365	12	638	0.6	0.763	3689	130549	4.1	67.5

Tableau V.5 – Résultats de l'approche proposée **DSR** sur l'ensemble de test de la base de données *MOT16*. Résultats comparés à ceux des autres approches de l'état de l'art publiées sur cette base de données au 15/12/2016 (meilleures valeurs en gras et rouge, secondes meilleures soulignées en bleu). Troisième colonne : type d'approche avec E pour en ligne et H pour hors ligne.

approche est deuxième en précision de localisation des cibles (MOTP), en nombre moyen de fausses alarmes par image (FAF) et en faux positifs (FP). Certaines de ces métriques, comme le nombre de changement d'identité (IDS) ou de fausses alarmes (FAF) dépendent du nombre de faux négatifs (FN), une méthode estimant plus de cibles ayant naturellement plus de risque de générer de telles erreurs. Cependant, notre approche reste bien meilleure sur ces métriques que d'autres approches avec un nombre de faux négatifs (FN) similaire.

Les résultats pour la version 2016 du *MOTChallenge* sont indiqués au tableau V.5. Notre méthode **DSR** se classe correctement en termes de MOTA, en étant positionnée 5ème sur 15. Les résultats des métriques secondaires sont alors comparables aux autres méthodes publiées et notre approche atteint le meilleur score en MOTP. En conclusion, les bonnes performances en MOTP sur le *MOTChallenge* montrent que l'approche de lissage de type RDP proposée en sous-section V.3.2 est suffisamment pertinente pour affiner correctement les positions des détections.

Conclusion

Dans ce chapitre, nous avons proposé de travailler sur les dictionnaires utilisés pour calculer les représentations parcimonieuses en norme $l_{\infty,1}$ pondérée. Des dictionnaires denses sont formulés de manière à inclure un grand nombre de positions des images de la fenêtre glissante qui ne sont pas nécessairement détectées par le détecteur d'objets. Tout élément du dictionnaire final n'étant plus lié à une détection mais à une position, un nouveau modèle d'apparence $App_{Dense}(C)$ pour l'énergie

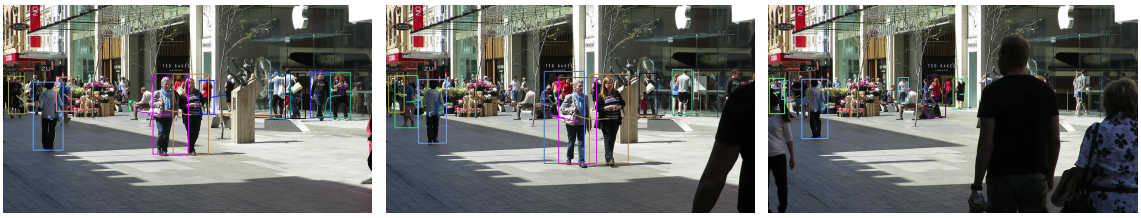


Image n°300 | **Image n°350** | **Image n°400**

ADL-Rundle-3

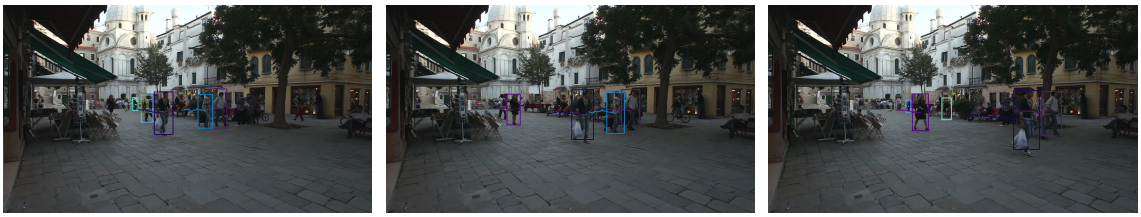


Image n°200 | **Image n°250** | **Image n°300**

Venice1

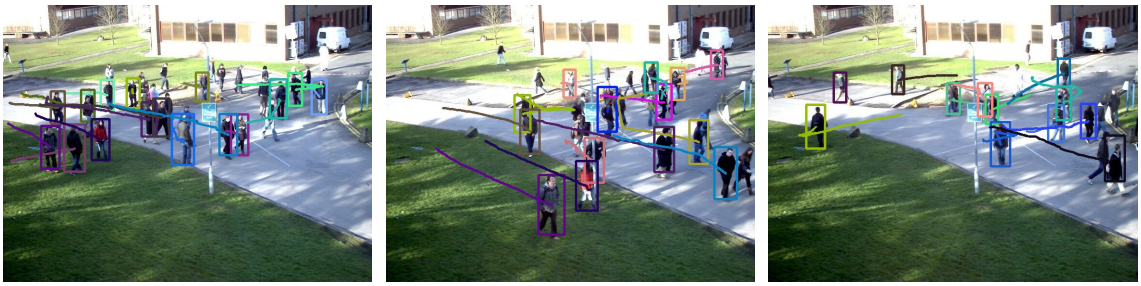


Image n°50 | **Image n°100** | **Image n°150**

PETS S2L2

FIGURE V.7 – Certaines trajectoires estimées par notre méthode **DSR** sur l'ensemble de vidéos de test de la base de données *2DMOT2015* en utilisant les détections publiques fournies.

globale $E(C)$ est alors défini pour tenir compte de tels changements. Afin de pouvoir calculer efficacement les représentations parcimonieuses avec les dictionnaires proposés, qui sont composés d'un nombre considérable d'éléments, nous adaptons de façon appropriée les méthodes d'optimisation proximales avec ensembles actifs. La méthode d'optimisation qui en résulte exploite alors des techniques de calcul de corrélations croisées normalisées à base de transformées de Fourier rapides et d'images intégrales. Pour finir, un lissage plus précis des boîtes des pistes, utilisé au sein de l'association de données par MCMCDA, est proposée en s'inspirant de la méthode de simplification de polygones RDP (*Ramer-Douglas-Peucker*).

L'emploi de dictionnaires denses pour calculer des représentations parcimonieuses structurées, au sein d'une approche de suivi à fenêtre glissante, permet de moins dépendre de la qualité du détecteur d'objets. En effet, les dictionnaires denses incluent un grand nombre de positions non détectées qui compensent d'éventuelles erreurs du détecteur d'objets, comme des cibles non détectées ou des détections imprécises. Un grand nombre d'expérimentations quantitatives ont été effectuées pour évaluer ce système de suivi, à partir des bases de données du *MOTChallenge*. Ces évaluations montrent que l'emploi de dictionnaires denses améliore dans tous les cas, de façon

certes modérée, les performances en termes de MOTA. Dans la majorité des cas, les métriques secondaires, en particulier le nombre de changements d'identité (IDS) et de fragments (FM) ainsi que les taux de pistes majoritairement suivies (MT) et perdues (ML), sont meilleures lorsque les dictionnaires denses sont utilisés. Des tests réalisés en simulant des détecteurs d'objets de qualité variable indiquent que les dictionnaires proposés rendent l'approche de suivi plus robuste vis-à-vis de la qualité du détecteur d'objets. Pour finir, les très bons résultats sur le *MOTChallenge* en MOTP, valeur liée à la précision de la localisation des cibles, montrent l'efficacité de l'approche proposée pour lisser les détections des pistes au sein du MCMCDA.

Ces résultats, très récents, n'ont pas donné lieu à publication à ce jour.

Chapitre VI

Conclusion et perspectives

Sommaire

VI.1 Conclusion

Au cours de cette thèse, plusieurs approches de suivi multi-objets ont été proposées en cherchant à exploiter des représentations parcimonieuses pour modéliser l'apparence des cibles. Cette axe d'études était initialement motivé pour les raisons suivantes. Tout d'abord, de nombreuses méthodes de suivi multi-objets cherchaient à gagner en performances en exploitant l'apparence des cibles, en utilisant éventuellement des classifieurs multi-classes pour différencier ces cibles les unes des autres. Les représentations parcimonieuses avaient alors été largement employées dans plusieurs domaines en Vision par Ordinateur. Comme ces représentations avaient été employées, et avec succès, à la fois pour le suivi mono-objet et pour la classification multi-classes, il était intéressant de s'interroger sur l'intérêt des représentations parcimonieuses pour le suivi multi-objets.

Nous avons ainsi cherché à exploiter des représentations parcimonieuses au sein de méthodes de suivi multi-objets par détection, en nous inspirant des méthodes de suivi mono-objet à base de représentations parcimonieuses mais surtout des techniques de classification multi-classes utilisant de telles représentations. Nous considérons en effet que certaines difficultés cruciales en suivi multi-objets par détection, en particulier celles liées à l'étape d'association de données, sont plus proches des problématiques de classification que des problématiques de suivi mono-objet.

Au chapitre III, une approche de suivi multi-objets en ligne qui exploite des représentations parcimonieuses collaboratives entre les cibles est ainsi proposée. Les représentations parcimonieuses sont alors utilisées pour définir des valeurs d'affinité

entre les trajectoires et les détections en s'inspirant de la méthode de classification multi-classes [124]. Tout comme cette approche de classification, des représentations collaboratives entre les trajectoires sont utilisées pour déterminer les valeurs d'affinité. Néanmoins, plusieurs possibilités sont envisageables pour le dictionnaire utilisé. Ce dictionnaire peut faire intervenir uniquement les trajectoires proches de la détection considérée ou bien faire aussi intervenir les autres trajectoires de la scène observée, même très éloignées de la détection. Des expérimentations quantitatives montrent alors un gain en performances lorsque le dictionnaire considère l'ensemble des trajectoires, c'est-à-dire en considérant une représentation collaborative entre toutes les cibles observées. Cela a mené à l'approche de suivi nommée **GSCR**, qui montre de bonnes performances par rapport à d'autres approches de suivi (surtout en nombre de changements d'identité).

Considérer des représentations collaboratives globales entre toutes les cibles pose néanmoins un problème en termes de coût CPU. En effet, calculer des représentations parcimonieuses est assez coûteux en temps de calcul et cela d'autant plus que le dictionnaire considéré comporte un grand nombre d'éléments. Afin de permettre un fonctionnement suffisamment rapide de notre approche, nous avons proposé d'exploiter des techniques d'optimisation plus adaptées à ce cas de figure. En utilisant une optimisation à base de méthodes proximales et d'ensembles actifs, nous avons alors montré que notre approche pouvait fonctionner à une vitesse proche du temps réel. Cette première approche de suivi a alors été publiée à *IEEE ICIP* [36].

En nous inspirant d'une approche proposée en suivi mono-objet [55], l'approche **GSCR** a été étendue au cas de descriptions locales des cibles. Employer une description locale des cibles, par exemple avec des patchs locaux pris autour de points d'intérêt, permet alors d'améliorer les performances de notre approche. Cela est possible en prenant en compte des considérations spatiales des patchs locaux au sein du dictionnaire et des erreurs de reconstruction résiduelle, menant à une nouvelle approche de suivi appelée **CSSR**. Cette amélioration des performances se fait néanmoins au détriment de la vitesse de notre approche. Cette seconde méthode de suivi a alors mené à une publication à *IEEE AVSS* [38].

Au chapitre IV, nous avons cherché à exploiter davantage d'information temporelle future pour obtenir une approche de suivi plus robuste et performante. Prendre en compte des images futures pour traiter l'instant courant est une stratégie exploitée par les méthodes de suivi hors ligne, méthodes de suivi très probantes dans la littérature récente. Cela permet en effet d'attendre le temps nécessaire pour désambiguïser certaines situations délicates, plutôt que de devoir faire des choix stricts d'appariement dès que l'image courante est reçue. Afin de gagner en performances tout en maintenant un temps de latence faible, nous avons considéré une approche à fenêtre glissante de type MCMCDA, qui résout l'étape d'association de données à l'aide d'un échantillonnage de type MCMC. Nous avons alors formulé une énergie globale E pour résoudre l'étape d'association de données, désormais multi-images, en exploitant les représentations parcimonieuses des détections de la fenêtre glissante.

Il s'avère cependant que les représentations parcimonieuses précédemment proposées ne sont pas adaptées à un tel contexte. En effet, les représentations parcimonieuses considérées pour nos approches précédentes (**GSCR** et **CSSR**) sont des représentations parcimonieuses définies, de façon classique, à partir d'une pénalisation en norme l_1. Cette pénalisation induit une structure de parcimonie, le support des représentations étant restreint à un faible nombre d'éléments. Si cette structure

de parcimonie était adaptée pour nos approches précédentes, afin de permettre de différencier des cibles d'apparence proche, cette structure n'est pas appropriée pour le cas d'une association de données multi-images. Les détections représentées ici ne sont pas uniquement associées à une trajectoire, mais sont aussi associées à d'autres détections de la fenêtre glissante. Des représentations parcimonieuses usuelles en norme l_1 tendent à représenter chaque détection par un nombre trop faible de détections de la fenêtre glissante et ne permettent pas d'exploiter judicieusement toute l'information temporelle alors disponible.

Nous avons de ce fait envisagé au chapitre IV d'autres pénalisations que la norme l_1 afin d'induire une structure de parcimonie plus adaptée à notre contexte d'association de données multi-images. Nous avons alors proposé l'emploi d'une norme $l_{\infty,1}$ pondérée, qui permet de définir des représentations parcimonieuses structurées plus appropriées. Ces représentations favorisent notamment chaque détection à être représentée uniquement par les autres détections de sa cible, et facilitent de ce fait l'association de données multi-images. Calculer des représentations parcimonieuses en norme $l_{\infty,1}$ pondérée s'avère cependant plus complexe que les représentations en norme l_1. Le problème d'optimisation associé peut cependant se ramener à des calculs de projections sur la boule unité d'une norme $l_{1,\infty}$ pondérée, projections pour lesquelles il existe des algorithmes efficaces [103]. De telles considérations nous permettent d'optimiser efficacement les représentations parcimonieuses en norme $l_{\infty,1}$ pondérée, et d'aboutir à une méthode de suivi avec une vitesse proche temps réel.

L'approche de suivi ainsi proposée, appelée **LINF1**, se compare alors favorablement aux autres approches de suivi récentes sur la base de données du *MOT-Challenge*. Cette méthode obtient en particulier de très bonnes performances pour certaines métriques secondaires qui jugent davantage la qualité de l'association de données, comme les changements d'identité (IDS) ou le nombre de fragmentation (FM). Cette approche a été publiée à *ECCV* [37].

Bien que les représentations parcimonieuses structurées en norme $l_{\infty,1}$ soient plus adaptées pour une association de données multi-images, ces représentations supposent idéalement que chaque cible soit détectée sur chaque image. En pratique le détecteur d'objets employé génère des non détections et des détections mal positionnées qui ne sont pas prises en compte par ces représentations structurées. Ce problème est en grande partie dû au fait que le dictionnaire considéré fait seulement intervenir les détections effectives des cibles. Dans le chapitre V, nous avons de ce fait proposé d'utiliser des dictionnaires denses, qui ne se limitent pas aux seules détections, pour définir les représentations parcimonieuses en norme $l_{\infty,1}$ pondérée. Ces dictionnaires font intervenir un grand nombre de positions au sein des images de la fenêtre glissante. De tels dictionnaires permettent alors de représenter chaque détection avec les positions de la cible associée tout au long de la fenêtre glissante, y compris sur les images où elle n'est pas détectée.

Les dictionnaires denses que nous avons considérés comportent alors un nombre d'éléments trop important pour utiliser sans adaptation les techniques d'optimisation précédentes (à base d'ensembles actifs et de méthodes proximales). Nous nous sommes alors inspirés de techniques utilisées pour optimiser les représentations parcimonieuses à convolution, et nous avons proposé une modification des méthodes d'optimisation à base d'ensemble actifs qui exploite des techniques de recherche de motifs à base de transformées de Fourier et d'images intégrales. L'emploi de telles techniques permet d'optimiser rapidement les représentations avec diction-

naires denses et rend alors leur usage possible dans une approche de suivi multi-objets.

Des évaluations quantitatives sur les bases de données du *MOTChallenge* ont alors montré que l'approche proposée, nommée **DSR**, se comparait favorablement aux autres approches de suivi récentes. Cette approche a en particulier d'excellents résultats en termes de changement d'identité (IDS) et de fragmentation (FM) sur la base de données *2DMOT2015*. Ces derniers résultats, très récents, n'ont pas encore donné lieu à publication à ce jour.

En résumé, les contributions de cette thèse sont listées ci-après. Nos principales contributions sont les suivantes :

(i) Formulation d'une énergie E exploitant des représentations parcimonieuses pour l'association de données multi-images. Une énergie E, qui permet de prendre en compte des représentations parcimonieuses pour l'association de données multi-images, a été proposée dans cette thèse. Une telle énergie permet alors d'exploiter des représentations parcimonieuses des détections au sein d'une méthode de suivi à fenêtre glissante.

(ii) Emploi de représentations parcimonieuses en norme $l_{\infty,1}$ pondérée pour le suivi multi-objets à fenêtre glissante. Nous avons proposé d'utiliser des représentations parcimonieuses structurées présentant une structure de parcimonie plus adaptée pour l'association de données multi-images. Cela nous a mené à considérer des représentations parcimonieuses définies à partir d'une norme $l_{\infty,1}$ pondérée, en montrant que leurs calculs pouvaient s'effectuer efficacement par des considérations sur l'opérateur proximal associé.

Ces contributions centrales font suite à certaines contributions effectuées dans la première partie de cette thèse :

(iii) Emploi de représentations parcimonieuses collaboratives globales pour le suivi multi-objets. Nous avons justifié dans cette thèse l'emploi de telles représentations tout en montrant que des techniques d'optimisation appropriées, à base de méthodes proximales et d'ensembles actifs, rendaient leur utilisation possible dans une approche de suivi proche temps réel.

(iv) Descriptions locales et considérations spatiales pour les représentations parcimonieuses en suivi multi-objets. L'utilisation de descriptions locales des cibles a été envisagée, et nous avons montré que de telles descriptions menaient à des gains en performances sous réserve d'employer certaines considérations spatiales au sein des représentations parcimonieuses.

Enfin, notre dernière contribution consiste en une extension de notre approche de suivi à fenêtre glissante **LINF1**, afin d'employer des dictionnaires denses :

(v) Emploi de représentations parcimonieuses avec dictionnaires denses pour le suivi multi-objets à fenêtre glissante. Des dictionnaires denses, qui incluent un grand nombre de positions non détectées, ont été envisagés pour rendre notre approche de suivi moins dépendante vis-à-vis de la qualité du détecteur d'ob-

2DMOT2015 - Base de test - Détections publiques												
Méthode	**Réf.**	**T.**	**MOTA** ↑	**IDS** ↓	**IR** ↓	**FM** ↓	**FAF** ↓	**MOTP** ↑	**FP** ↓	**FN** ↓	**MT** (%)↑	**ML** (%)↓
GSCR	**[36]**	**E**	0.158	514	17.7	1010	1.3	0.694	7597	43633	1.8	61.0
CSSR	**[38]**	**E**	0.172	350	12.2	909	1.1	0.698	6624	43882	1.7	64.2
LINF1	**[37]**	**H**	0.245	298	8.6	744	**1.0**	0.713	5864	40207	5.5	64.6
DSR	**-**	**H**	**0.298**	**269**	**6.8**	**688**	**1.0**	**0.722**	**5692**	**37153**	**9.7**	**54.5**

Tableau VI.1 – Résultats des approches proposées au cours de cette thèse, évaluées sur l'ensemble de test de la version 2015 du *MOTChallenge*. La configuration des paramètres est déterminée pour chaque méthode par une procédure d'hyper-optimisation sur l'ensemble d'entraînement. Les approches **GSCR** et **CSSR** sont légèrement modifiées par rapport au chapitre III pour permettre de prendre en compte les scores des détections, et en désactivant le filtrage des détections (qui n'est pas adapté pour les vidéos mobiles).

jets. Nous avons alors montré que les techniques d'optimisation avec ensembles actifs pouvaient s'adapter à ce type particulier de dictionnaires à l'aide de techniques de reconnaissances de motifs avec des transformées de Fourier rapides et des images intégrales.

Ces différentes contributions ont abouti à plusieurs méthodes de suivi, de plus en plus performantes, et avec une vitesse de fonctionnement suffisamment rapide pour pouvoir être appliquées avec une latence faible. Nous indiquons les résultats de ces différentes approches sur l'ensemble de test de la version 2015 du *MOTChallenge* en tableau VI.1, qui illustre les gains successifs entre ces différentes approches. Malgré l'aspect très compétitif de la problématique de suivi multi-objets, nos approches se comparent favorablement en performances par rapport aux autres méthodes récentes.

Notre travail met en évidence la pertinence de l'emploi de représentations parcimonieuses pour le suivi multi-objets. Ces représentations parcimonieuses peuvent s'adapter à des contextes particuliers en choisissant une structure de parcimonie appropriée, comme fait au cours de cette thèse avec le cas de la fenêtre glissante. D'autres structures de parcimonie pourraient par exemple être envisagées pour s'adapter plus particulièrement à d'autres contextes, comme du suivi d'objets de classes multiples ou du suivi multi-caméras. L'emploi de représentations parcimonieuses peut de plus être effectué conjointement à l'utilisation de caractéristiques visuelles plus appropriées, comme par exemple celles déterminées par un apprentissage de métrique pour différencier plus aisément les cibles. Nous espérons que ce travail permettra d'amener des perspectives nouvelles pour le problème du suivi multi-objets et de progresser vers des méthodes de suivi encore plus efficaces.

VI.2 Perspectives

Tout au long de cette thèse, les approches proposées ont été progressivement améliorées, en cherchant à corriger les faiblesses observées, en suivant une démarche incrémentale. Cette démarche peut encore être poursuivie, et les travaux présentés

peuvent ainsi donner lieu à de nombreuses extensions. Nous détaillons dans ce qui suit certaines de ces perspectives.

VI.2.1 Représentations structurées plus élaborées

Au chapitre IV, la norme $l_{\infty,1}$ employée permet de prendre en compte un certain a-priori lié à notre contexte de suivi multi-objets à fenêtre glissante. En effet, si la norme l_1 de cette norme de groupes oblige à représenter les détections avec peu de détections au sein d'une même image, la norme l_∞ sur les groupes favorise à être représenté par au moins une détection de chaque image. La norme $l_{\infty,1}$ ainsi proposée permet donc de différencier à la fois les cibles au sein d'une même image tout en prenant en compte le fait qu'une cible est normalement présente sur toutes les images de la fenêtre temporelle. Est-il possible de considérer d'autres a-priori pour favoriser une structure de parcimonie encore plus appropriée ?

Une technique assez employée en Vision par Ordinateur est de représenter les cibles d'intérêt avec plusieurs caractéristiques visuelles (description *multi-features*). Le but est alors de chercher à tirer avantage de l'ensemble des caractéristiques visuelles pour définir une signature plus discriminante des cibles. Employer une telle stratégie dans notre approche de suivi suppose de considérer un ensemble de caractéristiques visuelles $(y_d^k)_k$ pour chaque détection d. Ce cadre est assez similaire aux descriptions locales considérées au chapitre chapitre III, mais ici toutes les caractéristiques visuelles décrivent toujours l'ensemble de l'objet. Au lieu de représenter séparément ces caractéristiques avec une représentation parcimonieuse $\alpha_{y_d^k}$, il est possible de représenter conjointement ces caractéristiques en utilisant une pénalisation Ω pour l'ensemble des représentations $(\alpha_{y_d^k})_k$, c'est à dire résoudre le problème :

$$\min_{\alpha_{y_d^1},..,\alpha_{y_d^{n_k}}} [\sum_k \frac{1}{2}||y_d^k - D_k\alpha_{y_d^{n_k}}||_2^2] + \lambda\Omega(\alpha_{y_d^1},..,\alpha_{y_d^{n_k}})\,. \tag{VI.1}$$

La pénalisation Ω peut être utilisée de manière à favoriser deux conditions :

(i) Idéalement, toute caractéristique y_d^k devrait, comme dans notre approche sans caractéristiques visuelles multiples, être représentée à partir de peu de caractéristiques en chaque image mais être représentée sur l'ensemble des images de la fenêtre temporelle.

(ii) Il est aussi souhaitable de favoriser la participation de toutes les caractéristiques $(y_d^k)_k$ d'une même détection, afin de forcer les différentes caractéristiques de la détection représentée à être reconstruites à partir de caractéristiques provenant des mêmes détections.

Si le point (i) peut être traité avec des normes $l_{\infty,1}$ appliquées indépendamment à chaque représentation $(\alpha_{y_d^k})_k$, le second point (ii) peut typiquement être traité via une norme l_∞ appliquée aux caractéristiques d'une même détection. Il est alors possible de traiter simultanément les points (i) et (ii) en prenant pour pénalisation Ω une norme $l_{\infty,1,\infty}$ [1].

Une perspective intéressante est ainsi d'envisager des caractéristiques multiples pour décrire les cibles et d'employer des représentations parcimonieuses en norme $l_{\infty,1,\infty}$. Toute la difficulté est ici d'arriver à optimiser correctement de telles représentations, éventuellement en adaptant les méthodes proximales utilisées dans cette

1. $l_{\infty,1,\infty}(u) = max_{G_i}||u|_{G_i}||_{1,\infty}$

thèse pour traiter une telle pénalisation. Les propositions formulées en annexe B indiquent déjà que la norme duale de la norme $l_{\infty,1,\infty}$ est la norme $l_{1,\infty,1}$, et l'opérateur proximal de la norme $l_{\infty,1,\infty}$ peut donc se ramener à une projection sur la boule unité de la norme $l_{1,\infty,1}$. Pouvoir calculer rapidement cette projection serait alors suffisant pour pouvoir déterminer efficacement des représentations parcimonieuses en norme $l_{\infty,1,\infty}$.

Il est aussi possible de chercher à définir des représentations structurées plus complexes pour pénaliser l'ensemble des représentations des détections de la fenêtre glissante. L'idée serait ici d'éviter que deux détections d'une même image, qui correspondent normalement à deux cibles différentes, se représentent à partir des mêmes détections. La principale difficulté est que les pénalisations Ω favorisant une telle structure (au niveau de l'ensemble des représentations) feraient intervenir des groupes d'éléments non disjoints, ce qui constitue un cas de figure beaucoup plus délicat à gérer pour l'optimisation par méthodes proximales.

VI.2.2 Représentations parcimonieuses à noyaux

Dans notre approche de suivi, les représentations parcimonieuses ont pour objectif de représenter chaque détection par d'autres détections proches en apparence tout en favorisant une certaine structure des représentations. Cette proximité en apparence est évaluée via le terme $||y - D\alpha||_2^2$, ce qui se ramène en réalité à comparer les détections en utilisant le produit scalaire de l'espace de leurs caractéristiques. On peut d'ailleurs remarquer que, lors de l'optimisation par ensembles actifs de représentations parcimonieuses usuelles en norme l_1, on cherche à agrandir l'ensemble actif $\mathcal{A}$ en sélectionnant les éléments dont le produit scalaire avec la partie non reconstruite de y, $(D\widetilde{\alpha_{\mathcal{A}}} - y)$, est élevé.

Plusieurs algorithmes utilisés en Vision par Ordinateur, notamment d'apprentissage, peuvent être étendus de façon à exploiter des *noyaux*. Un noyau est une fonction permettant de comparer deux éléments, qui ne sont pas forcément des vecteurs (par exemple, on peut définir des noyaux pour comparer des graphes). Sous certaines conditions, un noyau peut vérifier toutes les caractéristiques d'un produit scalaire, dans un espace $\mathcal{E}$ non nécessairement explicite. Un tel noyau peut alors être utilisé pour remplacer les produits scalaires d'un algorithme afin de permettre d'appliquer virtuellement l'algorithme dans l'espace $\mathcal{E}$. Les méthodes à noyaux présentent deux intérêts principaux. Elles peuvent être plus pertinentes que leurs versions d'origine en se ramenant virtuellement à un espace $\mathcal{E}$ plus approprié, et elles permettent aussi d'appliquer des algorithmes sur des éléments non vectoriels (pour faire une classification sur des graphes par exemple).

Les représentations parcimonieuses peuvent aussi être formulées avec un noyau comme proposé dans [42]. Des représentations parcimonieuses à noyaux pourraient être exploitées dans notre travail pour deux raisons principales :

(i) Tout d'abord, des représentations parcimonieuses à noyaux peuvent être envisagées pour étendre nos approches de suivi vers des méthodes qui raisonneraient sur des éléments plus complexes que des détections. Par exemple, on peut envisager de raisonner directement sur des fragments de pistes (*tracklets*) et représenter chaque fragment avec une représentation parcimonieuse à noyaux. La difficulté d'une telle approche est de déterminer un noyau approprié.

(ii) Une seconde possibilité consiste à utiliser l'astuce du noyau dans notre méthode

de suivi afin de pouvoir utiliser des fonctions de similarité, pour comparer deux détections, plus pertinentes que le produit scalaire de leurs caractéristiques visuelles. Par exemple, plusieurs méthodes de suivi multi-objets récentes utilisent des valeurs d'affinité entre détections qui se basent sur du flot optique ou des trajectoires de points d'intérêt [24, 127]. Il est alors envisageable d'utiliser de telles valeurs d'affinité dans notre méthode en définissant des représentations structurées à partir de ces valeurs d'affinité. On peut noter qu'il n'est pas nécessaire d'expliciter ici la fonction noyau. En effet, on peut utiliser des valeurs d'affinités quelconques et déterminer le noyau qui permet de générer des valeurs d'affinité proches [2]. Il serait alors particulièrement intéressant d'étudier si notre approche permet toujours de gagner en performances lorsque les valeurs d'affinité considérées sont déjà très pertinentes.

Cette perspective, tout comme celle concernant des représentations structurées plus élaborées, nécessite une étude approfondie de certains éléments théoriques et de réaliser de nouvelles implémentations. Ces deux premières perspectives sont donc à envisager sur du long terme, plusieurs variantes devant de plus être considérées et évaluées.

VI.2.3 Restriction de l'espace des configurations pour l'association de données par MCMCDA

Une des limitations importantes de l'association de données de type MCMCDA est que l'espace $\mathcal{C}$ des configurations admissibles peut atteindre une taille très importante lorsque la fenêtre glissante temporelle comporte de nombreuses images ou lorsque la scène observée comporte beaucoup de cibles. Plus l'ensemble $\mathcal{C}$ est de taille importante, plus l'optimisation par MCMC risque d'être approchée et risque de donner au final une configuration assez éloignée d'une solution optimale.

On peut remarquer que, dans notre approche de suivi, cet ensemble des configurations admissibles $\mathcal{C}$ est totalement indépendant des représentations parcimonieuses. De même, la proposition d'une nouvelle configuration dans l'algorithme de Metropolis-Hastings employé est indépendante des représentations parcimonieuses, qui n'interviennent que dans la probabilité d'acceptation au travers de la valeur $E(C)$ de la configuration C proposée.

Une piste intéressante serait d'étudier la possibilité de réduire l'espace des configurations $\mathcal{C}$ à partir des représentations parcimonieuses. Par exemple, une configuration C est admissible si les détections consécutives au sein d'une même piste τ sont suffisamment proches spatialement. On peut alors envisager de modifier ce critère pour ne considérer admissibles que les configurations C dont les détections consécutives au sein d'une même piste τ se représentent au sein de leurs représentations parcimonieuses respectives. Arriver à réduire la taille de l'espace $\mathcal{C}$, tout en conservant la configuration optimale, permettrait d'accélérer la convergence du MCMCDA. Cela aboutirait à améliorer la qualité de l'association de données si cette étape doit être réalisée en un temps pré-défini. Il est aussi possible d'envisager de modifier les propositions de configurations en tenant compte des représentations parcimonieuses. Lors d'une mise à jour ou d'une création de piste, il serait intéressant d'attribuer une probabilité (dans la distribution de proposition, avant la probabilité

2. Il est possible pour cela de déterminer la matrice définie positive qui approche au mieux la matrice constituée des valeurs d'affinité de toutes les paires d'éléments possibles.

2DMOT2015 - Base d'entraînement - Détections privées [131]											
Méthode	**Caract.**	**MOTA** ↑	**IDS** ↓	**IR** ↓	**FM** ↓	**FAF** ↓	**MOTP** ↑	**FP** ↓	**FN** ↓	**MT** (%) ↑	**ML** (%) ↓
DOSR	**I. RGB**	0.562	865	13.2	799	**1.7**	0.821	9182	38271	35.7	**17.0**
DSR	**I. RGB**	0.567	747	11.4	738	**1.7**	0.821	9210	**37798**	36.6	17.2
DOSR	**DL**	**0.568**	**694**	**10.6**	**728**	**1.7**	**0.822**	**8939**	38063	**37.8**	**17.0**

Tableau VI.2 – Résultats avec un détecteur de type Faster-RCNN [131]. Tests effectués sur l'ensemble d'entraînement de la base *MOT16*. Les caractéristiques visuelles sont des valeurs d'intensité RGB (**I. RGB**) ou des caractéristiques apprises par DeepLearning (**DL**) [131]. Ces méthodes sont expérimentées avec des dictionnaires denses (**DSR**) ou avec des dictionnaires utilisant uniquement des détections (**DOSR**). Meilleures valeurs en gras et rouge, secondes meilleures soulignées en bleu.

d'acceptation) plus élevée à une piste dont les éléments sont cohérents vis-à-vis de leurs représentations parcimonieuses.

Cette perspective peut être réalisée à très court terme, les changements à envisager ne nécessitant pas une étude théorique poussée ou une modification significative de l'implémentation du MCMCDA.

VI.2.4 Dictionnaires denses avec caractéristiques visuelles par apprentissage profond

Une dernière perspective porte sur les dictionnaires denses proposés au chapitre V. Nous avons évalué au chapitre V, au tableau V.2, nos approches **DSR** et **DOSR** avec des détections privées de type Faster-RCNN fournies par [131]. Cet article fournit également des caractéristiques visuelles apprises par apprentissage profond, dans un contexte de ré-identification de personnes, pour décrire les détections. Nous avons également évalué notre approche **DOSR** avec de telles caractéristiques visuelles. Les résultats de nos approches en utilisant les détections privées de [131], et éventuellement leurs caractéristiques visuelles, sont indiqués au tableau VI.2.

On peut remarquer que, en utilisant les mêmes détections, notre approche avec dictionnaires denses **DSR** permet alors d'améliorer les résultats de suivi par rapport à notre approche sans dictionnaires denses **DOSR**. Cependant, il est possible d'améliorer aussi ces performances sans employer de dictionnaires denses, en utilisant les caractéristiques visuelles par apprentissage profond pour comparer les détections. Une question naturelle est alors de savoir si ces gains en performances sont complémentaires. Est-il possible de cumuler ces gains en employant des caractéristiques visuelles par apprentissage profond avec des dictionnaires denses ?

Plusieurs méthodes de suivi mono-objet, et en particulier [30], exploitent des couches de réseaux convolutionnels comme caractéristiques. Considérer simultanément plusieurs couches convolutionnelles comme caractéristiques visuelles est l'une des stratégies actuelles qui atteint de très bonnes performances en suivi mono-objet. Une piste intéressante serait alors d'étudier la possibilité d'utiliser ces caractéristiques pour définir nos représentations avec dictionnaires denses.

Annexes

Sommaire

Ce chapitre rassemble certains points de détails sur des sujets précédemment développés. En annexe A, des résultats pour les variantes de l'approche **CSSR** sont présentés en exploitant d'autres caractéristiques visuelles que les valeurs d'intensité. Une preuve qui permet de déterminer la norme duale de la norme $l_{\infty,1}$ pondérée est détaillée en annexe B, et les résultats complets des expérimentations effectuées avec des détecteurs simulés en sous-section V.4.2 sont précisés en annexe C.

A. Descriptions locales et caractéristiques visuelles

Nous détaillons ici les résultats obtenus pour l'approche **CSSR**, présentée en section III.4, pour des caractéristiques visuelles autres que des valeurs d'intensité. Nous considérons pour caractéristiques visuelles des HOG, des LBP, et éventuellement des histogrammes de couleurs RGB (*HIST* correspond à la concaténation de trois histogrammes spécifiques à chaque canal RGB tandis que *HIST3D* correspond à un histogramme 3D dans l'espace RGB). Les tendances générales indiquées en section III.4 sont toujours valides, à savoir que l'on gagne généralement en performances en employant des représentations les plus collaboratives possibles (**GSCR** couplé éventuellement avec **SFL**) une fois la description fixée (holistique, points d'intérêt, grille). Néanmoins, décrire localement les cibles ne permet pas de gagner en performances en termes de MOTA lorsque des HOG ou LBP sont employés. C'est néanmoins toujours le cas en termes de changements d'identité, qui sont toujours moins nombreux dans le cas de descriptions locales.

Intensités				
Description		**TSSR**	**LSCR**	**GSCR**
Holistique		0.615	0.622	0.627
Points d'intérêt	**NSL**	0.588	0.613	0.626
	SSL	0.600	0.618	0.631
	SFL	0.624	0.624	**0.634**
Grille	**NSL**	0.581	0.609	0.622
	SSL	0.597	0.616	0.629
	SFL	0.627	0.623	0.627
MOTA ↑				

Intensités				
Description		**TSSR**	**LSCR**	**GSCR**
Holistique		129.0	127.5	125.7
Points d'intérêt	**NSL**	157.5	118.1	107.2
	SSL	135.2	122.8	104.7
	SFL	116.4	107.4	**101.8**
Grille	**NSL**	172.7	137.1	114.8
	SSL	147.2	127.0	108.2
	SFL	119.8	110.8	107.5
IDS ↓				

HIST				
Description		**TSSR**	**LSCR**	**GSCR**
Holistique		0.609	0.610	0.615
Points d'intérêt	**NSL**	0.610	0.620	0.627
	SSL	0.625	0.624	**0.632**
	SFL	0.628	0.628	0.631
Grille	**NSL**	0.607	0.615	0.624
	SSL	0.622	0.624	0.628
	SFL	0.624	0.626	0.629
MOTA ↑				

HIST				
Description		**TSSR**	**LSCR**	**GSCR**
Holistique		161.9	167.1	157.3
Points d'intérêt	**NSL**	137.3	116.4	120.9
	SSL	**110.7**	114.0	113.0
	SFL	120.4	111.3	113.0
Grille	**NSL**	142.9	130.9	124.3
	SSL	123.0	121.7	117.6
	SFL	120.0	116.9	116.7
IDS ↓				

HIST3D				
Description		**TSSR**	**LSCR**	**GSCR**
Holistique		0.614	0.617	0.622
Points d'intérêt	**NSL**	0.606	0.626	0.629
	SSL	0.620	0.632	0.636
	SFL	0.634	0.634	0.636
Grille	**NSL**	0.604	0.623	0.628
	SSL	0.617	0.626	0.634
	SFL	0.634	0.631	**0.637**
MOTA ↑				

HIST3D				
Description		**TSSR**	**LSCR**	**GSCR**
Holistique		149.1	145.0	135.3
Points d'intérêt	**NSL**	151.1	117.0	108.3
	SSL	123.9	108.4	**100.4**
	SFL	117.6	102.4	104.1
Grille	**NSL**	148.1	125.7	115.6
	SSL	133.3	113.6	110.4
	SFL	121.7	111.6	108.1
IDS ↓				

Tableau VI.3 – Valeurs moyennes en MOTA et en changements d'identité (IDS) avec emploi de valeurs d'intensité ou d'histogrammes de couleurs. Meilleure valeur en gras et rouge, seconde meilleure soulignée en bleu.

HOG				
Description		**TSSR**	**LSCR**	**GSCR**
Holistique		0.620	0.620	**0.635**
Points d'intérêt	**NSL**	0.599	0.605	0.622
	SSL	0.607	0.609	0.625
	SFL	0.619	0.619	0.625
Grille	**NSL**	0.602	0.604	0.618
	SSL	0.611	0.612	0.621
	SFL	0.626	0.618	0.621
MOTA ↑				

HOG				
Description		**TSSR**	**LSCR**	**GSCR**
Holistique		130.7	129.1	117.1
Points d'intérêt	**NSL**	141.0	126.4	117.7
	SSL	112.4	128.9	123.7
	SFL	119.3	114.7	**102.0**
Grille	**NSL**	147.1	133.9	134.9
	SSL	135.3	135.7	115.7
	SFL	118.0	123.1	128.7
IDS ↓				

LBP				
Description		**TSSR**	**LSCR**	**GSCR**
Holistique		0.620	0.623	**0.631**
Points d'intérêt	**NSL**	0.578	0.599	0.607
	SSL	0.600	0.610	0.615
	SFL	0.603	0.615	0.617
Grille	**NSL**	0.585	0.601	0.609
	SSL	0.607	0.610	0.619
	SFL	0.616	0.616	0.617
MOTA ↑				

LBP				
Description		**TSSR**	**LSCR**	**GSCR**
Holistique		129.3	127.4	123.3
Points d'intérêt	**NSL**	178.2	145.4	134.4
	SSL	147.1	134.9	114.0
	SFL	149.6	125.0	118.0
Grille	**NSL**	169.3	136.7	133.9
	SSL	141.3	135.0	114.3
	SFL	135.0	116.9	**112.6**
IDS ↓				

Tableau VI.4 – Valeurs moyennes en MOTA et en changements d'identité (IDS) avec emploi de HOG ou de LBP. Meilleure valeur en gras et rouge, seconde meilleure soulignée en bleu.

B. Normes duales de normes de groupes généralisées

La norme duale d'une catégorie particulière de normes de groupes est déterminée en Proposition 4. Ce résultat est ensuite appliqué au cas spécifique de la norme $l_{\infty,1}$ pondérée, afin de déterminer sa norme duale que nous utilisons en sous-section IV.3.3.

B.1 Normes de groupes généralisées

Soit $E = \mathbb{R}^n$ un espace vectoriel de dimension finie avec $n > 0$. Soit G une partition de $\{1, ..., n\}$ qui distribue les dimensions de E dans k groupes disjoints, $G = (G_i)_{1 \leq i \leq k}$.

Étant donné $u \in E$, on définit par $u|_{G_i}$ le vecteur de $\mathbb{R}^{\chi(G_i)}$ déduit du vecteur u en retirant toutes les coordonnées qui ne sont pas incluses dans G_i. Pour chaque groupe G_i, on considère une fonction f_i de $\mathbb{R}^{\chi(G_i)}$ vers $\mathbb{R}$. On écrit alors $u^G_{(f_i)}$ le vecteur de $\mathbb{R}^k$ défini par :

$$u^G_{(f_i)} = \begin{bmatrix} f_1(u|_{G_1}) \\ \vdots \\ f_k(u|_{G_k}) \end{bmatrix}.$$

Étant donnée une norme Ω sur $\mathbb{R}^k$ et des normes Ω_i sur $\mathbb{R}^{\chi(G_i)}$ respectivement, $||.||_{\Omega,(\Omega_i)}$ est alors définie par :

$$||u||_{\Omega,(\Omega_i)} = \Omega(u^G_{(\Omega_i)}) = \Omega(\begin{bmatrix} \Omega_1(u|_{G_1}) \\ \vdots \\ \Omega_k(u|_{G_k}) \end{bmatrix}).$$

Proposition 1. *Si $G = (G_i)_{1 \leq i \leq k}$ est une partition de $\{1, ..., n\}$, et si $\Omega : \mathbb{R}^k \to \mathbb{R}$ et, pour $i \in \{1, ..., k\}$, $\Omega_i : \mathbb{R}^{\chi(G_i)} \to \mathbb{R}$ sont des applications qui vérifient :*

(i) Ω est croissante pour chacun de ses arguments sur le domaine $(\mathbb{R}^+)^n$,

(ii) $\forall i \in \{1, ..., k\}$, Ω_i est une norme,

alors $||.||_{\Omega,(\Omega_i)}$ est une norme.

Démonstration. $||.||_{\Omega,(\Omega_i)}$ doit vérifier les propriétés suivantes :

(i) $\forall u \in \mathbb{R}^n, \forall \lambda \in \mathbb{R}, \quad ||\lambda u||_{\Omega,(\Omega_i)} = |\lambda| \, ||u||_{\Omega,(\Omega_i)}$,

(ii) $\forall u \in \mathbb{R}^n, \forall v \in \mathbb{R}^n, \quad ||u + v||_{\Omega,(\Omega_i)} \leq ||u||_{\Omega,(\Omega_i)} + ||v||_{\Omega,(\Omega_i)}$,

(iii) si $||u||_{\Omega,(\Omega_i)} = 0$ alors $u = 0$.

(i) : Soit $u \in \mathbb{R}^n$ et $\lambda \in \mathbb{R}$,

$$||\lambda u||_{\Omega,(\Omega_i)} = \Omega((\lambda u)^G_{(\Omega_i)}).$$

Nous avons

$$(\lambda u)^G_{(\Omega_i)} = \begin{bmatrix} \Omega_1(\lambda u|_{G_1}) \\ \vdots \\ \Omega_k(\lambda u|_{G_k}) \end{bmatrix} = \begin{bmatrix} |\lambda|\Omega_1(u|_{G_1}) \\ \vdots \\ |\lambda|\Omega_k(u|_{G_k}) \end{bmatrix} = |\lambda| u^G_{(\Omega_i)} \,,$$

donc,

$$\begin{aligned} ||\lambda u||_{\Omega,(\Omega_i)} &= \Omega((\lambda u)^G_{(\Omega_i)}) \\ &= \Omega(|\lambda| u^G_{(\Omega_i)}) \\ &= |\lambda| \Omega(u^G_{(\Omega_i)}) \\ &= |\lambda| \, ||u||_{\Omega,(\Omega_i)} \,. \end{aligned}$$

(ii) : Soit $u \in \mathbb{R}^n$ et $v \in \mathbb{R}^n$,

$$||u+v||_{\Omega,(\Omega_i)} = \Omega((u+v)^G_{(\Omega_i)}) \,.$$

Nous avons

$$(u+v)^G_{(\Omega_i)} = \begin{bmatrix} \Omega_1(u|_{G_1} + v|_{G_1}) \\ \vdots \\ \Omega_k(u|_{G_k} + v|_{G_k}) \end{bmatrix} ,$$

et

$$\begin{bmatrix} \Omega_1(u|_{G_1} + v|_{G_1}) \\ \vdots \\ \Omega_k(u|_{G_k} + v|_{G_k}) \end{bmatrix} \preceq \begin{bmatrix} \Omega_1(u|_{G_1}) + \Omega_1(v|_{G_1}) \\ \vdots \\ \Omega_k(u|_{G_k}) + \Omega_k(v|_{G_k}) \end{bmatrix} ,$$

et

$$\begin{bmatrix} \Omega_1(u|_{G_1}) + \Omega_1(v|_{G_1}) \\ \vdots \\ \Omega_k(u|_{G_k}) + \Omega_k(v|_{G_k}) \end{bmatrix} = \begin{bmatrix} \Omega_1(u|_{G_1}) \\ \vdots \\ \Omega_k(u|_{G_k}) \end{bmatrix} + \begin{bmatrix} \Omega_1(v|_{G_1}) \\ \vdots \\ \Omega_k(v|_{G_k}) \end{bmatrix} = u^G_{(\Omega_i)} + v^G_{(\Omega_i)} \,,$$

ce qui amène à

$$(u+v)^G_{(\Omega_i)} \preceq u^G_{(\Omega_i)} + v^G_{(\Omega_i)} \,,$$

où $\preceq$ correspond à une comparaison coordonnée par coordonnée de deux vecteurs. Puisque nous supposons Ω croissante vis-à-vis de chacun de ses arguments sur le domaine $(\mathbb{R}^+)^n$, il en découle que

$$\Omega((u+v)^G_{(\Omega_i)}) \leq \Omega(u^G_{(\Omega_i)} + v^G_{(\Omega_i)}) \,.$$

Mais nous avons aussi

$$\Omega(u^G_{(\Omega_i)} + v^G_{(\Omega_i)}) \leq \Omega(u^G_{(\Omega_i)}) + \Omega(v^G_{(\Omega_i)}) \,,$$

et par conséquent,

$$\Omega((u+v)^G_{(\Omega_i)}) \leq \Omega(u^G_{(\Omega_i)}) + \Omega(v^G_{(\Omega_i)}) \,,$$

ce qui peut s'écrire comme

$$||u+v||_{\Omega,(\Omega_i)} \leq ||u||_{\Omega,(\Omega_i)} + ||v||_{\Omega,(\Omega_i)}\,.$$

(iii) : Soit $u \in \mathbb{R}^n$ tel que $||u||_{\Omega,(\Omega_i)} = 0$, nous avons

$$\begin{aligned}\Omega(u^G_{(\Omega_i)}) = 0 &\implies u^G_{(\Omega_i)} = 0\\ &\implies \forall i \in \{1,...,k\},\ \Omega_i(u|_{G_i}) = 0\\ &\implies \forall i \in \{1,...,k\},\ u|_{G_i} = 0\,.\end{aligned}$$

Donc,

$$||u||_{\Omega,(\Omega_i)} = 0 \implies u = 0\,.$$

□

B.2 Normes duales

Étant donnée une norme Ω sur $\mathbb{R}^n$, sa norme duale Ω^* est définie par :

$$\Omega^*(u) = \max_{v/\Omega(v)\leq 1} u^\mathsf{T} v\,.$$

Proposition 2. *Soit Ω une norme sur $\mathbb{R}^n$, alors :*

$$\forall u \in \mathbb{R}^n, \forall v \in \mathbb{R}^n,\ u^\mathsf{T} v \leq \Omega^*(u)\Omega(v)\,.$$

Démonstration. Soit $u \in \mathbb{R}^n$ et $v \in \mathbb{R}^n$, nous supposons que $v \neq 0$ (ce cas particulier est évident). Ensuite, $\Omega(\frac{v}{\Omega(v)}) = 1$ et donc

$$u^\mathsf{T} \frac{v}{\Omega(v)} \leq \max_{v/\Omega(v)\leq 1} u^\mathsf{T} v\,.$$

Par conséquent, $u^\mathsf{T} \frac{v}{\Omega(v)} \leq \Omega^*(u)$ et finalement

$$u^\mathsf{T} v \leq \Omega^*(u)\Omega(v)\,.$$

□

Proposition 3. *Soit Ω une norme sur $\mathbb{R}^n$ et $u \in \mathbb{R}^n$, alors :*

$$\exists v \in \mathbb{R}^n :\ u^\mathsf{T} v = \Omega^*(u) \text{ et } \Omega(v) = 1\,.$$

Démonstration. Considérons $B_\Omega = \{v/\Omega(v) \leq 1\}$ et $u \in \mathbb{R}^n$, nous avons

$$\Omega^*(u) = \max_{v\in B_\Omega} u^\mathsf{T} v = \max_{v\in B_\Omega} f_u(v)\,,$$

où $f_u(v) = u^\mathsf{T} v$ est une fonction continue vis-à-vis de v. Du fait de la dimension finie de E et de l'équivalence des normes sur E, B_Ω est fermé et borné, et est donc un ensemble compact. Il en découle que f_u atteint son maximum sur B_Ω, ce qui mène à

$$\exists w \in B_\Omega :\ u^\mathsf{T} w = \max_{v\in B_\Omega} u^\mathsf{T} v\,.$$

Nous montrons maintenant que w peut être choisi avec la contrainte $\Omega(w) = 1$. Tout d'abord, si $u = 0$ alors la fonction f_u est constante et n'importe quel $w \in B_\Omega$ avec $\Omega(w) = 1$ maximise f_u sur B_Ω.
Nous supposons que $u \neq 0$, et considérons $w \in B_\Omega$ qui maximise f_u sur B_Ω. Alors $u^\mathsf{T} w \geq u^\mathsf{T} \frac{u}{\Omega(u)} > 0$. De plus, $\frac{w}{\Omega(w)} \in B_\Omega$ implique que

$$u^\mathsf{T} w \geq u^\mathsf{T} \frac{w}{\Omega(w)} ,$$

et donc $\Omega(w) \geq 1$ ce qui mène à $\Omega(w) = 1$. □

Proposition 4. *Si $G = (G_i)_{1 \leq i \leq k}$ est une partition de $\{1, ..., n\}$ et si Ω, $(\Omega_i)_{1 \leq i \leq k}$ sont des normes choisies telles que*
(i) $||.||_{\Omega,(\Omega_i)}$ est une norme,
(ii) Ω est invariante en valeur absolue pour chacune de ses coordonnées,
alors la norme duale de $||.||_{\Omega,(\Omega_i)}$ vérifie

$$||.||^*_{\Omega,(\Omega_i)} = ||.||_{\Omega^*,(\Omega_i^*)} .$$

Démonstration. Soit $u \in \mathbb{R}^n$, nous avons

$$\begin{aligned} ||u||^*_{\Omega,(\Omega_i)} &= \max_{v/||v||_{\Omega,(\Omega_i)} \leq 1} u^\mathsf{T} v \\ &= \max_{v/||v||_{\Omega,(\Omega_i)} \leq 1} \sum_{1 \leq i \leq k} (u|_{G_i})^\mathsf{T} v|_{G_i} . \end{aligned}$$

En utilisant la Proposition 2, nous avons pour $1 \leq i \leq k$,

$$(u|_{G_i})^\mathsf{T} v|_{G_i} \leq \Omega_i^*(u|_{G_i}) \Omega_i(v|_{G_i}) .$$

Donc,

$$\begin{aligned} ||u||^*_{\Omega,(\Omega_i)} &= \max_{v/||v||_{\Omega,(\Omega_i)} \leq 1} \sum_{1 \leq i \leq k} (u|_{G_i})^\mathsf{T} v|_{G_i} \\ &\leq \max_{v/||v||_{\Omega,(\Omega_i)} \leq 1} \sum_{1 \leq i \leq k} \Omega_i^*(u|_{G_i}) \Omega_i(v|_{G_i}) . \end{aligned}$$

Nous pouvons écrire

$$\begin{aligned} \max_{v/||v||_{\Omega,(\Omega_i)} \leq 1} \sum_{1 \leq i \leq k} \Omega_i^*(u|_{G_i}) \Omega_i(v|_{G_i}) &= \max_{v/\Omega(v^G_{(\Omega_i)}) \leq 1} (u^G_{(\Omega_i^*)})^\mathsf{T} v^G_{(\Omega_i)} \\ &\leq \max_{w/\Omega(w) \leq 1} (u^G_{(\Omega_i^*)})^\mathsf{T} w . \end{aligned}$$

Comme $\max_{w/\Omega(w) \leq 1} (u^G_{(\Omega_i^*)})^\mathsf{T} w = \Omega^*(u^G_{(\Omega_i^*)})$, nous avons finalement

$$||u||^*_{\Omega,(\Omega_i)} \leq \Omega^*(u^G_{(\Omega_i^*)}) .$$

En utilisant la Proposition 3, il existe $\mu \in \mathbb{R}^k$ tel que

$$\Omega^*(u^G_{(\Omega_i^*)}) = (u^G_{(\Omega_i^*)})^\mathsf{T} \mu \text{ et } \Omega(\mu) = 1 .$$

Pour $i \in \{1, ..., k\}$, il existe $w|_{G_i} \in \mathbb{R}^{\chi(G_i)}$ tel que

$$\Omega_i^*(u|_{G_i}) = (u|_{G_i})^\mathsf{T} w|_{G_i} \text{ et } \Omega_i(w|_{G_i}) = 1\,.$$

Nous considérons le vecteur $v \in \mathbb{R}^n$ défini par

$$\forall i \in \{1, ..., k\}, v|_{G_i} = \mu_i w|_{G_i}\,.$$

Alors nous avons

$$||v||_{\Omega,(\Omega_i)} = \Omega(v^G_{(\Omega_i)}) = \Omega(\begin{bmatrix} \Omega_1(\mu_1 w|_{G_1}) \\ \vdots \\ \Omega_k(\mu_k w|_{G_k}) \end{bmatrix}) = \Omega(\begin{bmatrix} |\mu_1|\Omega_1(w|_{G_1}) \\ \vdots \\ |\mu_k|\Omega_k(w|_{G_k}) \end{bmatrix}) = \Omega(\begin{bmatrix} |\mu_1| \\ \vdots \\ |\mu_k| \end{bmatrix})\,,$$

et puisque Ω est invariante en valeur absolue pour chacune de ses coordonnées,

$$||v||_{\Omega,(\Omega_i)} = \Omega(\begin{bmatrix} |\mu_1| \\ \vdots \\ |\mu_k| \end{bmatrix}) = \Omega(\mu) = 1\,.$$

De plus,

$$u^\mathsf{T} v = \sum_{1 \leq i \leq k} (u|_{G_i})^\mathsf{T} (v|_{G_i}) = \sum_{1 \leq i \leq k} \mu_i (u|_{G_i})^\mathsf{T} (w|_{G_i}) = \sum_{1 \leq i \leq k} \mu_i \Omega^*_{G_i}(u|_{G_i})\,,$$

ce qui mène à

$$u^\mathsf{T} v = \sum_{1 \leq i \leq k} \mu_i \Omega^*_{G_i}(u|_{G_i}) = \mu^\mathsf{T}(u^G_{(\Omega_i^*)}) = \Omega^*(u^G_{(\Omega_i^*)})\,.$$

Donc,

$$||u||^*_{\Omega,(\Omega_i)} = \max_{v/||v||_{\Omega,(\Omega_i)} \leq 1} u^\mathsf{T} v \geq \Omega^*(u^G_{(\Omega_i^*)})\,,$$

et finalement,

$$||u||^*_{\Omega,(\Omega_i)} = \Omega^*(u^G_{(\Omega_i^*)}) = ||u||_{\Omega^*,(\Omega_i^*)}\,.$$

□

B.3 Application au cas de la norme $l_{\infty,1}$ pondérée

Les résultats précédents sont utilisés ici dans le cas spécifique d'une norme $l_{\infty,1}$ pondérée.

Proposition 5. *Soit Ω une norme sur $\mathbb{R}^n$ et $\lambda \in \mathbb{R}^{+*}$, alors*

$$(\lambda\Omega)^* = \frac{1}{\lambda}\Omega^*\,.$$

Démonstration. Soit $u \in \mathbb{R}^n$,

$$\begin{aligned}(\lambda\Omega)^*(u) = \frac{1}{\lambda}\Omega^* &= \max_{v/\lambda\Omega(v)\leq 1} u^\mathsf{T} v \\ &= \max_{v/\Omega(\lambda v)\leq 1} \frac{1}{\lambda} u^\mathsf{T} \lambda v \\ &\underset{w=\lambda v}{=} \frac{1}{\lambda} \max_{w/\Omega(w)\leq 1} u^\mathsf{T} w \\ &= \frac{1}{\lambda}\Omega^*(u)\,.\end{aligned}$$

□

Proposition 6. *Soit $(w_i)_{1\leq i\leq k} \in (\mathbb{R}^{+*})^k$ des poids strictement positifs et $G = (G_i)_{1\leq i\leq k}$ une partition de $\{1, ..., n\}$. La fonction $l^w_{\infty,1}$, ou $||.||^w_{\infty,1}$, est définie par*

$$\forall u \in \mathbb{R}^n,\ ||u||^w_{\infty,1} = \max_{1\leq i\leq k} w_i ||u|_{G_i}||_1\,,$$

tandis que la fonction $l^{\frac{1}{w}}_{1,\infty}$, ou $||.||^{\frac{1}{w}}_{1,\infty}$, est définie par

$$\forall u \in \mathbb{R}^n,\ ||u||^{\frac{1}{w}}_{1,\infty} = \sum_{1\leq i\leq k} \frac{1}{w_i} ||u|_{G_i}||_\infty\,.$$

Alors $||.||^w_{\infty,1}$ est une norme sur $\mathbb{R}^n$ et sa norme duale vérifie

$$||.||^w_{\infty,1}{}^* = ||.||^{\frac{1}{w}}_{1,\infty}\,.$$

Démonstration. Nous avons

$$||.||^w_{\infty,1} = ||.||_{l_\infty,(w_i l_1)}\,.$$

La norme l_∞ est croissante pour chacun de ses arguments sur le domaine $(\mathbb{R}^+)^n$ et pour chaque $i \in \{1, ..., k\}$, $w_i l_1$ est une norme valide. En conséquence, la Proposition 1 indique que $||.||^w_{\infty,1}$ est une norme.

Puisque la norme l_∞ est invariante en valeur absolue vis-à-vis de chacune de ses coordonnées, la Proposition 4 peut être appliquée et donne

$$||.||^*_{l_\infty,(w_i l_1)} = ||.||_{l^*_\infty,((w_i l_1)^*)} = ||.||_{l^*_\infty,(\frac{1}{w_i} l^*_1)}\,.$$

Les normes duales des normes l_1 et l_∞ sont données par $l^*_1 = l_\infty$ et $l^*_\infty = l_1$ [101], ce qui mène à

$$||.||^*_{l_\infty,(w_i l_1)} = ||.||_{l^*_\infty,(\frac{1}{w_i} l^*_1)} = ||.||_{l_1,(\frac{1}{w_i} l_\infty)}\,.$$

Donc,

$$||.||^w_{\infty,1}{}^* = ||.||^{\frac{1}{w}}_{1,\infty}\,.$$

□

C. Expérimentations avec jeux de détections simulés

L'ensemble des résultats obtenus en simulant des jeux de détections, comme décrit en sous-section V.4.2, est détaillé ici pour notre approche **DSR** en utilisant ou non des dictionnaires denses.

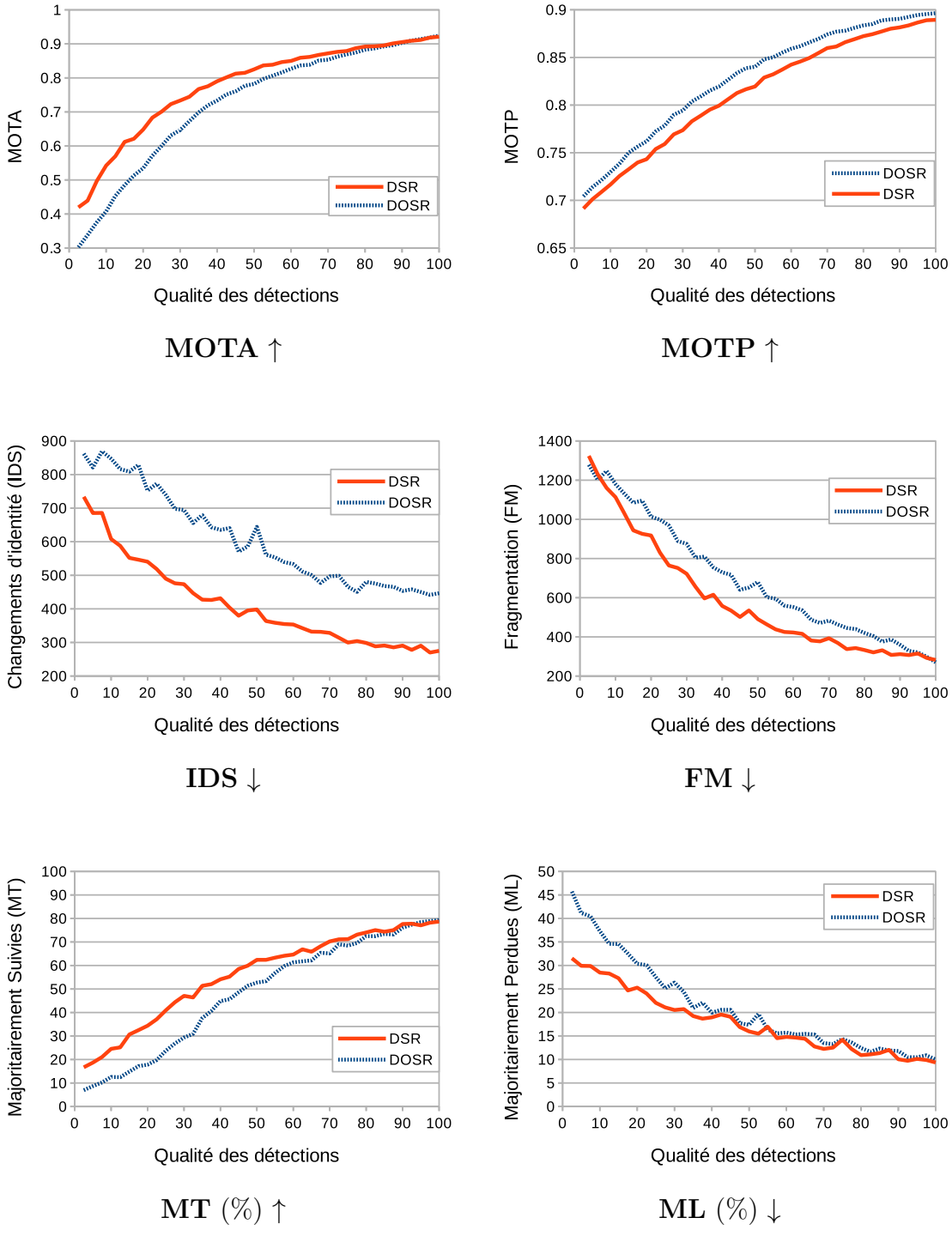


FIGURE VI.1 – Résultats de notre approches **DSR** avec ou sans dictionnaires denses (dénommée dans ce cas **DOSR**) avec des jeux de détections simulés.

Bibliographie

[1] M. Andriluka, S. Roth, and B. Schiele. People-tracking-by-detection and people-detection-by-tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[2] M. Andriluka, S. Roth, and B. Schiele. Monocular 3D pose estimation and tracking by detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[3] S. Avidan. Support vector tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2004.

[4] B. Babenko, M.-H. Yang, and S. Belongie. Visual tracking with online multiple instance learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[5] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. Optimization with sparsity-inducing penalties. *Foundations and Trends in Machine Learning*, 2012.

[6] S.-H. Bae and K.-J. Yoon. Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[7] Y. Ban, S. Ba, X. Alameda-Pineda, and R. Horaud. Tracking multiple persons based on a variational bayesian model. In *European Conference on Computer Vision (ECCV) Workshops*, 2016.

[8] C. Bao, Y. Wu, H. Ling, and H. Ji. Real time robust l1 tracker using accelerated proximal gradient approach. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[9] Y. Bar-Shalom and E. Tse. Tracking in a cluttered environment with probabilistic data association. *Automatica*, 1975.

[10] H. Bay, T. Tuytelaars, and L. Van Gool. Surf : Speeded up robust features. In *European Conference on Computer Vision (ECCV)*, 2006.

[11] A. Beck and M. Teboulle. Gradient-based algorithms with applications to signal recovery problems. In *Convex Optimization in Signal Processing and Communications*, 2010.

[12] B. Benfold and I. Reid. Guiding visual surveillance by tracking human attention. In *British Machine Vision Conference (BMVC)*, 2009.

[13] B. Benfold and I. Reid. Stable multi-target tracking in real-time surveillance video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[14] K. Bernardin and R. Stiefelhagen. Evaluating Multiple Object Tracking Performance : The CLEAR MOT Metrics. *EURASIP J. Image and Video Processing*, 2008.

[15] A. Bewley, L. Ott, F. Ramos, and B. Upcroft. ALExTRAC : Affinity Learning by Exploring Temporal Reinforcement within Association Chains. In *ICRA*, 2016.

[16] Y. Boureau, N. Le Roux, F. Bach, J. Ponce, and Y. LeCun. Ask the locals : multi-way local pooling for image recognition. In *International Conference on Computer Vision (ICCV)*, 2011.

[17] G. R. Bradski. Real time face and object tracking as a component of a perceptual user interface. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 1998.

[18] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. V. Gool. Robust tracking-by-detection using a detector confidence particle filter. In *International Conference on Computer Vision (ICCV)*, 2009.

[19] W. Brendel, M. R. Amer, and S. Todorovic. Multiobject tracking as maximum weight independent set. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[20] H. Bristow, A. Eriksson, and S. Lucey. Fast convolutional sparse coding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[21] P. Brémaud. Markov chains : Gibbs fields, monte carlo simulation and queues. *Texts in applied mathematics*, 1999.

[22] F. Campbell and G. Allen. Within group variable selection through the exclusive lasso. In *arXiv :1505.07517*, 2015.

[23] R. Chalasani, J. C. Principe, and N. Ramakrishnan. A fast proximal method for convolutional sparse coding. In *International Joint Conference on Neural Networks*, 2013.

[24] W. Choi. Near-online multi-target tracking with aggregated local flow descriptor. In *International Conference on Computer Vision (ICCV)*, 2015.

[25] C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 1995.

[26] I. J. Cox. A review of statistical data association techniques for motion correspondence. *International Journal of Computer Vision (IJCV)*, 1993.

[27] F. C. Crow. Summed-area tables for texture mapping. *ACM SIGGRAPH computer graphics*, 1984.

[28] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[29] N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. In *European Conference on Computer Vision (ECCV)*, 2006.

[30] M. Danelljan, A. Robinson, F. Shahbaz Khan, and M. Felsberg. Beyond correlation filters : Learning continuous convolution operators for visual tracking. In *European Conference on Computer Vision (ECCV)*, 2016.

[31] A. Dehghan, S. M. Assari, and M. Shah. GMMCP-tracker : Globally optimal generalized maximum multi clique problem for multiple object tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[32] C. Dicle, M. Sznaier, and O. Camps. The way they move : Tracking targets with similar appearance. In *International Conference on Computer Vision (ICCV)*, 2013.

[33] P. Dollar, R. Appel, S. Belongie, and P. Perona. Fast feature pyramids for object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2014.

[34] D. H. Douglas and T. K. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica : The International Journal for Geographic Information and Geovisualization*, 1973.

[35] A. Ess, B. Leibe, and L. V. Gool. Depth and appearance for mobile scene analysis. In *International Conference on Computer Vision (ICCV)*, 2007.

[36] L. Fagot-Bouquet, R. Audigier, Y. Dhome, and F. Lerasle. Online multi-person tracking based on global sparse collaborative representations. In *IEEE International Conference on Image Processing (ICIP)*, 2015.

[37] L. Fagot-Bouquet, R. Audigier, Y. Dhome, and F. Lerasle. Improving multi-frame data association with sparse representations for robust near-online multi-object tracking. In *European Conference on Computer Vision (ECCV)*, 2016.

[38] L. Fagot-Bouquet, R. Audigier, Y. Dhome, and F. Lerasle. Collaboration and spatialization for an efficient multi-person tracking via sparse representations. In *IEEE Advanced Video and Signal-based Surveillance (AVSS)*, 2015.

[39] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2010.

[40] J. Ferryman and A. Shahrokni. Pets2009 : Dataset and challenge. In *Performance evaluation of tracking and surveillance (PETS-Winter)*, 2009.

[41] T. Fortmann, Y. Bar-Shalom, and M. Scheffe. Sonar tracking of multiple targets using joint probabilistic data association. *Journal of Oceanic Engineering*, 1983.

[42] S. Gao, I. W.-H. Tsang, and L.-T. Chia. Kernel sparse representation for image classification and face recognition. In *European Conference on Computer Vision (ECCV)*, 2010.

[43] W. Ge and R. T. Collins. Multi-target data association by tracklets with unsupervised parameter estimation. In *British Machine Vision Conference (BMVC)*, 2008.

[44] A. Geiger, M. Lauer, C. Wojek, C. Stiller, and R. Urtasun. 3D traffic scene understanding from movable platforms. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2014.

[45] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving ? The KITTI vision benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[46] S. Hare, A. Saffari, and P. H. Torr. Struck : Structured output tracking with kernels. In *International Conference on Computer Vision (ICCV)*, 2011.

[47] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey Vision Conference*, 1988.

[48] F. Heide, W. Heidrich, and G. Wetzstein. Fast and flexible convolutional sparse coding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[49] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista. High-speed tracking with kernelized correlation filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2015.

[50] Z. Hong, X. Mei, D. Prokhorov, and D. Tao. Tracking via robust multi-task multi-view joint sparse representation. In *International Conference on Computer Vision (ICCV)*, 2013.

[51] J. Hong Yoon, C.-R. Lee, M.-H. Yang, and K.-J. Yoon. Online multi-object tracking via structural constraint event aggregation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[52] F. Hutter, H. H. Hoos, and K. Leyton-Brown. Sequential model-based optimization for general algorithm configuration. In *LION*, 2011.

[53] M. Isard and A. Blake. Condensation—conditional density propagation for visual tracking. *International Journal of Computer Vision (IJCV)*, 1998.

[54] M. Isard and A. Blake. A smoothing filter for condensation. In *European Conference on Computer Vision (ECCV)*, 1998.

[55] X. Jia, H. Lu, and M. Yang. Visual tracking via adaptive structural local sparse appearance model. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[56] J. Ju, D. Kim, B. Ku, D. Han, and H. Ko. Online multi-person tracking with two-stage data association and online appearance model learning. In *IET Computer Vision*, 2016.

[57] S. J. Julier and J. K. Uhlmann. New extension of the kalman filter to nonlinear systems. In *AeroSense*, 1997.

[58] Z. Kalal, K. Mikolajczyk, and J. Matas. Tracking-learning-detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2012.

[59] R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 1960.

[60] Z. Khan, T. Balch, and F. Dellaert. Mcmc-based particle filtering for tracking a variable number of interacting targets. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2005.

[61] H. Kieritz, S. Becker, W. Hübner, and M. Arens. Online multi-person tracking using integral channel features. In *IEEE Advanced Video and Signal-based Surveillance (AVSS)*, 2016.

[62] C. Kim, F. Li, A. Ciptadi, and J. M. Rehg. Multiple hypothesis tracking revisited. In *International Conference on Computer Vision (ICCV)*, 2015.

[63] D. Kong, R. Fujimaki, J. Liu, F. Nie, and C. Ding. Exclusive feature learning on arbitrary structures via l1,2-norm. In *International Conference on Neural Information Processing Systems (NIPS)*, 2014.

[64] S. Kong and D. Wang. A dictionary learning approach for classification : Separating the particularity and the commonality. In *European Conference on Computer Vision (ECCV)*, 2012.

[65] Kristan, Matej et al. The visual object tracking vot2016 challenge results. In *European Conference on Computer Vision (ECCV) Workshops*, 2016.

[66] N. Le, A. Heili, and J.-M. Odobez. Long-term time-sensitive costs for crf-based tracking by detection. In *European Conference on Computer Vision (ECCV) Workshops*, 2016.

[67] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler. MOTChallenge 2015 : towards a benchmark for multi-target tracking. *arXiv :1504.01942 [cs]*, 2015.

[68] L. Leal-Taixe, C. Canton-Ferrer, and K. Schindler. Learning by tracking : Siamese cnn for robust target association. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2016.

[69] L. Leal-Taixé, M. Fenzi, A. Kuznetsova, B. Rosenhahn, and S. Savarese. Learning an image-based motion context for multiple people tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[70] B. Lee, E. Erdenee, S. Jin, M. Y. Nam, Y. G. Jung, and P. K. Rhee. Multi-class multi-object tracking using changing point detection. In *European Conference on Computer Vision (ECCV) Workshops*, 2016.

[71] B. Leibe, K. Schindler, N. Cornelis, and L. V. Gool. Coupled object detection and tracking from static cameras and moving vehicles. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2008.

[72] J. Lewis. Fast normalized cross-correlation. In *Vision interface*, 1995.

[73] H. Li, C. Shen, and Q. Shi. Real-time visual tracking using compressive sensing. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[74] Y. Li, A. Dore, and J. Orwell. Evaluating the performance of systems for tracking football players and ball. In *IEEE Advanced Video and Signal-based Surveillance (AVSS)*, 2005.

[75] Y. Li, C. Huang, and R. Nevatia. Learning to associate : Hybridboosted multi-target tracker for crowded scene. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[76] B. Liu, J. Huang, L. Yang, and C. Kulikowsk. Robust tracking using local sparse appearance model and k-selection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[77] J. Liu, X. Tong, W. Li, T. Wang, Y. Zhang, and H. Wang. Automatic player detection, labeling and tracking in broadcast soccer video. *Pattern Recognition*, 2009.

[78] D. G. Lowe. Object recognition from local scale-invariant features. In *International Conference on Computer Vision (ICCV)*, 1999.

[79] W. Luo, J. Xing, X. Zhang, X. Zhao, and T.-K. Kim. Multiple object tracking : A literature review. *arXiv :1409.7618*, 2014.

[80] E. Maggio, M. Taj, and A. Cavallaro. Efficient multitarget visual tracking using random finite sets. *Transactions on Circuits and Systems for Video Technology*, 2008.

[81] R. P. S. Mahler. Multitarget bayes filtering via first-order multitarget moments. *Transactions on Aerospace and Electronic Systems*, 2003.

[82] J. Mairal, F. Bach, and J. Ponce. Sparse modeling for image and vision processing. *Found. Trends. Comput. Graph. Vis.*, 2014.

[83] J. Mairal and B. Yu. Supervised feature selection in graphs with path coding penalties and network flows. *Journal of Machine Learning Research (JMLR)*, 2013.

[84] N. McLaughlin, J. Martinez Del Rincon, and P. Miller. Enhancing linear programming with motion modeling for multi-target tracking. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2015.

[85] X. Mei and H. Ling. Robust visual tracking and vehicle classification via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2011.

[86] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, and K. Schindler. MOT16 : a benchmark for multi-object tracking. *arXiv :1603.00831 [cs]*, 2016.

[87] A. Milan, S. H. Rezatofighi, A. Dick, I. Reid, and K. Schindler. Online multi-target tracking using recurrent neural networks. In *AAAI*, 2017.

[88] A. Milan, S. Roth, and K. Schindler. Continuous energy minimization for multitarget tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2014.

[89] A. Milan, K. Schindler, and S. Roth. Multi-target tracking by discrete-continuous energy minimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2016.

[90] A. Milan, L. Leal-Taixé, K. Schindler, and I. Reid. Joint tracking and segmentation of multiple targets. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[91] A. Milan, K. Schindler, and S. Roth. Challenges of ground truth evaluation of multi-target tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2013.

[92] C. Morefield. Application of 0-1 integer programming to multitarget tracking problems. *Transactions on Automatic Control*, 1977.

[93] J. Munkres. Algorithms for the assignment and transportation problems. *Journal of the Society of Industrial and Applied Mathematics*, 1957.

[94] M. A. Naiel, M. O. Ahmad, M. N. S. Swamy, Y. Wu, and M. Yang. Online multi-person tracking via robust collaborative model. In *IEEE International Conference on Image Processing (ICIP)*, 2014.

[95] H. Nam, M. Baek, and B. Han. Modeling and propagating cnns in a tree structure for visual tracking. *CoRR*, 2016.

[96] Y. Nesterov. Introductory lectures on convex optimization : a basic course. *Applied optimization*, 2004.

[97] A. T. Nghiem, F. Brémond, M. Thonnat, and V. Valentin. Etiseo, performance evaluation for video surveillance systems. In *IEEE Advanced Video and Signal-based Surveillance (AVSS)*, 2007.

[98] B. ngu Vo and S. Singh. Sequential monte carlo implementation of the phd filter for multi-target tracking. In *International Conference on Information Fusion*, 2003.

[99] S. Oh, S. J. Russell, and S. Sastry. Markov chain monte carlo data association for multi-target tracking. *Transactions on Automatic Control*, 2009.

[100] T. Ojala, M. Pietikäinen, and D. Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, 1996.

[101] N. Parikh and S. Boyd. Proximal algorithms. In *Foundations and Trends in Optimization*, 2013.

[102] H. Pirsiavash, D. Ramanan, and C. C. Fowlkes. Globally-optimal greedy algorithms for tracking a variable number of objects. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[103] A. Quattoni, X. Carreras, M. Collins, and T. Darrell. An efficient projection for l1,infinity regularization. In *International Conference on Machine Learning (ICML)*, 2009.

[104] U. Ramer. An iterative procedure for the polygonal approximation of plane curves. *Computer Graphics and Image Processing*, 1972.

[105] H. E. Rauch, C. Striebel, and F. Tung. Maximum likelihood estimates of linear dynamic systems. *AIAA journal*, 1965.

[106] D. B. Reid. An algorithm for tracking multiple targets. *Transactions on Automatic Control*, 1979.

[107] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn : Towards real-time object detection with region proposal networks. In *International Conference on Neural Information Processing Systems (NIPS)*, 2015.

[108] S. H. Rezatofighi, A. Milan, Z. Zhang, Q. Shi, A. R. Dick, and I. D. Reid. Joint probabilistic data association revisited. In *International Conference on Computer Vision (ICCV)*, 2015.

[109] R. Rigamonti, M. A. Brown, and V. Lepetit. Are sparse representations really relevant for image classification ? In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[110] M. J. Roshtkhari and M. D. Levine. Multiple object tracking using local motion patterns. In *British Machine Vision Conference (BMVC)*, 2014.

[111] R. Sanchez-Matilla, F. Poiesi, and A. Cavallaro. Online multi-target tracking with strong and weak detections. In *European Conference on Computer Vision (ECCV) Workshops*, 2016.

[112] G. Shu, A. Dehghan, O. Oreifej, E. Hand, and M. Shah. Part-based multiple-person tracking with partial occlusion handling. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[113] A. W. M. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah. Visual tracking : An experimental survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2014.

[114] K. Smith, D. Gatica-Perez, J.-M. Odobez, and S. Ba. Evaluating multi-object tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2005.

[115] F. Solera, S. Calderara, and R. Cucchiara. Towards the evaluation of reproducible robustness in tracking-by-detection. In *IEEE Advanced Video and Signal-based Surveillance (AVSS)*, 2015.

[116] Y. Song and M. Jeon. Online multiple object tracking with the hierarchically adopted gm-phd filter using motion and appearance. In *The International Conference on Consumer Electronics Asia*, 2016.

[117] S. Tang, B. Andres, M. Andriluka, and B. Schiele. Multi-person tracking by multicuts and deep matching. In *European Conference on Computer Vision (ECCV) Workshops*, 2016.

[118] S. Tang, M. Andriluka, and B. Schiele. Detection and tracking of occluded people. *International Journal of Computer Vision (IJCV)*, 2014.

[119] P. Viola and M. Jones. Robust real-time object detection. *International Journal of Computer Vision (IJCV)*, 2001.

[120] B. Wang, G. Wang, K. L. Chan, and L. Wang. Tracklet association by online target-specific metric learning and coherent dynamics estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2016.

[121] B. Wang, L. Wang, B. Shuai, Z. Zuo, T. Liu, K. Luk Chan, and G. Wang. Joint learning of convolutional neural networks and temporally constrained metrics for tracklet association. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2016.

[122] S. Wang and C. C. Fowlkes. Learning optimal parameters for multi-target tracking. In *British Machine Vision Conference (BMVC)*, 2015.

[123] X. Wang, E. Turetken, F. Fleuret, and P. Fua. Tracking interacting objects optimally using integer programming. In *European Conference on Computer Vision (ECCV)*, 2014.

[124] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2009.

[125] B. Wu and R. Nevatia. Tracking of multiple, partially occluded humans based on static body part detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

[126] Z. Wu, J. Zhang, and M. Betke. Online motion agreement tracking. In *British Machine Vision Conference (BMVC)*, 2013.

[127] Y. Xiang, A. Alahi, and S. Savarese. Learning to track : Online multi-object tracking by decision making. In *International Conference on Computer Vision (ICCV)*, 2015.

[128] M. Yang and Y. Jia. Temporal dynamic appearance modeling for online multi-person tracking. *Computer Vision and Image Understanding (CVIU)*, 2016.

[129] A. Yilmaz, O. Javed, and M. Shah. Object tracking : A survey. *ACM Comput. Surv.*, 2006.

[130] J. H. Yoon, M.-H. Yang, J. Lim, and K.-J. Yoon. Bayesian multi-object tracking using motion context from multiple objects. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2015.

[131] F. Yu, W. Li, Q. Li, Y. Liu, X. Shi, and J. Yan. Poi : Multiple object tracking with high performance detection and appearance feature. In *European Conference on Computer Vision (ECCV)*, 2016.

[132] Q. Yu and G. Medioni. Multiple-target tracking by spatiotemporal monte carlo markov chain data association. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2009.

[133] A. R. Zamir, A. Dehghan, and M. Shah. GMCP-tracker : Global multi-object tracking using generalized minimum clique graphs. In *European Conference on Computer Vision (ECCV)*, 2012.

[134] J. Zhang, L. L. Presti, and S. Sclaroff. Online multi-person tracking by tracker hierarchy. In *IEEE Advanced Video and Signal-based Surveillance (AVSS)*, 2012.

[135] S. Zhang, H. Yao, X. Sun, and X. Lu. Sparse coding based visual tracking : Review and experimental comparison. *Pattern Recognition*, 2013.

[136] T. Zhang, A. Bibi, and B. Ghanem. In defense of sparse tracking : Circulant sparse tracker. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[137] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja. Robust visual tracking via structured multi-task sparse learning. *International Journal of Computer Vision (IJCV)*, 2013.

[138] T. Zhang, S. Liu, C. Xu, S. Yan, B. Ghanem, N. Ahuja, and M.-H. Yang. Structural sparse tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[139] Y. Zhang, D.-Y. Yeung, and Q. Xu. Probabilistic multi-task feature selection. In *International Conference on Neural Information Processing Systems (NIPS)*, 2010.

[140] W. Zhong, H. Lu, and M. Yang. Robust object tracking via sparsity-based collaborative model. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[141] Y. Zhou, R. Jin, and S. Hoi. Exclusive lasso for multi-task feature selection. *Journal of Machine Learning Research (JMLR)*, 2010.

Representations for tracking : Exploiting sparse representations for multi-object tracking

ABSTRACT :

Despite recent advances in object detection, multi-object tracking still raises some specific issues and therefore remains a challenging problem. In this thesis, we propose to investigate the use of sparse representations within multi-object tracking approaches in order to gain in performances.

The first contribution of this thesis consists in designing an online tracking approach that takes advantage of collaborative sparse representations to better distinguish between the targets. Then, structured sparse representations are considered in order to be more suited to traking approaches based on a sliding window. In order to rely less on the object detector quality, we consider for the last contribution of this thesis to use dense dictionaries that are taking into account a large number of undetected locations inside each frame.

KEYWORDS : Multi-object tracking, tracking by detection, sparse representations.

Représenter pour suivre : Exploitation de représentations parcimonieuses pour le suivi multi-objets

AUTEUR : Loïc Pierre FAGOT-BOUQUET.

DIRECTEURS DE THÈSE : Frédéric LERASLE, Romaric AUDIGIER (co-directeur).

LIEU ET DATE DE SOUTENANCE : Thèse soutenue le 20 mars 2017 au centre d'intégration Nano-Innov du CEA LIST à Palaiseau.

RÉSUMÉ :

Le suivi multi-objets, malgré les avancées récentes en détection d'objets, présente encore plusieurs difficultés spécifiques et reste ainsi une problématique difficile. Au cours de cette thèse nous proposons d'examiner l'emploi de représentations parcimonieuses au sein de méthodes de suivi multi-objets, dans le but d'améliorer les performances de ces dernières.

La première contribution de cette thèse consiste à employer des représentations parcimonieuses collaboratives dans un système de suivi en ligne pour distinguer au mieux les cibles. Des représentations parcimonieuses structurées sont ensuite considérées pour s'adapter plus spécifiquement aux approches de suivi à fenêtre glissante. Une dernière contribution consiste à employer des dictionnaires denses, prenant en considération un grand nombre de positions non détectées au sein des images, de manière à être plus robuste vis-à-vis de la performance du détecteur d'objets employé.

MOTS-CLÉS : suivi visuel multi-objets, suivi par détection, représentations parcimonieuses.

ÉCOLE DOCTORALE ET SPÉCIALITÉ : EDSYS, Automatique 4200046.

UNITÉ DE RECHERCHE : UPR 8001 LAAS Laboratoire d'Analyse et d'Architecture des Systèmes.