République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Kasdi Merbah Ouargla
Faculté des Sciences Appliquées
Département de Génie électrique

THÈSE

Présentée pour l'obtention du diplôme de
DOCTORAT en SCIENCES
Spécialité : Génie électrique

Par : DJARAH Djalal

Thème

ESTIMATION DE LA DYNAMIQUE DES ROBOTS MOBILES EN MOUVEMENT

Soutenu le .../…/……devant le jury composé de :

Dr. SAMAI Djamel	PRESIDENT	MCA	Univ Ouargla
Pr. LOUAER Med Redha	EXAMINATEUR	PROF	Univ Tébessa
Dr. BENCHAABANE Abderrazak	EXAMINATEUR	MCA	Univ Ouargla
Dr. AMIEUR Toufik	EXAMINATEUR	MCA	Univ Tébessa
Pr. MERAOUMIA Abdellah	DIRECTEUR	PROF	Univ Tébessa
Dr. LOUAZEN Med Lakhdar	CO- DIRECTEUR	MCA	Univ Ouargla

Année 2021/2022

Estimation de la dynamiques des robots mobiles en mouvement

Djalal DJARAH
Université de Ouargla, Département de Génie Électrique, Faculté des Sciences Appliquées, Ouargla, Algérie.
Laboratoire LAGE de l'université de Ouargla.
E-mail: d.djarah@gmail.com

LISTE DES TRAVAUX

Revues Internationales

- **D. Djalal;** A. Meraoumia, Abdallah and M. L. Louazene, "The Impact of the Detector on the Performances of a Multi Person Tracking System", **:** Recent Patents on Engineering, Volume 16, Number 2, pp. 100-108(9), 2021.
- A. Benmakhlouf, A. Louchene, and **D. Djarah**, "Fuzzy Logic and Modified Crisp Logic Applied to a Motor Position Control", Journal Mechatronic Systems and Control (formerly Control and Intelligent Systems), Volume 38, Number 3, 2010.

Conférences Internationales

- **D. Djarah**, A. Benmakhlouf and G. Zidani, " Application of Neural Networks in Perception System Management for an Indoor Mobile Robot," The 5nd International Conference on Green Energy and Environmental Engineering GEEE-2018, Sousse, Tunisie, 28-30, Avril, 2018.
- G. Zidani, **D. Djarah** and A. Benmakhlouf, " Tracking Control of Wheeled Mobile Robot through Neural Networks," The 5nd International Conference on Green Energy and Environmental Engineering GEEE-2018, Sousse, Tunisie, 28-30, Avril, 2018.
- G. Zidani, A. Louchene, A. Benmakhlouf and **D. Djarah**, "Exécution de trajectoire pour robot mobile par réseaux de neurones," The 2nd International Conference on Electronics and Oil : FromTheory to Application ICEO'13, Ouargla, Algérie, 05-06, Mars, 2013.
- F. Kadri, S. Drid, **D. Djarah**, and F. Djeffal, "Direct Torque Control of Induction Motor Fed by Three Phase PWM Inverter Using Fuzzy logic and Neural Network," 6th International Conference on Electrical Engineering, CEE'10. 2010.
- F. Kadri, **D. Djarah** and S. Drid, "Neural Network Direct Torque Control of Induction Motor Fed by Three Phase PWM Inverter", Conference: First International Congress on Models, Optimization and Security of Systems, ICMOSS'2010.
- A. Benmakhlouf, **D. Djarah** and G. Zidani, "A new fuzzy path following controller for a mobile robot," International Conference on Systems and Information Processing ICSIP'09, Guelma, Algérie, 2-4, Mai, 2009.

REMERCIEMENTS

Je tiens à remercier vivement mon promoteur Monsieur MERAOUMIA Abdellah, Professeur à l'université de Tébessa.

Je remercie mon co-promoteur Monsieur LOUAZEN Med Lakhdar, maitre de conférences à l'université de Ouargla.

Je tiens à remercier Monsieur SAMAI Djamel, maitre de conférences à l'université de Ouargla, pour l'honneur qu'il me fait en acceptant de présider le jury de cette thèse.

Qu'ils soient tous remerciés d'avoir accepté et participer à ce jury de la présente thèse, à savoir :

Pr. LOUAER Med Redha, professeur à l'université de Tébessa.

Dr. BENCHAABANE Abderrazak, maitre de conférences à l'université de Ouargla.

Dr. AMIEUR Toufik, maitre de conférences à l'université de Tébessa.

Merci à tous ceux qui de près ou de loin m'ont aidé.

Estimation de la dynamique des robots mobiles en mouvement

Résumé

Les systèmes intelligents, en particulier la détection et le suivi d'objets, ont reçu une attention considérable au fil des dernières années. Ce dernier représente un domaine de recherche très important avec des applications avancées en robotique, systèmes de protection des piétons, interaction homme-machine, etc. Par conséquent, il a suscité un grand intérêt dans la communauté scientifique. Dans ce travail, nous nous intéressons au suivi de plusieurs objets MOT (Multiple Object Tracking) dans **un environnement dynamique**. Le suivi multi-objets implique la détection et la reconnaissance d'objets dans la vidéo, présentant des défis tels que **l'encombrement de la scène, la dynamique des objets, la variation intra/inter-classe, le bruit de mesure, le mouvement du capteur et la fréquence d'images**. Le couplage de tracker à des détecteurs, dans un paradigme appelé suivi par détection, permet de mieux répondre à ces défis du domaine. Dans ce travail, nous avons exploité diverses méthodes de détection et de suivi multi-personnes sur des ensembles de données publics, représentées par un tracker SORT (Simple On line Real time Tracking) combiné avec des détecteurs visuels de personnes. Le système mis en œuvre modélise le mouvement des personnes en résolvant le problème de filtrage, à l'aide d'un algorithme Hongrois qui a pour fonction d'associer les détections à de nouveaux emplacements prédits dans de nouvelles images. Nos résultats expérimentaux montrent que le suivi est sensible au choix du détecteur et doit être soigneusement évalué avant utilisation.

Mots clés — détection d'objets, suivi par détection, environnement dynamique, suivi multi-objets, vision par ordinateur, association de données.

Estimation of the dynamics of mobile robots in motion

Abstract

Intelligent systems, especially object detection and tracking, have received considerable attention over the past years. The latter represents a very important area of research with advanced applications in robotics, pedestrian protection systems, human-computer interaction, etc. Therefore, it has aroused great interest in the scientific community. In this work, we are interested in the Multiple Object Tracking (MOT) in a **dynamic environment**. Multi-object tracking involves the detection and recognition of objects in the video, presenting challenges such as **scene clutter, object dynamics, intra-/inter-class variation, measurement noise, sensor movement, and frame rate**. The coupling of trackers to detectors, in a paradigm called tracking by detection, makes it possible to better respond to several challenges in the field. In this work, we exploited various methods of multi-person detection and tracking on public datasets, represented by a SORT (Simple On-line Real-Time tracking) tracker combined with visual detectors of people. The implemented system models the movement of people by solving a filtering problem, using a Hungarian algorithm that has the function of associating detections with new predicted locations in new images. Our experimental results show that tracking is sensitive to detector choice and should be carefully evaluated before use.

Keywords **—** object detection, tracking by detection, dynamic environment, multi-object tracking, computer vision, data association.

ملخص

حظيت الأنظمة الذكية، وخاصة اكتشاف الأشياء وتتبعها، باهتمام كبير في السنوات االاخيرة. يمثل هذا الأخير مجالا مهما للغاية للبحث مع تطبيقات متقدمة في الروبوتات، وأنظمة حماية المشاة، والتفاعل بين الإنسان والحاسوب ...الخ، مما أثار اهتماما كبيرا في المجتمع العلمي. في عملنا هدا نهتم بتتبع كائنات متعددة في **بيئة ديناميكية**. يتضمن التتبع متعدد الكائنات اكتشاف الأشياء والتعرف عليها في الفيديو، مما يؤدي الى تحديات متمثلة في **فوضى المشهد وديناميكيات الكائن والتباين داخل/بين-الأصناف وضوضاء القياس وحركة المستشعر ومعدل الإطارات**. إن اقتران أجهزة التتبع بالكشف، في نموذج يسمى التتبع عن طريق الكشف، يجعل من الممكن الاستجابة بشكل أفضل للعديد من التحديات. هناك عدة طرق للكشف عن الأشخاص وتتبعهم في مجموعات البيانات العامة. يدرس هذا العمل التعقب البسيط على المباشر في الوقت الحقيقي وأجهزة الكشف البصرية على الاشخاص. يقوم النظام المنفذ بنمذجة حركة الأشخاص من خلال حل مشكلة الترشيح واستخدام خوارزمية مجرية لربط عمليات الكشف بالمواقع الجديدة المتوقعة في الصور الجديدة. تثبت النتائج التجريبية المحصل عليها أن المراقبة حساسة الاختيار الكاشف ويجب تقييمه بصفة دقيقة قبل ادماجه في النظام.

الكلمات المفتاحية ـــ الكشف عن الأشياء، التتبع عن طريق الكشف، البيئة الديناميكية، تتبع الكائنات المتعددة، الرؤية الحاسوبية، ربط البيانات.

SOMMAIRE

LISTE DES TABLEAUX ET FIGURES

ACRONYMES

ACF: Aggregate Channel Features.

DPM: Deformable Part Model.

FA: False Alarms.

FN: False Negatives.

FP: False Positives.

GT: Ground Truth.

HOG: Histogram of Orientated Gradients.

ID: Identity.

IOU: Intersection Over Union.

LCDF: Local Decorrelated Channel Features.

JPDAF: Joint Probabilistic Data Association Filters.

MAP: Maximum a Posteriori.

MAP: Mean Average Precision.

MCMCDA: Markov Chain Monte Carlo Data Association.

MHT: Multiple Hypothesis Tracking.

ML: Mostly Lost (not tracked).

MOT: Multiple Object Tracking.

MOTA: Multiple Object Tracking Accuracy.

MOTP: Multiple Object Tracking Precision.

MT: Mostly Tracked.

NMS: Non-Maximal Suppression.

ROI: Regions of Interest.

SIFT: Scale Invariant Feature Transform.

SVM: Support Vector Machine.

SORT: Simple Online Realtime Tracking.

Chapitre 1
Introduction Generale

CHAPITRE 1

INTRODUCTION GÉNÉRALE

1.1. Problématique

Dans un environnement structuré ou non structuré, un robot mobile (ou un véhicule autonome) doit suivre une trajectoire définie par une courbe au sol, ou par une consigne définie dans les données sensorielles. Dans le contexte des environnements naturels dynamiques, un robot doit embarquer des capacités fonctionnelles et décisionnelles qui lui permettant de contrôler l'exécution de tous ses mouvements, d'estimer les mouvements réellement effectués et de détecter en temps réel que cette trajectoire est libre d'obstacles, afin de rester sur la trajectoire prévue.

Inspirée par la perception humaine et sa capacité naturelle à identifier les dangers potentiels en utilisant principalement la vision, cette thèse explore l'utilité des approches basées sur la vision pour détecter et suivre d'objets à proximité de type personnes qui pose une problématique à fort enjeu applicatif et très investiguée dans la communauté scientifique [Nguyen, 2016].

La vision par ordinateur a le potentiel d'être l'outil le plus répandu de la perception robotique avec un ensemble prolifératif de capacités y compris mais sans s'y limiter : la reconstruction de scène 3D, la localisation de lieu, la segmentation sémantique de scène, la reconnaissance, la détection d'objets et le suivi. Bien que la vision par ordinateur est appliquée avec succès dans des environnements structurés tels que l'intérieur [Castro et al., 2004, Marron et al., 2006] et les zones urbaines [Cornelis et al., 2008], l'extension de cette capacité aux environnements extérieurs et non structurés reste un défi. Cette recherche se concentre sur le développement de techniques de vision par ordinateur spécifiquement pour la détection et le suivi d'objets sans faire d'hypothèses sur la scène ou les objets qui ont un déploiement limité dans de nouveaux environnements.

La responsabilité principale d'un système de détection et de suivi est de recueillir et de maintenir une estimation de l'état actuel des objets externes. Ceci est particulièrement important dans de nombreux systèmes d'aide à la conduite et applications de véhicules autonomes où le potentiel de collisions est préoccupant. À la suite de la course à la mise en place de véhicules autonomes sur les routes principales, les problèmes de détection et de suivi sont devenus un domaine de recherche de plus en plus actif au sein de la communauté robotique. Bien que des progrès significatifs étaient réalisés avec les télémètres laser à haute résolution [Yang et Wang, 2011] et les cartes détaillées [Ferguson et al., 2008, Zhu et al., 2012], les approches basées sur la vision restent sous-utilisées.

La détection d'objets visuels consiste à localiser des instances d'une certaine classe sémantique dans l'image, le suivi visuel des objets cherche à localiser en permanence une instance d'objet spécifique à travers des images dans une séquence d'images afin de maintenir efficacement une trajectoire temporellement cohérente. Ensemble, ces capacités offrent une approche passive pour localiser et estimer la vitesse des objets en mouvement à proximité, comme requis dans de nombreuses applications industrielles. Par exemple, dans un contexte minier, il est nécessaire qu'un système de perception sur un camion de transport fasse la distinction entre les objets en mouvement (tels que les autres véhicules et les piétons) de l'encombrement de fond. De plus, le suivi de ces objets en associant les détections à travers les images permet de prédire leur position future, ce qui est essentiel pour éviter les collisions.

Les détecteurs basés sur la vision par ordinateur utilisent de plus en plus des approches basées sur les données, telles qu'un classificateur, pour décider si une région d'image contient un objet d'intérêt ou seulement un arrière-plan [Dalal et Triggs, 2005]. Ces classificateurs sont généralement formés hors ligne à l'aide de méthodes supervisées (également appelées apprentissage par lots) où le modèle appris est corrigé pour le déploiement en ligne. La formation de tels modèles nécessite un étiquetage d'objet approfondi qui est effectué par un humain, ce qui prend du temps et aboutit à des modèles qui sont généralement inflexibles aux changements. En outre, dans de nombreux scénarios réels, la variation d'apparence entre les images utilisées pour entraîner le modèle et les données rencontrées pendant le déploiement peut augmenter les erreurs de détection. Cette variation devient particulièrement difficile lors de l'utilisation d'un petit ensemble d'entraînement pour apprendre un détecteur pour un grand nombre d'objets différents, car le nombre de cas pouvant être appris est limité.

Idéalement, il est préférable que les modèles d'apparence utilisés pour la détection et le suivi soient appris automatiquement à partir de l'environnement observé avec une supervision

humaine minimale ou nulle. Cela permettra aux systèmes de détection et de suivi basés sur la vision par ordinateur d'être rapidement déployés dans de nouveaux environnements, où leurs performances augmentent au fil du temps à mesure qu'ils observent les différents modèles visuels spécifiques à l'environnement dans lequel le système a été déployé.

Les deux sections suivantes présentent une brève introduction à la détection et au suivi basés sur la vision où des lacunes spécifiques sont identifiées, ce qui conduit à trois questions de recherche ouvertes, voir section (1.4).

1.2. Détection visuelle d'objets

L'objectif de la détection visuelle d'objets est de localiser les instances multiples (le cas échéant) d'un type d'objet spécifié dans la limite de l'image. De nombreux algorithmes de détection classiques [Dalal et Triggs, 2005, Viola et Jones, 2001] et modernes [Benenson et al., 2013, 2012, Dollar et al., 2014] utilisent le paradigme populaire de la fenêtre coulissante descendante qui traite une seule image comme une collection d'images de sous-fenêtre. Ceci est généralement décrit par les étapes suivantes :

1. Sélectionner un ensemble de sous-fenêtres couvrant toutes les positions, échelles et proportions possibles contient un objet potentiel ;
2. Extraire les caractéristiques représentant l'apparence visuelle de chaque sous-fenêtre ;
3. Utiliser un modèle basé sur un classificateur pour étiqueter chaque fenêtre comme l'une des nombreuses classes connues Il peut s'agir d'un arrière-plan ou d'un objet d'intérêt.

Ces algorithmes présentent deux avantages clés, d'une part, plusieurs objets sont naturellement détectés par l'échantillonnage de plusieurs fenêtres de l'étape 1, et d'autre part, il peut être facilement adapté aux différents types d'objets en modifiant les caractéristiques et le classificateur utilisés respectivement aux étapes 2 et 3. Par exemple, le classificateur peut être formé pour faire la distinction entre les antécédents et les visages [Viola et Jones, 2004], les piétons [Dalal et Triggs, 2005, Viola et Jones, 2003] ou les voitures [Tzomakas et von Seelen, 1998]. Cependant, ces classificateurs s'appuient généralement sur une préformation supervisée dans le cadre d'un processus d'optimisation des lots hors ligne, nécessitant un grand nombre d'échantillons de formation étiquetés manuellement pour capturer les variations dans la distribution des fonctionnalités.

L'obtention d'ensembles de données étiquetés complets est généralement coûteuse et souvent peu pratique pour des applications extérieures spécifiques, en particulier lorsque l'on considère plusieurs types d'objets, chacun avec une grande variation d'apparence visuelle. Au

cours des dernières années, de vastes bases de données d'images étiquetées pour la classification d'objets [Deng et al., 2009, Fei-Fei et al., 2007, Torralba et al., 2008] ont été mises à disposition pour normaliser l'évaluation des algorithmes de détection d'objets visuels. Malgré l'amélioration continue de la performance mesurée par ces repères, on peut encore se demander dans quelle mesure cette performance se traduit par des scénarios du monde réel [Pinto et al., 2008].

D'autant plus que plusieurs de ces ensembles de données de référence sont connus pour contenir diverses formes de biais d'ensemble de données [Torralba et Efros, 2011], leur performance est susceptible d'être affectée négativement lorsqu'elle est appliquée à des images capturées dans un contexte différent avec des propriétés statistiques différentes. Les différences entre les apparences capturées dans les données d'entraînement et celles observées pendant le déploiement présentent un défi pour la détection d'objets.

1.3. Suivi visuel d'objets

Une fois qu'un objet d'intérêt est détecté, il est souhaitable de suivre l'objet d'image en image dans une séquence vidéo. Bien que le suivi des objets visuels a fait l'objet de recherches approfondies [Li et al., 2013, Wu et al., 2013, Yang et al., 2011, Yilmaz et al., 2006], la majorité de ce qui est considéré comme l'état de l'art en matière de suivi est soit conçu pour le suivi d'objets uniques [Babenko et al., 2010, Hare et al., 2011, Kalal et al., 2012, Pernici et Del Bimbo, 2014] soit mis en œuvre dans le cadre d'un processus par lots hors ligne où les informations contenues dans les futures trames peuvent être exploitées [Dicle et al., 2013, Pirsiavash et al., 2011, Zamir et al., 2012].

Perera et al. [2006] et Huang et al. [2008] ont montré qu'une technique classique d'association de données [Kuhn, 1955] peut être appliquée pour faire correspondre les détections aux hypothèses de trajectoire existantes. Dans ce cadre, chaque détection de la trame actuelle est adaptée à une trajectoire unique basée sur l'optimisation du coût total de l'affectation [Kuhn, 1955]. Le coût d'affectation est traditionnellement représenté comme la différence entre la position détectée et une position prédite le long de la trajectoire. Cependant, ces méthodes reposent sur une estimation précise du mouvement les limitant au suivi dans un plan de masse 2D enregistré [Perera et al., 2006] ou avec une caméra statique [Huang et al., 2008]. Les informations sur l'apparence sont de plus en plus utilisées pour aider à alléger ces restrictions en évaluant la probabilité que deux détections représentent le même objet, appelée affinité visuelle. Cela prend généralement la forme de l'utilisation du coefficient de

Bhattacharrya [Milan et al., 2014] ou via une corrélation croisée normalisée [Geiger et al., 2014]. Cependant, ces modèles d'affinité sont fixes, ce qui les rend sous-optimaux car ils ne prennent pas en compte les nuances spécifiques au déploiement de l'apparence des objets.

Comme plusieurs de ces ensembles de données de référence sont connus pour contenir diverses formes de biais d'ensemble de données [Torralba et Efros, 2011], leur performance est susceptible d'être affectée négativement lorsqu'elle est appliquée à des images capturées dans un contexte différent avec des propriétés statistiques différentes. Les différences entre les apparences capturées dans les données d'entraînement et celles observées pendant le déploiement présentent un défi pour la détection d'objets.

L'objectif est d'adapter les modèles d'apparence en apprenant de l'environnement déployé pour la détection. Cette idée peut également être appliquée à l'estimation d'un coût d'affectation basé sur l'affinité visuelle pour le suivi. Comme l'association de données est au centre de MOT (Multi Object Tracking), cette thèse aborde une question très importante de recherche en formulant l'estimation de l'affinité visuelle comme un problème de classification binaire où le but est de prédire si deux détections correspondent au même objet. Dans ce cadre, les données d'entraînement se présentent sous la forme d'une paire de détections avec leurs caractéristiques d'apparence et d'une étiquette indiquant si elles correspondent. Les étiquettes de ces données sont obtenues automatiquement en exploitant les contraintes spatiales pour les paires non correspondantes et la cohérence temporelle pour trouver des paires avec une forte affinité. Enfin, comme ce classificateur modélise l'affinité de deux détections quelconques, il se généralise à MOT tout en permettant l'apprentissage tout au long de la vie des nuances spécifiques au déploiement grâce au cadre auto-supervisé.

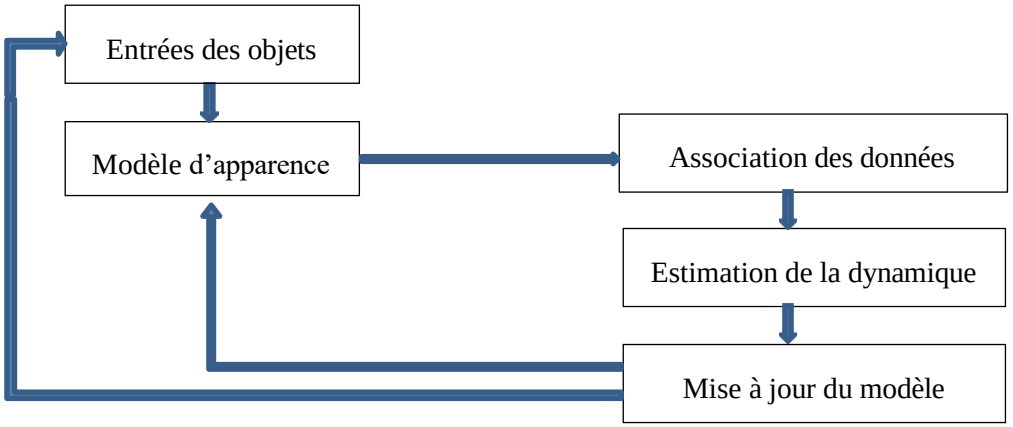


Figure (1.1) : Algorithme d'un suivi multi objets

1.4. Contribution

Les algorithmes de suivi se répartissent généralement en deux catégories : les méthodes en ligne et les méthodes hors ligne. Cette dernière utilise les informations des trames passées et futures pour prédire l'emplacement actuel de l'objet à suivre, tandis que les méthodes MOT en ligne n'utilisent que les informations qui existent déjà. Les méthodes en ligne conviennent aux applications en temps réel. Notre méthode est une méthode de suivi en ligne, d'autre part, ce travail fait principalement partie d'une méthode de suivi MOT basée sur la détection, et l'aspect détection signifie que les détecteurs d'objets sont indispensables dans le processus du suivi. L'algorithme de suivi MOT se compose de deux étapes principales : la construction d'un modèle d'apparence et la combinaison des données pour sélectionner le meilleur candidat pour chaque objet cible.

Par conséquent, la conception de l'algorithme de suivi implique de répondre aux questions : Comment déterminer le meilleur candidat ? Quand un objet cible peut-il être considéré comme invisible (occultation) ? La dissimulation est-elle partielle ou totale ? Cela nécessite une description robuste et efficace du modèle pour un objet cible à priori inconnu.

Nos contributions portent sur les deux composantes mentionnées ci-dessus, détection et suivi visuels d'objets, voir sections 1.2 et 1.3. En réponse aux objectifs de cette thèse, nous proposons une approche en ligne de suivi multi objets par détection en utilisant plusieurs détecteurs afin de surmonter les difficultés mentionnées ci-dessus.

Ce travail présente une évaluation comparative d'approches de suivi par détection, avec différents choix de détecteurs et de tracker, sur sept ensembles de données publiques pertinents avec des caméras statiques et d'autres mobiles. Sur la base des résultats expérimentaux exemplaires obtenus, qui dépassent le cadre de la présentation d'une idée et d'une discussion générale, l'impact du choix du détecteur et du tracker sur les performances de suivi est mis en évidence. En raison de leur large utilisation dans le domaine de recherche, de leur pertinence et de leurs performances dans MOT [L. Leal-Taixe et al, 2015], nous sélectionnons le tracker en temps Réel en Ligne Simple (SORT) [A. Bewley et al. 2016] parmi plusieurs tracker cités dans la littératures, ce dernier est couplé à quatre détecteurs sélectionnés, à savoir : Un détecteur basé sur un Histogramme de Gradients Orientés (HOG) désigné sous le nom de HOG-SVM [N. Dalal, B.Triggs, 2005], un détecteur de Méthodes basées sur des Pièces Déformables (DPM) [P. F. Felzenszwalb, 2010], un détecteur basé sur des Fonctions de Canal Agrégées (ACF) [P. Dollár et al, 2014], des Caractéristiques de Canal Localement Décorrélées (LDCF) [W. Nam et al., 2014] qui s'appuie sur ACF.

Les détecteurs et le tracker sélectionnés sont tout à fait pertinents et représentatifs pour l'évaluation comparative envisagée. Il s'agit d'un défi et d'une contribution à l'état de l'art. Par conséquent, nos principales contributions peuvent être résumées comme :

1. Evaluer des approches de suivi par détection fondées sur une combinaison pertinente de tracker et des détecteurs sur différents contextes applicatifs.
2. Sur la base des résultats expérimentaux obtenus à l'aide des détecteurs et du tracker choisis, nous essayons de présenter des idées et des interprétations qui mettent en évidence l'influence des choix de détecteurs et de tracker sur les performances du suivi.

1.5. Structure de la thèse

La suite de cette thèse sera organisée de la manière suivante :

Le deuxième chapitre sera consacré à l'état de l'art de la détection et suivi visuels d'objets avec une discussion critique des différents aspects de la problématique, on abordera les méthodes de suivi classique et les différentes approches de suivi MOT.

On traitera dans le troisième chapitre les théories concernant le suivi des personnes qui incluent l'association de détections dans différentes images et le filtrage.

Dans le quatrième chapitre on présentera l'approche développée pour le suivi des objets en mouvement dans une séquence vidéo. Ceci inclut la présentation en détail des contributions proposées à savoir la construction des différentes combinaisons détecteurs-trackers des objets cibles et l'association des données entre les objets candidats et les objets cibles.

Les expérimentations menées et les discussions seront décrites et développées dans le cinquième chapitre afin de montrer un aperçu qualitatif et quantitatif des différents aspects de la méthodologie proposée.

Finalement, le dernier chapitre sera destiné à la conclusion et aux perspectives à envisager comme suite à ce travail.

Chapitre 2
Travaux Connexes

CHAPITRE 2

TRAVAUX CONNEXES

2.1. Introduction

Ce chapitre présente un aperçu de la recherche existante dans les domaines liés à la détection et au suivi basés sur la vision par ordinateur. Bien que l'accent soit mis principalement sur le suivi par détection dans les environnements non structurés, ce chapitre contient des diverses formulations de détection visuelle d'objets et plus précisément des personnes. Dans la section 2.1 on expliquera pourquoi on a opté pour des suivis de personnes, comment des combinaisons de représentations de caractéristiques sont utilisées pour identifier et localiser des objets d'intérêt dans des images fixes et des séquences vidéo. Malgré que la détection a fait depuis longtemps l'objet de plusieurs recherches, un intérêt particulier est donné pour les méthodes de détection visuelle basées sur l'apprentissage automatique couvrant à la fois les approches non supervisées et supervisées. Pour plus d'exhaustivité, cette section explique les métriques d'évaluation de détection courantes qui sont utilisées dans les chapitres suivants de cette thèse.

La section 2.2 aborde la question du maintien des identités d'objets dans le temps dans une séquence vidéo pour le suivi d'objets. Cela inclut les problèmes d'association de données, de prédiction de mouvement et d'accumulation d'informations d'apparence. Différentes approches sont comparées en considérant leurs fonctionnalités et leurs limites. Cette section contient des mesures de suivi qui sont utilisées pour l'évaluation dans toute la littérature et dans les chapitres ultérieurs de ce travail.

On présente dans la section 2.3 certaines des principales tendances en matière de détection et de suivi basés sur la vision ainsi que leur application aux environnements extérieurs difficiles.

2.2. Détection visuelle d'objets

Contrairement à la reconnaissance qui suppose que les images contiennent l'une des catégories d'objets connues, la détection d'objets consiste à localiser potentiellement de nombreux objets dans l'image ou à identifier lorsqu'aucun objet n'est visible. Étant donné un ensemble de caractéristiques d'entrée extraites des données visuelles, les techniques de détection d'objets visent à prédire l'emplacement de l'objet cible. Cette partie aborde les techniques utilisées pour localiser les objets ainsi que les techniques de reconnaissance d'objets lors de la détermination de leurs emplacements, de plus, les approches qui prennent en compte le mouvement pour segmenter l'arrière-plan du premier plan sont également examinées comme une alternative aux méthodes basées sur l'apparence.

2.2.1. Localisation d'objets

Traditionnellement, la détection d'objets est formulée comme un problème de classification dans le paradigme bien connu des fenêtres coulissantes [Ziming et al., 2014, Zitnick and Dalal, 2014], où le classificateur est évalué sur une liste exhaustive de positions, d'échelles et de rapports d'aspect. Cette liste exhaustive peut contenir des millions de sous-fenêtres qui doivent être évaluées séquentiellement. Lors de l'utilisation d'un classificateur sophistiqué, le coût de calcul de cette approche devient rapidement prohibitif pour les applications en temps réel.

Pour réduire cette charge de calcul, de nombreux chercheurs ont essayé à réduire le nombre de fenêtres évaluées en analysant des fonctionnalités qui capturent l'indépendance d'échelle [Benenson et al., 2012, Dalal et al., 2010, Gavrila et Munder, 2006]. D'approches en cascade [Kalal et al., 2010, Viola et Jones, 2001] sont fréquemment utilisées pour accélérer la détection. L'approche classique de [Viola et Jones, 2001] ne calcule les caractéristiques que de manière incrémentielle sur la base des réponses précédemment évaluées des classificateurs simples dans le cadre d'AdaBoost [Freund et Schapire, 1997]. [Kalal et al., 2010] ont utilisé une cascade en trois étapes utilisant un classificateur plus sophistiqué après que la majorité des candidats ont été rejetés à l'étape précédente. De plus, le contexte géométrique de la scène peut être exploité pour concentrer le calcul sur les emplacements et les échelles les plus susceptibles de contenir les piétons [Benenson et al., 2012, Gavrila et Munder, 2006]. Au cours des dernières années, la réduction des fenêtres, à prendre en compte dans la détection, a formé un sous-domaine de la vision par ordinateur appelé propositions de détection (alias objectivité, propositions d'objets ou propositions de régions). Les propositions de détection sont considérées comme une étape de pré-traitement réduisant les millions de fenêtres candidates

par image à des centaines ou quelques milliers. Plusieurs techniques de proposition d'objet sont basées sur la fusion de techniques de segmentation de bas niveau [Alexe et al., 2012, Carreira et Sminchisescu, 2012, Manen et al., 2013, Rantalankila et al., 2014, Uijlings et al., 2013]. Étant donné que ces techniques utilisent la segmentation d'image, telle que [Felzenszwalb et Huttenlocher, 2004], comme étape de prétraitement, leur utilité en tant que technique de réduction de calcul s'accompagne d'une surcharge supplémentaire.

D'autres techniques combinent des fonctionnalités simples avec une approche basée sur des fenêtres coulissantes conçue pour la vitesse en tirant parti des structures de données efficaces telles que des images intégrales [Ziming et al., 2014, Zitnick et Dollár, 2014]. Ces derniers] ont observé que les objets sont généralement caractérisés par le nombre de contours entièrement entourés d'une boîte englobante. [Zhiming et al. 2014] introduirent une caractéristique de gradient normé binarisé qui est calculée efficacement sur les architectures informatiques modernes et utilisée avec un simple classificateur SVM (Support Vector Machine ou Machine à vecteurs de support) linéaire. Récement une comparaison détaillée de diverses techniques de proposition de détection a été évaluée dans [Hosang et al., 2014] et de plus outre les avantages informatiques, il a été démontré que les propositions de détection amélioraient également les performances de détection [Girshick, 2015].

2.2.2 Modèles de Reconnaissance d'Objets

En détection supervisée, les types d'objets sont connus a priori, tels que les piétons ou les voitures. Ces techniques sont généralement conçues pour une seule image où chaque emplacement est inspecté pour la présence de l'objet d'intérêt. Étant donné une sous-fenêtre sélectionnée soit via une technique de proposition de détection, soit avec une fenêtre coulissante, un descripteur de longueur fixe est ensuite extrait pour une classification d'objet ou d'arrière-plan. Comme ces méthodes sont basées sur l'apparence visuelle de l'objet (couleur, texture ou forme) dans une seule image, elles évitent de nombreux problèmes liés à la modélisation du mouvement pour les caméras en mouvement. Dans un cadre supervisé, des images positives de l'objet cible et des images d'arrière-plan négatives sont présentées à un classificateur qui apprend un mappage de l'espace d'entités à l'espace d'étiquettes. Le classificateur code la limite de décision dans l'espace d'apparence qui spécifie si une fenêtre nouvellement présentée contient l'objet cible d'intérêt.

L'un des premiers détecteurs d'objets réussis est le cadre de détection Viola-Jones appliqué à la détection de visages [Viola et Jones, 2001], il est généralisé plus tard pour détecter

d'autres classes d'objets [Viola et Jones, 2003]. Leur approche a utilisé AdaBoost [Freund et Schapire, 1997] pour sélectionner un ensemble de classificateurs d'arbres de décision faibles basés sur des caractéristiques d'ondelettes de Haar et ordonner les réponses de décision pour rejeter une hypothèse dans un calcul de structure en cascade -focalisant sur les régions les plus susceptibles d'être l'objet d'intérêt-. Dalal et Triggs [2005] exploitent le pouvoir expressif du descripteur HOG pour former un classificateur SVM linéaire qui bénéficie d'une réduction d'un ordre de grandeur des faux positifs au même taux de détection que le détecteur Viola-Jones (tel que rapporté par [Dollár et al., 2009b]). Plus récemment, [Zhang et al., 2009] ont également proposé une architecture AdaBoost utilisant des classificateurs de base LDA avec des caractéristiques de covariance [Tuzel et al., 2007].

Une limitation majeure de ces techniques, basées sur la reconnaissance d'objets, est dû à l'exigence d'ensembles complets d'entraînement et de tests étiquetés, qui capturent une grande variance d'apparence visuelle causée par le point de vue, la déformation et l'occlusion. Par exemple, [Junior et al., 2009] ont utilisé environ 9600 images marquées positives et 10000 images marquées négatives pour entraîner un détecteur de piétons avec 9800 images supplémentaires pour les tests. De plus, une fois entraîné, le classificateur boosté ne peut pas s'adapter au scénario particulier dans lequel il est utilisé [Javed et al., 2005]. Pour traiter la question du point de vue [Rybski et al., 2010] ont formé plusieurs classificateurs, un pour chaque angle de vision de 45o d'une voiture. Les problèmes de déformation et d'occlusion ont récemment été résolus avec des modèles basés sur des parties qui décrivent un objet comme une alection de parties visuelles.

Pour surmonter les problèmes d'occlusion et d'objets non rigides, plusieurs modèles basés sur des pièces déformables ont été proposés [Agarwal et al., 2004, Andriluka et al., 2008, Azizpour et Laptev, 2012, Bouchard et Triggs, 2005, Felzenszwalb et al., 2008, 2010, Leibe et al., 2004]. Ces méthodes capturent les relations spatiales entre différentes parties. Par exemple, le torse d'une personne est placé sous sa tête et entre ses bras gauches et droit. La détection conjointe de plusieurs parties et leur relation spatiale entre elles peuvent être utilisées pour améliorer les performances de détection d'objets de niveau supérieur [Felzenszwalb et al., 2010].

[Leibe et al., 2004] a lancé ce domaine avec le Modèle de Forme Implicite (ISM) qui enrichit un livre de codes de modèles d'apparence visuelle locaux avec une distribution de probabilité spatiale définissant le décalage relatif de l'ensemble de la classe d'objets spécifique. [Wu et Nevatia, 2007] apprennent plusieurs classificateurs basés sur les parties humaines en

fonction des caractéristiques de bord avant de combiner les parties en un détecteur humain entier dans un cadre d'estimation maximale a Posteriori (MAP). [Felzenszwalb et al., 2008] formalisent le Modèle de Parties Déformable (DPM) en tant que structure hiérarchique en extrayant des caractéristiques à deux échelles, l'échelle de parcours représentant des objets et la plus fine représentant des pièces. [Zhu et al., 2010] étendront ce cadre à une hiérarchie à trois couches et montrent que la structure plus profonde surpasse la DPM à deux couches et démontrent également que des structures de parties plus simples sont suffisantes pour obtenir des résultats solides.

Un autre problème critique pour la détection d'objets est le coût de calcul qui limite leur valeur pour la détection en temps réel à partir de plates formes mobiles. Le détecteur avec le meilleur compromis de performance au moment de l'exécution est celui de [Dollár et al., 2010] qui remplace l'extraction d'entités à plusieurs échelles par un ensemble de classificateurs à plusieurs échelles. D'autres techniques limitent la recherche à des régions d'intérêt spécifiques (ROI) dans l'image en utilisant la stéréovision avec l'hypothèse du plan de masse [Bajracharya et al., 2009, Benenson et al., 2012, Gavrila et Munder, 2006, Howard et al., 2007].

2.2.3 Contextes

Pour la détection d'objets en mouvement, des techniques de segmentation de mouvement peuvent également être appliquées pour identifier des objets en mouvement sans restriction d'une classe sémantique spécifique ni connaissance préalable de leur apparence visuelle. Contrairement à la reconnaissance d'objets d'intérêt précédemment modélisés, [Grimson et al., 1998] ont montré que la détection peut être obtenue en modélisant d'abord la scène d'arrière-plan, puis en identifiant les régions de l'image qui ne correspondent pas à ce modèle. Comme cette approche se concentre sur la modélisation de l'arrière-plan, les variations de l'apparence de l'objet à partir du changement de point de vue, de l'occlusion ou de la déformation deviennent sans importance car tout objet qui contrevient au modèle de fond sera détecté. Leur approche fonctionne en comparant continuellement le cadre actuel à un modèle de l'arrière-plan afin que les régions où il y a une différence significative soient mises en évidence comme premier plan, cette approche simple est généralement décrite comme une soustraction de fond.

La majorité de la littérature [Grimson et al., 1998, Heikkila; et al., 2004, Heikkila et Pietikainen, 2006, Piccardi, 2004, Singh et al., 2009] de soustraction de fond s'intéresse à la détection de nouveaux objets de premier plan à partir d'une caméra vidéo fixe, car les

applications de surveillance ont déjà été le moteur clé de la recherche dans ce domaine. Dans de telles applications, il s'agit d'une technologie mature avec des centaines de techniques proposées [Bouwmans, 2011], mais exclusivement pour une caméra statique. Alors que certaines méthodes [Borges, 2013, Elgammal et al., 2002, Reddy et al., 2013, Zivkovic et van der Heijden, 2006] peuvent gérer des situations dynamiques douces telles que l'ondulation d'arbres ou une légère oscillation de la caméra due au vent, elles ne sont pas adaptées au niveau de scènes dynamiques vécues à partir d'une caméra montée sur une plateforme mobile.

Les techniques conçues pour gérer le mouvement de la caméra impliquent généralement le calcul du flux optique suivi soit d'une estimation du mouvement, soit d'un regroupement de mouvements sur la distribution des vecteurs de flux extraits. L'Ego-motion est l'estimation du mouvement relatif de la caméra par rapport à l'environnement statique rigide et a longtemps été utilisé dans Structure From Motion (SfM). Les effets du mouvement de la caméra peuvent être observés à l'aide du flux optique et estimés à l'aide de contraintes géométriques telles que la contrainte épipolaire ou trifocale [Kim et al., 2012]. Comme le flux optique généré par des objets en mouvement indépendant ne correspond pas au mouvement général de la scène, un estimateur robuste aberrant tel que RANSAC [Fischler et Bolles, 1981] ou LMedS [Rousseeuw, 1984] sont généralement utilisés pour estimer le modèle de mouvement de la caméra [Kitt et al., 2010]. [Guizilini et Ramos, 2013] utilisent la sortie d'une approche basée sur RANSAC [Hartley, 1997] pour sélectionner des points candidats pour l'apprentissage en ligne d'un modèle non paramétrique basé sur la classification des processus gaussiens afin de mieux identifier les parties statiques et dynamiques de la scène. De même, [Sheikh et al., 2009] a proposé de modéliser la base de trajectoire des points d'arrière-plan saillants qui sont identifiés à l'aide de RANSAC avant d'apprendre une segmentation en pixels dans un cadre de champ aléatoire de Markov (MRF). Cependant, cette méthode est à la fois lente sur le plan du calcul et nécessite également jusqu'à 30 images pour estimer la base de la trajectoire.

Au-delà du problème de segmentation d'arrière-plan/avant-plan, peu d'œuvres ont utilisé des techniques de clustering pour séparer plusieurs objets d'avant-plan dans la scène. [MacLean et al., 1994] décrivent le mouvement de la scène comme un GMM utilisant des caractéristiques de mouvement et utilisent l'algorithme EM pour estimer les portions de mélange des échantillons de données. Cependant, ils supposent que le nombre d'objets en mouvement est connu. [Lenz et al., 2011] détectent un nombre quelconque d'objets en connectant des points d'intérêt clairsemés à l'aide de la triangulation de Delaunay, puis regroupent les points en supprimant les bords avec des fonctions stéréo et de mouvement non similaire.

2.2.4 Détection des humains

La détection visuelle d'objets est très importante dans le domaine de la vision par ordinateur, et ses dérivés incluent l'assistance à la conduite [Gerónimo, 2010a], la classification d'images [Zhang, 2009], vidéosurveillance [Breitenstein, 2011], robot mobile [Ess, 2010]. Cet intérêt est étroitement lié aux capacités toujours en expansion des ressources informatiques, comme les œuvres de [Zhang et al., 2013], [Dalal et al., 2012], et [Gerónimo et al., 2010a].

Ceci est également vérifié par différents défis proposés, tels que Image-Net [Russakovsky, 2015], MOT Challenge [Leal Taixé, 2015], Pascal VOC Challenge [Everingham, 2010, Everingham, 2015]. Nous concentrons nos recherches sur la détection visuelle d'objets tels que les personnes, les problèmes d'application à fort enjeu, et un large corpus de recherche dans la communauté [Nguyen, 2016] pour pouvoir mieux localiser et comparer le présent.

Il existe des méthodes de comparaison. Rappelons que la détection visuelle vise à détecter toutes les personnes dans le plan image en décrivant leur position et leur échelle et les bounding box associées. Nous avons répertorié ci-dessous les principaux obstacles liés à ce problème :

- Variations morphologiques des candidats La forme humaine varie d'un individu à l'autre, et ces variations morphologiques peuvent être causées par le port de vêtements.
- Variations d'apparence des candidats l'apparence des vêtements, des cheveux et/ ou de la couleur de la peau peut également varier considérablement ;
- Modifications de attitudes des cibles, l'enveloppe corporelle est déformable ;
- L'éclairage de la scène change le système visuel est passif et dépend donc des conditions d'éclairage dans lesquelles la scène est vue ;
- Changement de joncture de vue la cible modification fortement sur le plan de l'image en raison de la joncture de vue relatif caméra/ scène ;
- Encombrement perçu de la scène l'environnement peut être encombré, entraînant ainsi de fausses détections, voire des occlusions ;
- Capteurs Les capteurs ont des finalités physiques liées à leur édifice optique et électronique. Les frames sont par définition bruitées ;

Dans la majorité des recueils de la littérature, les méthodes courantes de détection visuelle sont basées sur des techniques de fenêtres glissantes (Viola, 2004), en particulier lorsqu'il n'y a pas d'informations contextuelles préalables sur les frames analysées (Dalal, 2012). Ensuite, le principe est d'isoler l'objet en classant les boîtes englobantes générées après apprentissage supervisé, en balayant exhaustivement l'emplacement et l'échelle, ce qui est évidemment coûteux en temps de calcul.

La littérature sur la détection de personnes est abondante, comme en témoigne l'émergence récente de nombreuses techniques de pointe (Dollár 2012, Gerónimo 2010a). En résumé, le premier succès a été obtenu en utilisant des descripteurs tels que les ondelettes de Haar, qui peuvent capturer des différences locales d'intensités régionales mais avec un pouvoir descriptif limité (Papageorgiou 2000, Viola 2004), qui sont des descripteurs Histogramme Orienté objet (Dalal, 2005).

De nombreux travaux utilisent par la suite ces descripteurs HOG, parfois en combinaison avec d'autres descripteurs. Prenons un exemple : le détecteur HogLBP [Wang, 2009] combine des descripteurs HOG avec des motifs binaires locaux (LBP), et le détecteur MultiFTR [Wojek, 2008] combine HOG avec Haar et forme des descripteurs contextuels. Des gains significatifs ont ensuite été obtenus grâce au détecteur Deformable Part Model (DPM), qui utilise un descripteur HOG légèrement modifié dans un cadre de détection basé sur les parties qui recherche explicitement des parties distinctes automatiquement apprises d'une personne (Cinq pour être exact) pour le détecter [Felzenszwalb, 2010], suivi d'un détecteur basé sur les caractéristiques du canal [Dalal, 2009a] ou ses dérivés. Combiner les fonctionnalités du canal avec un processus Soft-Cascade via le Boosting [Zhang, 2007, Bourdev, 2005] offre un meilleur arrangement entre le taux de détection et le temps de calcul.

Récemment, les paradigmes d'apprentissage en profondeur ont été favorisés par les détecteurs visuels [Hosang, 2015, Tian, 2015, Ren 2017], et bien que ces paradigmes soient très prometteurs, des expériences récentes montrent que les détecteurs basés sur Soft-Cascade sont supérieurs en termes de performances de détection et de calcul. Temps encore compétitif [Zhang, 2016b].

Toutes les méthodes de la littérature respectent plus ou moins le schéma général de la figure (2.1), en générant d'abord une fenêtre ou boîte englobante sur l'image à analyser en supposant que le candidat existe à l'intérieur. Des descripteurs visuels considérés comme discriminants sont ensuite extraits des fenêtres candidates, qui sont apprises dans une étape

d'apprentissage supervisé hors ligne. Enfin, la fenêtre est classée comme cible ou non cible par les règles de décision du classifieur (détection = classification binaire).

Bien que cela ne soit pas mentionné dans la figure, il y a généralement une étape finale de post-traitement dont le but d'éliminer les détections qui se chevauchent grâce à la suppression non maximale (NMS), dont le but est de combiner plusieurs détections liées à la même personne, comme celles maintenant dans la figure (2.2). Les deux principales méthodes utilisées dans la littérature pour cela sont l'estimation du mode par décalage moyen (MS) [Dalal, 2006a] et la suppression du maximum par paires (PM) [Felzenszwalb, 2010]. Ce dernier sélectionne la boîte englobante la plus probable parmi les boîtes qui se chevauchent, le premier (MS) quant à lui, comme son nom l'indique, sélectionne la moyenne des boîtes qui se chevauchent.

La figure (2.1) est un schéma fonctionnel d'une détection visuelle en ligne classique. La nature des descripteurs et des classificateurs et le modèle de personne exact utilisé doivent être choisis à l'avance, mais le sous-ensemble de descripteurs à utiliser (discriminant) et les réglages des paramètres du classificateur sont déterminés par un apprentissage hors ligne basé sur les données d'entraînement contenant la cible. Annotez des échantillons ou des cadres de délimitation (personnes dans notre travail, (positif) ou autre (négatif).

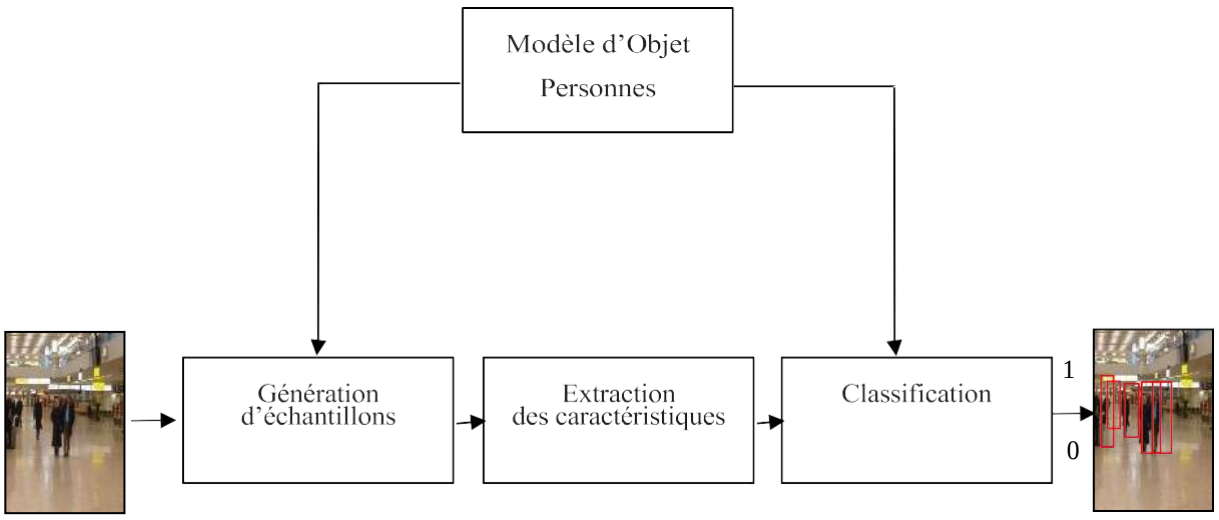
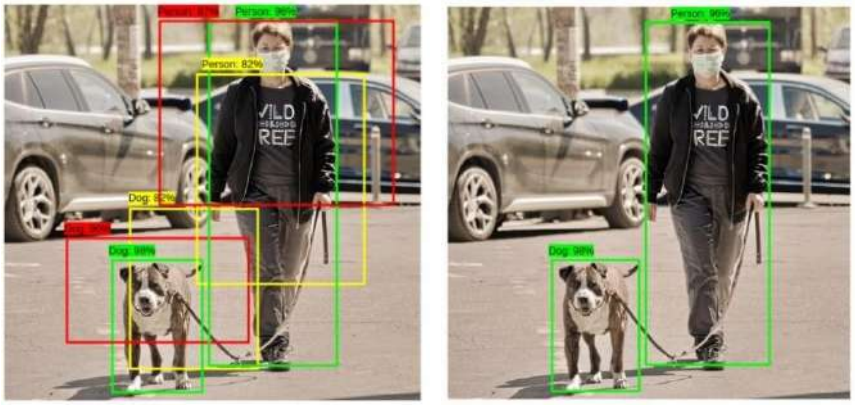


Figure (2.1) : Détection visuelle de cibles : classification des fenêtres candidates (processus online)



(a) Sans NMS (b) Avec NMS

Figure (2.2) : Exemple de détection sans / avec suppression des non-maxima (NMS)

2.2.4.1. Modèles de personnes

Classiquement, les détecteurs de personnes, Figure (2.1), séquencent trois étapes : génération de fenêtre candidate, extraction de descripteur et classification. Tous ces blocs recherchent généralement le corps humain complet à l'aide d'un modèle sous-jacent qui encapsule la signature commune de la personne ou, alternativement, recherchent des parties du corps et les connectent pour déduire l'existence d'une personne. La littérature distingue les méthodes implicites et explicites. Les méthodes implicites n'utilisent pas d'indices spécifiques à l'homme, préférant d'autres indices.

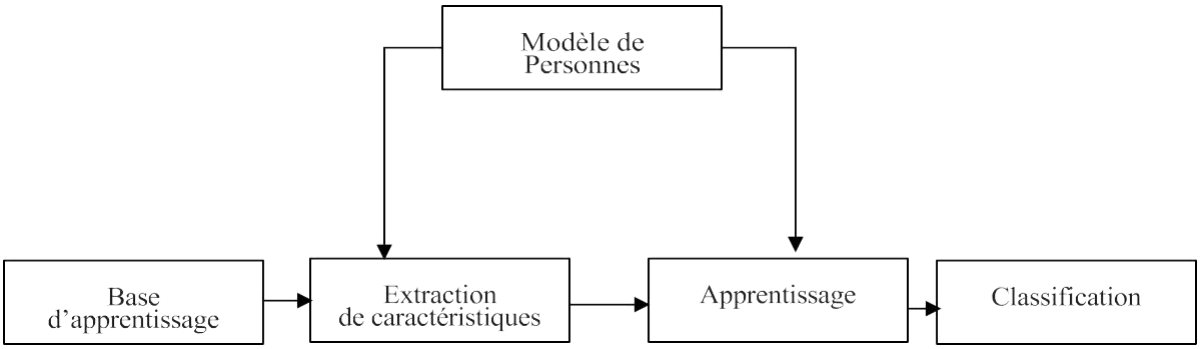


Figure (2.3) : Détection visuelle de Candidats : processus d'apprentissage offline

- **Méthodes implicites :**

Les méthodes implicites détectent les personnes en tenant compte de leurs différences par rapport à leur environnement. Ce sont des méthodes implicites car elles n'infèrent pas explicitement la présence des personnes, mais segmentent plutôt les objets en mouvement au premier plan et les étiquettent.

En tant que personnes si elles respectent le rapport d'aspect de la personne moyenne. Deux techniques populaires dans cette catégorie sont la soustraction de bruit de fond [Stauffer, 1999, Piccardi, 2004] et l'analyse de flux optique [Beauchemin, 1995]. L'inconvénient est que ces techniques ne conviennent qu'aux caméras statiques et aux objets dits en mouvement, qui sont facilement trompés par des objets de premier plan avec des rapports d'aspect comparables à ceux des humains. En bref, ces méthodes implicites sont moins efficaces et les méthodes explicites sont généralement préférées.

- **Méthodes explicites :**

Les méthodes explicites utilisent un modèle qui capture les propriétés discriminantes saillantes qui reconnaissent une personne des autres caractéristiques de l'image. Utilisez des exemples de formation positifs et négatifs pour déterminer la forme et les valeurs exacts du modèle. La plupart des méthodes sont basées sur cette notion [Dalal, 2012, Gerónimo, 2010, Enzweiler, 2009]. Les méthodes explicites peuvent être divisées en deux parties : holistiques (corps entier) et partielles. Les méthodes holistiques considèrent un modèle corporel complet d'une personne, tandis que les méthodes basées sur les parties visent à agréger les détections de parties du corps pour isoler ou segmenter une personne dans une image.

Méthodes holistiques : Ces méthodes considèrent un modèle corporel complet pour détecter une personne [Gavrila, 2000, Gavrila, 1999, Broggi, 2000, Broggi, 2006], et le principe est d'apprendre un ensemble de figures humaines hors ligne à partir d'exemples frontaux et représentatifs Template variabilité (apparence, contour, etc.). Le processus en ligne vise ensuite à faire correspondre la fenêtre du candidat avec l'un des modèles appris [Gerónimo, 2010a].

Une autre approche plus large tend à former (hors ligne) à discriminer les descripteurs sur des cases renfermant tout le corps qui devraient mieux séparer les exemples positifs et négatifs par un classificateur, ce dernier étant extrait en ligne des images. Les régions candidates sont marquées comme analysées en personne ou non, citons des descripteurs communs de type HOG [Dalal, 2005] ou du type Channel Features plus récent [Dollár, 2009a].

Les méthodes corps entier sont attractives en raison de leur : abstraction simple, apprentissage direct du modèle et temps de calcul réduit par rapport aux méthodes basées sur les parties.

Alors que, lorsque les modèles sont formés sur des personnes debout, Alors ils ne sont pas efficaces pour différentes poses ou en cas d'occlusion partielle.

Méthodes repose sur des parties : ces méthodes basées sur la détection des sections du corps explicites (tête, jambes, bras, etc.) ou implicites pour détecter les personnes. L'essentiel de ce travail est basé classiquement sur un modèle de structure d'image [Fischler, 1973], qui représente un objet comme un ensemble de deux parties reliées [Fischler, 1973, Felzenszwalb, 2005].

L'inconvénient des modèles de pièces anatomiquement pertinents est que les pièces manquantes dues à une occlusion partielle peuvent affecter la probabilité de l'ensemble du modèle composite. Pour remédier à cette situation, [Felzenszwalb et al., 2010] proposent un modèle de pièce déformable qui sélectionne les pièces en fonction de la saillance visuelle pour découvrir les pièces du modèle de manière non supervisée au lieu de s'appuyer sur des informations sémantiques. Les parties de vérité-terrain ne sont pas démontrées à l'avance, mais sont étiquetées comme des boîtes englobantes pour l'ensemble du corps et le nombre de parties, leur algorithme sélectionne les parties saillantes des données d'apprentissage par optimisation itérative avec des graphes de déformation associés. Une variante consiste à apprendre des parties basées sur l'efficacité visuelle par exemple ; [Dalal et al., 2008] résolvent ce problème en utilisant l'apprentissage multi-composants (MCL).

En général, les méthodes basées sur les pièces sont plus adaptées à la détection des humains car elles sont plus robustes :

- Variations de points de vue ;
- Partiellement Caché ;
- Changement et déformations de la posture humaine.

Associer des parties à des composants anatomiques humains équivalents facilite la tâche, mais un grand problème avec des parties qui manques [Mohan, 2001, Mikolajczyk, 2004].

Des modèles robustes peuvent être obtenus en utilisant des pièces communes ne contenant pas d'informations sémantiques et sont sélectionnées uniquement en fonction de leur visibilité [Felzenszwalb, 2010, Leibe, 2008, Dalal, 2008]. Cela évite également la tâche fastidieuse d'annoter manuellement chaque partie et simplifie la détection d'autres objets, de plus, la décision de la partie sémantique peut être ambiguë ou subjective. Cependant, les avantages ci-dessus se font au détriment du temps de calcul, qui est plus important pour l'apprentissage et la détection. Les méthodes basées sur les parties sont inefficaces sur les images à basse échelle parcequ'elle demandent un support spatial suffisant pour assurer la

robustesse. Lorsqu'une résolution suffisante est disponible, un compromis de méthode peut être mise en œuvre afin d'utiliser des approches partielles et holistiques [Park 2010].

2.2.4.2. Génération de fenêtres candidates

Il existe plusieurs stratégies dans la littérature pour générer des fenêtres candidates (pour une classification rigide ou non rigide) dans des images brutes, et trois tendances ont émergé : les méthodes par force brute, les méthodes géométriques et les méthodes par segmentation des points de vue. La plus simple est la méthode de la force brute, souvent appelée stratégie de la fenêtre glissante [Viola, 2004]. Les fenêtres candidates avec des rapports d'aspect fixes sont échantillonnées à toutes les positions et échelles du graphique d'image d'origine (2.4), cette stratégie ne nécessite aucune connaissance préalable de la scène et de la caméra [Dalal, 2012], elle est avantageuse car il n'y a pas d'hypothèses dans l'image (position et échelle) sont exclues, mais leur nature combinée augmente le temps de calcul en raison du balayage des zones de l'image incompatibles avec la présence humaine (plafonds, etc.).

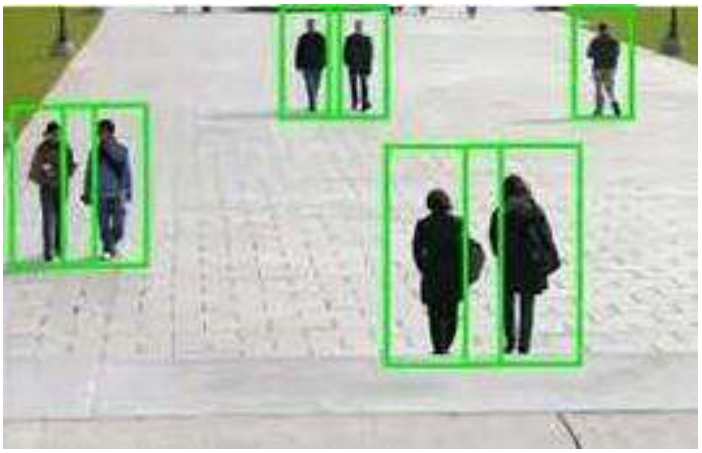


Figure (2.4) : Exemples de fenêtres glissantes sur une image

La géométrie entre la caméra et le sol est connue (caméra dite calibrée), ce qui réduit le nombre total de fenêtres candidates générées. Par conséquent, pour les caméras au sol plat (généralement à l'intérieur) et calibrées, seule la fenêtre d'image connectée à la projection du plan de sol est traitée. Ce type de contrainte géométrique réduit considérablement le nombre total de possibilités : toutes les hypothèses ou fenêtres dans les hauts murs et plafonds (en milieu intérieur) et dans le ciel (en milieu extérieur) sont rejetées. Pour les caméras sur des véhicules en mouvement, l'inférence préalable du plan au sol et du déplacement peut également réduire cette combinaison [Gerónimo, 2010b].

Un autre mécanisme pour identifier et segmenter les fenêtres candidates pour des essais supplémentaires. Par exemple, pour les caméras statiques ou ambiantes, les méthodes implicites susmentionnées, généralement la soustraction de fond, aident à pré-segmenter les régions mobiles qui peuvent être assimilées à des mécanismes d'attention. Tout mécanisme peut éliminer un certain pourcentage de fenêtres négatives sans perdre de cibles serait bénéfique pour réduire le coût de traitement global. Divers travaux proposent des métriques simples qui satisfont aux conditions ci-dessus, telles que le contraste des couleurs, la densité des contours, le chevauchement des pixels, la symétrie des couleurs et la symétrie des contours, comme indiqué dans [Alexe, 2010, Paleček, 2012], et d'autres indices tels que la verticalité. Les directions principales et dominantes sont également proposées dans [Paleček et al., 2012]. [Liu et al., 2016] ont construit un prototype de réseau neuronal a apprentissage profond pour extraire des fenêtres cibles de différentes échelles.

- **Extracteur de caractéristique**

Il y a beaucoup de gens qui font une inspection visuelle. L'utilisation d'une seule valeur de couleur de pixel dans la première image conduit directement à de mauvaises performances de détection et à une généralisation, car un seul point ne peut pas transmettre d'informations globales sur l'apparence d'une personne. En regardant simplement un ensemble de points adjacents, des informations utiles sur l'objet sous-jacent peuvent être obtenues, par exemple, en regardant les voisins immédiats d'un pixel, il est possible de déterminer l'ampleur et la direction des changements dans l'aspect spatial de l'objet sous-jacent à ce moment-là. Temps. Un agrandissement supplémentaire de la zone de support peut aider à capturer les contours de la structure de bord des objets, par conséquent, la représentation d'image modifiée est cruciale pour améliorer l'interprétation de l'image. Toute information extraite de cette façon on l'appelle un descripteur. Le descripteur nous permet de caractériser ou de différencier un type d'objet en extrayant des informations significatives des régions pertinentes de l'image. Ils encapsulent les caractéristiques et les propriétés de la cible et ont un impact significatif sur les performances du détecteur. Outre les classificateurs, le choix des caractéristiques robustesses discriminantes est également important [Dalal, 2012, Gerónimo, 2010a].

Au départ, les détecteurs tendaient vers des descripteurs basiques de type Haar dérivés des ondelettes de Haar [Papageorgiou, 2000, Lienhart, 2002, Viola, 2005]. Ces caractéristiques utilisent la somme des différences de l'intensité entre les contrastes locaux à l'objet à détecter. Les valeurs des descripteurs peuvent être calculées via des images complètes.

[Viola, 2004]. Plus précisément, ces descripteurs capturent les variations locales d'intensité selon différentes directions de l'image. Ce descripteur est paramétrable (position, orientation, échelle de l'image) et le descripteur le plus discriminant est sélectionné par boosting [Schapire, 2003].

Il existe également des descripteurs qui ne sont pas basés sur le contraste régional mais sur la structure spatiale des pixels de contraste dans la cible, tels que les descripteurs codant les histogrammes d'orientation des bords (EOH), proposés à l'origine pour détecter les visages [Levi, 2004]. Ces descripteurs représentent des rapports de gradient calculés à partir d'histogrammes d'orientation de contour. Dans une région de chevauchement donnée, les gradients sont d'abord calculés, puis un histogramme de gradient est bati en quantifiant les directions. Ces caractéristiques, connus d'être robustes aux variations globales d'éclairement, sont avant tout plus rentable que le descripteur de Haar [Gerónimo 2007].

Les variantes basées sur les gradients d'image sont proposées pour l'Histogramme de Gradients Orientés (HOG) (Dalal, 2005). Les descripteurs HOG sont d'abord extraits en calculant les gradients, puis en construisant un histogramme pondéré par l'amplitude des gradients dans ses intervalles à changement discrète appelés cellules. Les histogrammes des cellules adjacentes sont regroupés en blocs individuels, normalisés croisés et concaténés pour donner un vecteur descripteur pour chaque bloc. (Dalal et Triggs, 2005) concatènent tous les histogrammes de blocs dans la fenêtre cible afin de générer un seul descripteur. Alors que, (Zhu et al., 2006) les comparaisons entre les détecteurs sont obtenues en utilisant un module de HOG de taille flexible sélectionnés et combinés par boosting. Les descripteurs HOG sont composés de modules entiers (Dalal, 2012) ou de segments (Felzenszwalb, 2010).

Une autre variante du descripteur est basée sur le Modèle Binaire Local (LBP). Celles-ci, dans un premier temps, sont analogues à des extracteurs de texture [Ojala, 1996], alors que l'objectif est de calculer une étiquette entière quadratique modulée pour chaque pixel par un seuil de pixels voisins par lesquels passe uniformément le pixel central. L'étape de seuillage considère des valeurs d'intensité relative qui maintiennent les descripteurs invariants à l'illumination et au contraste. Les motifs de texture avec différents supports spatiaux peuvent être capturés par différents rayons de voisinage et points d'échantillonnage. Le descripteur final peut être calculé en construisant simplement un histogramme dans une région rectangulaire et en calculant toutes les régions rectangulaires possibles dans la fenêtre candidate, résultant en un ensemble complet de descripteurs ou en créant un descripteur de grande dimension tel que HOG [Mu, 2008, Satpathy, 2013]. Les variantes des descripteurs LBP proposées dans la

littérature comprennent le motif binaire local non redondant catégorique (NRLBP) [Mu, 2008], le motif binaire local robuste discriminatif (DRLBP) [Satpathy, 2013] et le motif binaire local structuré en cellule (CellLBP) [Wang, 2009].

En raison de l'apparence très modifiable causée par les vêtements, les caractéristiques de couleur ne sont toujours utilisées pour la détection de personnes. Mais même sur les vêtements, les couleurs montrent des similitudes approximatives. (Walke et al. ; 2010), encodant la similarité des différentes sous- régions. Les extracteurs sont a priori calculés en divisant la fenêtre candidate en blocs de pixels, après un histogramme de couleur est calculé dans chaque bloc. Pour chaque module, la similarité est calculée comme l'intersection des histogrammes de blocs individuels. Dans (Walk, 2010) un seul vecteur descripteur de grande magnitude est défini en concaténant toutes les valeurs des croisements d'histogrammes.

A ce stade, la concentration est sur des descripteurs inférés sur une seule image. Les descripteurs peuvent également être extraits en considérant des frames temporellement consécutifs. Deux descripteurs couramment utilisés exploitent ce notion les descripteurs de la dynamique des personnes avec des filtres rectangulaires (Viola, 2005) et les histogrammes (HOF) de (Dalal et al., 2006b).

La combinaison des descripteurs ci-dessus est liée, on dit alors descripteurs hétérogènes. Comme le démontrent de nombreuses études, cela capte des informations supplémentaires et est donc a priori plus discriminant. [Gerónimo et al., 2007] le démontrent avec des détecteurs de type Haar et EOH, [Wang et al., 2009] avec HOG et LBP ; [Wojek et Schiele, 2008] utilisent des descripteurs de type Haar, Histogramme orienté gradient ainsi que des descripteurs formels, [Walk et al., 2010] avec une concaténation de HOF, HOF et CSS. Des conclusions similaires sont également émises par [Schwartz et al., 2009b] et [Hussain et Triggs, 2010] en utilisant HOG, la fréquence des couleurs et des descripteurs de cooccurrence et des variantes HOG et LBP respectivement. Avec une superposition de HOF, HOF et CSS. [Schwartz et al., 2009b] et [Hussain et Triggs, 2010] sont également parvenus à des conclusions similaires en utilisant respectivement le HOG, la fréquence des couleurs et les descripteurs de cooccurrence, ainsi que les variantes HOG et LBP. Associez des descripteurs hétérogènes appelés caractéristiques intégrales du canal [Dalal, 2009b]. Ils représentent des couches de multiples d'image calculés à l'aide d'une seule transformation d'image pour chaque canal. Dans leur implémentation, [Dalal et al.] ont utilisé trois genres de descripteurs : les images en couleur, les images en dégradé ainsi que les histogrammes. Chaque composant du descripteur correspond à un canal spécifique, et chaque canal utilise l'image intégrale pour calculer efficacement des descripteurs spécifiques,

tels que des sommes locales, des histogrammes et des descripteurs de type Haar. Utilisation efficace des calculs d'image intégrale par chaque canal. Dans la littérature, il existe deux variantes notables des caractéristiques de canal intégrées : les caractéristiques de canal agrégées (ACF) [Dollár, 2014] et les fonctionnalités de canal dépendantes de la décoration locale (LDCF) [Nam, 2014]. ACF emploie dix canaux d'amplitude de gradient orienté histogramme avec six canaux et des canaux couleur LUV. Un seul canal est utilisé sur des blocs pour avoir d'autres canaux avec une résolution minimale. LDCF change l'ACF en intégrant un filtre de décorrélation à chaque canal. Les filtres sont déterminés en tant que vecteurs propres de matrices de covariance spécifiques au canal calculées à partir d'un grand ensemble d'images naturelles.

Considérant un ensemble hétérogène de descripteurs, différentes manières peuvent être utilisées pour construire le descripteur composite final. Quatre stratégies sont ainsi observées :

- Superposition directe [Walk, 2010, Wojek, 2008], où différents descripteurs sont concaténés pour créer un vecteur de descripteur de grande dimension ;
- Sélection de descripteurs classificateur après sélection par boosting [Schapire, 2003] ;
- Arrangements hiérarchiques ad hoc approximatifs [Mogelmose, 2012, Pan, 2013], où une cascade est construite en utilisant des descripteurs bon marché à un stade initial et des descripteurs complexes à un stade ultérieur ;
- Optimisation sur le temps de calcul et la détection [Jourdheuil, 2012, Wu, 2008].

Les travaux actuels se concentrent désormais sur les techniques d'apprentissage en profondeur. Par exemple, DeepPed [Tomè, 2016] utilise LDCF comme une combinaison d'algorithme de proposition de région et de réseaux de neurones à convolution profonde affinés pour extraire des descripteurs. Les réseaux de neurones convolutionnels régionaux (RCNN) sont basés sur les travaux de [Girshick et al., 2014]. Tout d'abord, ils génèrent des propositions de régions indépendantes de la classe qui définissent l'ensemble des détections candidates disponibles et sont basées sur une recherche sélective, puis ils utilisent un réseau de neurones convolutionnels pour extraire une longueur fixe. Le vecteur descripteur est converti en une zone rectangulaire prédéfinie et fixe. Ces paradigmes sont très prometteurs, mais nécessitent l'utilisation d'architecture matérielle.

- **Classification :**

L'étape de classification aboutit à étiqueter chaque fenêtre candidate générée et représentée par le descripteur sous-jacent comme humaine ou non humaine. Ce processus génère une étiquette binaire pour chaque fenêtre et quantifie éventuellement sa confiance dans l'étiquette. Ces classificateurs sont formés par apprentissage supervisé, c'est-à-dire en tirant parti d'échantillons préalablement annotés (fenêtres) à la fois positifs (humains) et négatifs (non humains). Les classificateurs discriminants les plus couramment utilisés pour la détection de personnes sont les supports vecteurs machine et les variantes des classificateurs boostés. On peut également citer l'Analyse Discriminante Linéaire de Fisher (LDA) telle que [Paisitkriangkrai, 2008] et les Réseaux de Neurones Artificiels tels que [Szarvas, 2005, Zhao, 1999, Zhao, 2000] et plus récemment Decision Tree Forests [Tang, 2012].

Les classificateurs boostés, également connus sous le nom de classificateurs d'ensemble, construisent des classificateurs en combinant avec d'autres classificateurs, où chaque classificateur successif se focalise sur des échantillons qui ont été mal classés. Cette méthode est la meilleure parce qu'elle sélectionne automatiquement des caractéristiques discriminantes. On trouve plusieurs types dans la littérature, AdaBoost discret, [Viola, 2004], Adaboost réel, [Gerónimo, 2010a], Logit-Boost, [Tuzel, 2008]. Par conséquent, un avantage du boosting est la sélection non aléatoire des classificateurs faibles. Dans la littérature, les classificateurs boostés sont presque toujours intégrés dans des architectures de cascades d'attention, également appelées cascades de rejet, avec la forme d'un arbre déséquilibré [Viola, 2004]. Chaque nœud de la cascade est entraîné avec un sous-ensemble des échantillons d'apprentissage afin de lui associer le meilleur classifieur.

2.2.4.3. Détecteurs importants

Comme mentionné précédemment, le détecteur consiste en une association de modèles humains, de descripteurs, de classificateurs et du processus habituel de suppression des non-maxima. Cette section se concentre sur les détecteurs de vision courants et/ou efficaces qui nous permettent de calibrer nos propres détecteurs. Nous nous concentrons sur les six détecteurs décrits ci-dessous car ils sont cohérents dans la littérature et couvrent un large éventail au regard de notre classification : HOG-SVM [Dalal, 2005], DPM [Felzenszwalb, 2010], ACF [Dalal, 2014], LDCF [Nam, 2014], DeepPed [Tomè, 2016] et RCNN [Girshick, 2014]. Leurs caractéristiques sont résumées dans le tableau 2.1.

Tableau 2.1 : détecteurs de personnes

Détecteur	Extraction Caracteristique	Modèle	Classificateur
HOG-SVM [Dalal 2005]	HOG	Holistique	SVM linéaire
DPM [Felzenszwalb 2010]	HOG	Basé sur parties	SVM latente
ACF [Dollár 2014]	ACF	Holistique	Soft-cascade
LDCF [Nam 2014]	LDCF	Holistique	Soft-cascade
DeepPed [Tomè 2016]	Apprentissage profound	Holistique	SVM linéaire
RCNN [Girshick 2014]	Apprentissage profound	Holistique	SVM linéaire

- **Modèle à base de Parties déformables (DPM) :**

Les détecteurs DPM (Felzenszwalb, 2010), également appelés LatSvm- V2 et LatSvm-L2, sont basés sur des modèles partiels. Il use un ensemble composé d'un modèle basé sur les parties déformables et d'une lecture modifiée du descripteur. Le modèle se compose d'un filtre racine et de plusieurs filtres partiels, dont le score sur la fenêtre est déterminé comme la somme du score du filtre racine plus le score de chaque filtre partiel, quel que soit le plus grand La position de la pièce moins le coût de déformation, ce qui pénalise la déviation de la position de l'image de la pièce idéale par le filtre racine. Il est appris en utilisant des données partiellement étiquetées avec des SVM latentes. La boîte englobante finale de la détection est déterminée par une fonction de cartographie apprise contenant les positions des parties détectées.

- **Caractéristiques des canaux agrégés (ACF) :**

Ce détecteur est basé sur un faible coût CPU et basé sur le concept de caractéristiques de canal, il surpasse de nombreux détecteurs et se trouve dans de nombreuses bases de données publiques [Dalal, 2014]. Il est basé sur le descripteur de type Caractéristiques des canaux agrégés, le classificateur très léger en cascade et la représentation holistique de la personne. Le classificateur est formé par AdaBoost et des arbres de décision de profondeur deux sur ces descripteurs.

- **Histogramme de Gradient Orienté (HOG-SVM) :**

Ce détecteur présenté par [Dalal et Triggs, 2005], est un détecteur historique et représente une excellente référence. Il est basé sur le descripteur de type HOG et le classificateur SVM linéaire. Les modèles d'apprentissage sont basés sur des abstractions holistiques. La sortie de détection est filtrée par une technique de suppression des non-maximums (NMS) de maxima par paires (PM), qui supprime la boîte englobante la moins probable pour toute paire de détections qui se chevauchent.

- **Réseaux de neurones Convolutifs basés sur la région (RCNN) :**

Le détecteur de personne RCNN est proposé par (Girshick etal., 2014). Un système qui englobe trois parties, la première partie génère des suppositions de régions indépendantes des catégories. Ces hypothèses définissent l'ensemble des détections candidates disponibles pour le détecteur et sont basées sur une recherche sélective. La deuxième partie est un grand réseau neuronal convolutif qui extrait un vecteur descripteur de longueur fixe de chaque région. La troisième partie est un SVM linéaire spécifique qui classe chaque vecteur descripteur comme personne ou non.

- **Caractéristiques de Canal Décorrélées Locales (LDCF) :**

Les détecteurs LDCF [Nam, 2014] sont également basés sur des descripteurs de caractéristiques de canal. Mais à la place, il applique une étape préalable de décorrélation des descripteurs puis construit un classificateur sur ces bases orthogonales.

- **RCNN Rapide :**

Le détecteur fast-RCNN de [Ren et al., 2015] substitue la première partie de RCNN par la partie réseau de proposition de région, Il prédit simultanément la boîte englobante de l'objet et le score de corrélation pour chaque emplacement de boîte. Le RPN est formé pour donner des propositions de zones appropriée, qui sont ensuite utilisées par le RCNN pour la classification. Les modules RPN et RCNN peuvent partager des descripteurs convolutionnels.

- **Réseau de Neurones Convolutifs Profonds :**

Le détecteur DeepPed [Tomè, 2016] est basé sur l'apprentissage en profondeur, c'est-à-dire que les couches inférieures d'un réseau de neurones convolutifs extraient des caractéristiques/descripteurs discriminants et la classification finale effectuée par un SVM. Le réglage du réseau et des paramètres associés on les trouve bien expliqués dans [Tomè, 2016].

2.2.4.4. Description des bases de données

Au fil des années, un certain nombre d'ensembles de données ont été proposés pour créer des repères normalisés permettant d'évaluer différentes techniques de classification. Cependant, lorsque la portée de la variabilité de ces ensembles de données est faible ou limitée à un seul contexte, la capacité des méthodes les plus performantes à se généraliser aux scénarios du monde réel est discutable [Pinto et al., 2008]. Alors que la communauté de la vision par ordinateur vise

à augmenter de plus en plus la taille des ensembles de données d'images annotées, les données elles-mêmes peuvent contenir des biais appelés biais d'ensembles de données.

Plusieurs ensembles de données de vision par ordinateur sont compilés à l'aide des grandes quantités d'images prises sur Internet [Deng et al., 2009, Everingham et coll., 2009, Fei-Fei et coll., 2007, Torralba et coll., 2008] qui partagent tous une source commune de biais d'ensemble de données. Autrement dit, lorsqu'un humain contrôle la caméra lorsqu'il prend l'image, il ne prend généralement que des photos de ce qui est intéressant, nouveau ou artistique. Ceci est illustré par l'un des jeux de données d'images les plus populaires et les plus grands connus sous le nom d'ImageNet [Deng et al., 2009], montré à la figure (2.5).

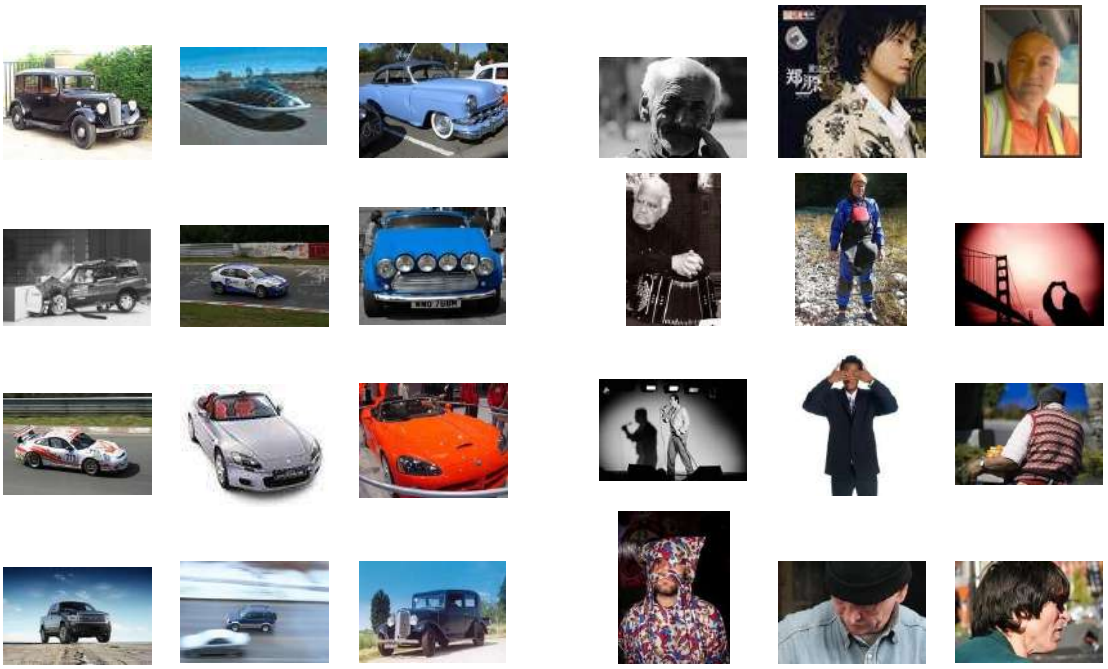
À savoir, le biais de sélection est indiqué en observant que les exemples de voitures sont principalement des voitures anciennes ou de course. Le biais de capture se présente sous la forme d'avoir les objets centrés dans l'image et couvrant la majorité de l'image. Ce biais de capture axé sur l'objet se traduit également fréquemment sur un seul objet visible dans la plupart des images. Les ensembles de données de [Dollar et al., 2009b, Geiger et coll., 2013] n'ont pas été collectées à l'aide de phrases (requêtes) de recherche sur internet ou d'une caméra portative, mais plutôt d'une caméra montée sur un véhicule dans des scénarios urbains. Dans, [Dollar et coll., 2009b] n'est annoté que pour les piétons et non les véhicules alors que [Geiger et al., 2013] contient des annotations pour les personnes et les véhicules, mais avec beaucoup moins d'annotations.

- **Base de données publiques INRIA :**

La base de données de personnes INRIA, présentée par [Dalal et Triggs, 2005], est largement exploitée dans la littérature afin d'étalonner les détecteurs de personnes. Elle se divise en deux formats :

1. Des images brutes (avec et sans cibles) et leurs annotations correspondantes (boîtes englobantes sur l'image entière) ;
2. Des échantillons (extraction des boîtes englobantes de l'image) positives et négatives de personnes. Chaque sous-ensemble est subdivisé en données d'apprentissage et de test.

La base d'apprentissage du premier format inclut 614 images positives (avec présence de personnes) et 1218 images négatives (sans personne). La base de test comprend 288 images positives et 453 images négatives. Quelques exemples d'images sont illustrés figure (2.6.).



Images des véhicules Images des personnes

Figure (2.5) : Sous-ensemble d'images de [Deng et al., 2009] utilisé dans la tâche de détection Du défi de reconnaissance visuelle à grande échelle ImageNet 2014 pour illustrer la sélection et la capture des ensembles de données.

La base d'apprentissage du deuxième format comprend 2416 échantillons positifs (versions originales et en miroir) et 1218 images négatives. Les échantillons ont une résolution de 160 × 96. Mais, la taille réelle de la boîte englobante des personnes est de 128 × 64. Le surplus sert à minimiser l'effet de bordure. La base de tests contient 1132 échantillons positifs et 453 images négatives, les échantillons ont une résolution de 134 × 70. Quelques échantillons de cette base sont donnés dans figure (2.7). Les échantillons positifs dans les bases d'apprentissage et de test sont obtenus à partir d'images de la vie courante et annotées manuellement.

- **Base de données CALTECH :**

La base de données contient environ 250 000 images de résolution 640 × 480 obtenues en conduisant des véhicules en circulation normale [Dollár, 2012]. La base de données est annotée de 350 000 boîtes englobantes, dont 2 300 piétons. La base de données se compose de 11 différentes banques (S0-S10), 6 banques d'apprentissage (S0-S5) et 5 banques de test (S6-S10). La figure (2.8) montre quelques exemples.

(a) : Image d'apprentissage avec personne

(b) : Image d'apprentissage sans personne

(c) : Image de test avec personne

(d) : Image de test sans personne

Figure (2.6) : Exemples d'images de la base INRIA.

(a) : Échantillons d'apprentissage positifs

(b) : Échantillons d'apprentissage négatifs

(c) : Échantillons test positifs.

(d) Échantillons test négatifs.

Figure (2.7) : Exemples d'échantillons de la base de données publique INRIA

(a) : Exemple d'image d'apprentissage.

(b) : Exemple d'image de test.

Figure (2.8) : Exemples d'images de la base de données publique CALTECH

Les boîtes englobantes annotées contiennent également des caractéristiques d'aspect piéton qui permettent de catégoriser les évaluations : à faible distance (piéton de plus de 80 pixels de hauteur), à moyenne distance (piétons de 30 à 80 pixels de hauteur), à grande distance (de moins de 30 pixels de haut), sans occultation (piétons non occultés de plus de 50 pixels), occultation partielle (piétons avec occultation partielle 1% à 35% de superficie occultée), forte occultation (piétons avec occultation d'une surface de 35 à 80%) et raisonnable (50 pixels ou plus grands piétons sans/avec occultation partielle).

2.2.5 Mesures d'évaluation de la détection

Dans un problème de décision binaire (c'est-à-dire que l'objet cible est présenté ou non), un classificateur attribue une étiquette positive ou négative à chaque échantillon de données présenté. Lorsque l'on considère les performances d'un tel système, deux types d'erreurs sont possibles. Une erreur de type I ou un Faux Positif (FP) se produit lorsqu'une détection ne correspond à aucun objet réel, tandis qu'une erreur de type II ou un Faux Négatif (FN) est l'échec de la détection de la présence réelle de l'objet cible. Le tableau 2.1 illustre les combinaisons potentielles des réponses du système et de la présence de vérité au sol d'un objet.

Tableau 2.1 : Matrice de confusion des réponses de détection.

	Objet actuel	Pas d'objet
Détection	TP	FP
Pas Détection	FN	TN

Un ensemble de données avec des étiquettes vraies connues est utilisé pour évaluer les performances d'un système de détection ou d'un algorithme de classification. Les véritables étiquettes sont généralement acquises grâce à l'étiquetage manuel de chaque échantillon par un expert humain dans le domaine. La réponse prévue du classificateur pour tous les échantillons

de l'ensemble d'évaluation est comparée aux étiquettes vraies. Compte tenu de la matrice de confusion du tableau 2.1, les métriques du taux vrai positif (TPR) et du taux faux positif (FPR) sont définies comme suit :

$$TPR = \frac{|TP|}{|total\ des\ positifs\ réels|} = \frac{|TP|}{|TP|+|FN|} \quad (2.1)$$

$$FPR = \frac{|FP|}{|total\ des\ négatifs\ réels|} = \frac{|FP|}{|FP|+|TN|} \quad (2.2)$$

Le TPR mesure la fraction des échantillons positifs correctement classés (à partir du total des positifs réels), tandis que le FPR mesure la fraction des échantillons négatifs qui sont mal classés comme positifs (à partir du total des négatifs réels). Par conséquent, un classificateur parfait devrait avoir 100% TPR et 0% FPR. Notons que ces métriques fonctionnent conjointement plutôt qu'isolément car il est trivial de marquer parfaitement l'un ou l'autre individuellement au détriment de l'autre. Par exemple, si un classificateur devait toujours renvoyer une réponse positive, tous les échantillons positifs réels pourraient être correctement identifiés, ce qui entraînerait un $TPR = 100\%$, tandis que tous les échantillons négatifs réels sont mal classés comme positifs, ce qui entraînerait un $FPR = 100\%$. De même, les deux métriques seraient de 0% si le classificateur renvoyait toujours négatif.

Les performances d'un classificateur donné en termes de TPR et de FPR sont généralement exprimées de manière illustrative à l'aide de l'espace Caractéristique de l'Opérateur Récepteur (ROC). Cet espace est défini comme un tracé sur le plan xy avec FPR sur l'axe x et TPR sur l'axe y. Si un classificateur produit un score continu (c'est-à-dire une probabilité), cette valeur peut être souillée à différents niveaux pour générer une courbe dans l'espace ROC. Cela permet de réduire les deux mesures en un seul score de performance en définissant un TPR ou un FPR souhaité et en mesurant les performances de l'autre mesure. La fixation du TPR ou du FPR est généralement spécifique à l'application, par exemple si un détecteur d'obstacles fait partie d'un système d'aide à la conduite alors un faible FPR est souhaitable, tandis que si on l'utilise comme capteur primaire dans un système autonome, un TPR élevé est souhaité. Alternativement, la performance globale peut être évaluée en considérant la surface totale sous la courbe ROC.

Une alternative courante pour évaluer quantitativement les performances d'un classificateur de détecteur binaire consiste à utiliser les métriques de précision et de rappel. Dans cet espace métrique, il est souhaitable de maximiser les deux métriques car la précision et le rappel sont définis comme suit :

$$precision = \frac{|TP|}{|TP|+|FN|} \quad (2.3)$$

$$recall = \frac{|TP|}{|TP|+|FP|} \quad (2.4)$$

Ici, le rappel est le même que le TPR, tandis que la précision est la fraction des positifs correctement classés sur le nombre total de prédictions effectuées. La précision et le rappel sont généralement privilégiés par rapport aux métriques ROC lorsqu'ils sont évalués sur un ensemble de données fortement biaisé. Une vue détaillée de la relation entre ces deux méthodes est fournie par [Davis et Goadrich, 2006].

F-measure ou F-score est utilisé pour mesurer la performance globale d'un classificateur en évaluant la moyenne harmonique de précision et de rappel comme :

$$F = (1 + \beta^2) \frac{precision.recall}{(\beta^2.precision)+recall} \quad (2.5)$$

Où β est un poids utilisé pour souligner soit la précision ($\beta > 1$), soit le rappel ($\beta < 1$). Typiquement, le score F1 est utilisé pour évaluer un système de détection où une importance égale est accordée à la fois à la précision et au rappel. Si le procédé de détection est capable d'estimer la probabilité ou un score de confiance pour chaque détection, la précision et le rappel peuvent être utilisés pour générer une courbe en ajustant le seuil de détection qui peut également être utilisé pour trouver l'équilibre approprié entre précision et rappel.

De plus, dans le paramètre classes multiples, la Précision Moyenne moyenne (mAP) est souvent utilisée pour agréger le score en calculant l'aire moyenne sous la courbe de rappel de précision pour chaque classe. Enfin, dans le cas d'un rappel élevé, le taux d'oubli moyen est devenu populaire [Dollar et al., 2009a, b].

2.3. Suivi Visuel d'objets

Le suivi est un sujet important en vision par ordinateur et il est étudié depuis plusieurs décennies. Malgré les recherches approfondies dans ce domaine, les problèmes liés à la variation de l'éclairage, au changement de point de vue, à l'occlusion partielle et à la déformation de l'objet ainsi qu'aux mouvements complexes de l'objet et de la caméra occupent le terrain. Plusieurs publications examinent les contributions importantes et récentes à cet art [Jalal, 2012, Li et al., 2013, Wu et al., 2013, Yang et al., 2011, Yilmaz et al., 2006]. Cette section décrit les concepts clés derrière les contributions les plus notables (pour relever les défis susmentionnés) de la littérature sur le suivi visuel à cible unique avant d'explorer les extensions multi-cibles.

2.3.1. Suivi d'Un Seul Objet

Un tracker visuel est généralement initialisé avec une observation de l'objet d'intérêt dans une seule image et le tracker est responsable du maintien de la connaissance de l'emplacement et de l'étendue de l'objet dans les images suivantes. L'observation initiale est généralement représentée sous la forme d'une boîte englobante contenant l'objet d'intérêt sélectionné soit par un utilisateur, soit par l'une des méthodes de détection automatique présentées à la section (2.1). Les principaux blocs fonctionnels pour la construction d'un tracker visuel sont : les estimateurs de mouvement, les modèles de description d'objets et de contextes, le mécanisme de décision (i.e. classificateur) et plus récemment une stratégie d'adaptation du modèle (c'est-à-dire l'apprentissage en ligne).

Des enquêtes récentes sur les techniques les plus remarquables de suivi visuel ont été publiées dans [Wu et al., 2013, Yang et Wang, 2011] avec [Salti et al., 2012] montrant des comparaisons de performances dans différentes conditions. Bien que la majorité des méthodes de ces enquêtes concernent le suivi d'une cible unique, elles offrent des informations précieuses sur la mise à jour efficace des modèles d'apparence des objets en ligne. De plus, les trackers visuels de pointe sont conçus avec les considérations suivantes [Yang et al., 2011] :

- Robustesse à l'encombrement, à l'occlusion, aux changements d'éclairage et aux mouvements complexes.
- Adaptable au changement de contexte et d'apparence d'objet.
- Efficacité de calcul permettant le traitement en temps réel des flux vidéo en direct.

Ces considérations sont abordées dans le choix des algorithmes utilisés dans les principaux blocs fonctionnels d'un tracker visuel.

[[Bradski, 1998] a proposé une version adaptée de l'algorithme de décalage moyen [Fukunaga et Hostetler, 1975] en codant la valeur de teinte d'un objet cible sous forme d'histogramme qui est constamment mis à jour avec chaque image. Plus récemment, [He et al., 2013] propose un nouvel histogramme sensible à la localité où des valeurs à virgule flottante sont ajoutées à plusieurs compartiments spatialement locaux qui se désintègrent de manière exponentielle en fonction de la distance par rapport à l'emplacement du pixel. Cette modification a montré une amélioration significative du suivi visuel.

De plus en plus, l'apprentissage automatique est utilisé pour entraîner progressivement un détecteur basé sur l'apparence afin de capturer l'apparence évolutive de l'instance d'objet spécifique au fil du temps. [Kalal et al., 2012] utilise explicitement la trajectoire prévue pour

identifier les détections manquées et une structure spatiale similaire est utilisée pour identifier les faux positifs. Toutes les erreurs identifiées dans le détecteur sont utilisées pour corriger le détecteur grâce à l'apprentissage en ligne. Méthodes basées sur des sous-espaces de faible dimension [Ross et al., 2007] ou SVM structurées [Hare et al., 2011] ont également été utilisés pour mettre à jour progressivement les modèles d'apparence de l'objet suivi en ligne. Bien que ces approches aient la propriété souhaitée d'apprendre en ligne pour améliorer le suivi au fil du temps, les contraintes utilisées pour identifier la fausse détection dans [Kalal et al., 2012] et la production structurée de [Hare et al., 2011] ne s'étendent pas à plusieurs objets. Au cours des dernières années, un certain nombre de défis de suivi [Kristan et al., 2013, 2014, 2015, 2016] ont été établis pour mieux quantifier et accélérer les progrès dans le suivi visuel. Cependant, ces défis se concentrent uniquement sur le cas du suivi d'un seul objet pour n'importe quelle classe arbitraire, alors que les travaux présentés dans cette thèse vise davantage le suivi d'objets multiples.

2.3.2. Suivi d'Objets Multiples

Cette section résume les études directement liées aux objectis de nos travaux, en particulier le suivi d'objets multiples basé sur la vision ou MOT. L'une des composantes les plus difficiles du MOT consiste à combiner des détections image par image pour estimer les trajectoires les plus probables d'un nombre inconnu de cibles, y compris leurs entrées et départs vers et depuis la scène [[Berclaz et coll., 2011, Maggio et Cavallaro, 2009].

Les approches classiques de suivi de cible combinent généralement un cadre d'association de données avec un estimateur d'état optimal pour évaluer les trajectoires de différents objets et prédire leur état à l'avance. Les méthodes alternatives utilisent généralement des algorithmes d'apprentissage sophistiqués pour le suivi par détection en utilisant des représentations visuelles riches des objets pour résoudre le problème d'association de données [Jalal, 2012].

- **Suivi visuel par détection :**

En raison de l'amélioration récente des algorithmes de détection d'objets, de nombreux trackers visuels adoptent un paradigme de suivi par détection. [Aeschlimann et al., 2010] ont montré que la segmentation joue un rôle essentiel dans la robustesse et la performance du suivi, en particulier pour le suivi de plusieurs cibles qui se chevauchent. De même, [Chen et Corso, 2010] combinent à la fois l'apparence et le flux optique pour propager des étiquettes entre des images dans une séquence vidéo. [Wu et Nevatia, 2007] associent directement des détections en

utilisant des informations provenant de la combinaison de détecteurs basés sur des pièces et du recours à un tracker de décalage moyen lorsqu'aucune association de données n'est trouvée. [Zhang et van der Maaten, 2013] intègrent des contraintes spatiales pour préserver la structure de la scène entre les trames via un cadre de structures picturales. [Felzenszwalb et coll.,2010], formation conjointe de classificateurs d'objets individuels et mise à jour des constantes structurelles avec un apprentissage SVM en ligne. Les approches à base partielle sont sujettes à un surajustement en raison de la flexibilité du modèle.

[Yao et al., 2013] résoudre ce problème en utilisant une chaine de formation en deux étapes comprenant une étape de suivi des pièces, ensuite l'estimation des paramètres de corrélation des objets et des pièces.

Le problème du suivi multi-cibles peut également être décomposé en tant que problème d'optimisation discret et continu [Andriyenko et al., 2012], [Milan et coll., 2013]. L'attribution de détections à des tracklets nouveaux ou existants ou à une identification en tant que fausse alarme est intrinsèquement dans le domaine discret, tandis que l'estimation optimale des états cibles (tels que la position, la taille et la vitesse) est intrinsèquement un problème d'estimation d'espace d'états continu. [Andriyenko et al., 2012] alterner entre l'ajustement de modèles de trajectoires polynomiales par morceaux pour cibler des hypothèses et la mise à jour de l'association de données en tenant compte des coûts globaux de trajectoire et d'étiquetage. [Milan et al., 2013] introduire une représentation mixte de champs aléatoires conditionnels discrets et continus (CRF) pour évaluer simultanément l'association des données et l'estimation de la trajectoire tout en imposant deux contraintes physiques. La première garantit que deux trajectoires continues ne doivent pas se chevaucher dans l'espace et dans le temps. La seconde est une contrainte d'exclusion mutuelle qui définit généralement qu'une seule détection doit être associée à une hypothèse à piste unique. Ensemble, ces contraintes aboutissent à des trajectoires plausibles.

- **Association de Données :**

Le suivi multi-objets peut être réalisé en détectant des objets dans des trames individuelles, puis en reliant les détections entre les trames. Une telle approche peut être rendue très robuste à l'échec occasionnel de détection : Si un objet n'est pas détecté dans une trame mais se trouve dans les précédentes et suivantes, une trajectoire correcte sera néanmoins produite. En revanche, une détection faussement positive dans quelques images doit être ignorée. Cependant, lorsqu'il s'agit d'un problème de cibles multiples, l'étape d'association des données entraîne un

problème d'optimisation difficile dans l'espace de toutes les familles de trajectoires possibles. Les méthodes de recherche ou d'échantillonnage gourmandes décrites dans cette section abordent ce problème en trouvant une solution à l'optimum global.

Lorsque plusieurs cibles doivent être suivies, l'attribution de la mesure appropriée à la trajectoire d'objet correspondante peut devenir une tâche difficile appelée association de données. Traditionnellement, cela a été résolu en utilisant le Suivi d'Hypothèses Multiples (MHT) [Cox et Hingorani, 1996, Reid, 1979] ou les Filtres d'Association de Données Probabilistes Conjoints (JPDAF) [Bar-Shalom, 1987, Schulz et al., 2003], qui retardent toutes deux la prise de décisions difficiles alors que les objets sont à proximité. Dans leur forme pure, la complexité combinatoire de ces approches est exponentielle en nombre d'objets suivis, ce qui les rend non conforme pour des applications en temps réel dans des environnements hautement dynamiques avec de nombreuses cibles. Cependant, en incorporant des heuristiques basées sur l'apparence dans le coût d'affectation [Kim et al., 2015] et des approximations appropriées au problème d'optimisation [Rezatofighi et al., 2015], il a récemment été démontré que MHT et JPDAF restent compétitifs avec seulement une fraction des exigences de calcul précédentes.

Lorsque l'on considère uniquement des correspondances un à un modélisées comme une correspondance de graphes en deux parties, des solutions optimales universelles telles que l'algorithme hongrois [Kuhn, 1955] peuvent être utilisées [Huang et al., 2008, Perera et al., 2006]. Des méthodes approximatives telles que la Chaîne de Markov Monte Carlo Data Association (MCMCDA) ont montré des résultats prometteurs dans le suivi d'un nombre variable d'objets [Oh et al., 2004, Yu et al., 2007] tout en tolérant les détections manquées et les faux positifs. [Giant Collins, 2008] propose un modèle bayésien hiérarchique pour déduire à la fois les paramètres optimaux et les partitions de tracklet à partir de données non étiquetées dans le cadre du MCMCDA. Récemment, [Zamir et al.,2012] incorporer toute la période pour résoudre le problème d'association de données pour un objet à la fois en modélisant les associations inter-trames d'un seul objet.

- **Estimation de la Prédiction de Mouvement :**

Lors de l'attribution de détections à des objets suivis existants dans un cadre de réseau bayésien dynamique, il est avantageux de considérer l'état prédit de cibles individuelles à l'aide d'un estimateur bayésien récursif. L'estimateur le plus connu est probablement le filtre de Kalman [Kalman, 1960], qui estime l'état réel d'un système linéaire à partir d'une série d'observations contenant un bruit gaussien et un modèle de mouvement connu. Les variantes de

Filtre de Kalman Etendu (EKF) et de Filtre de Kalman non Parfumé (UKF) [Julier et Uhlmann, 2004] peuvent être appliquées à des modèles de mouvement et de mesure non linéaires en linéarisant autour de l'estimation actuelle ou en échantillonnant autour de la moyenne attendue pour estimer la moyenne réelle et la covariance. Ces méthodes reposent sur une hypothèse de Markov et comportent le risque associé de s'éloigner de la cible correcte.

Filtres à particules [Gordon et al., 1993, Isard et Blake, 1998] peuvent gérer un bruit non gaussien en utilisant un ensemble de « particules » pour simuler des perturbations aléatoires d'un état régi par le modèle de mouvement. Ces filtres prédictifs sont généralement appliqués pour estimer indépendamment l'état de chaque objet individuel. Cependant, comme le filtre à particules à la capacité de traiter des distributions multimodales, plusieurs objets, ainsi que plusieurs hypothèses, peuvent facilement être suivis simultanément [Khan et al., 2005, Koller-Meier et Ade, 2001].

[Ding et al., 2008] ont montré qu'il était possible d'utiliser la dynamique pour comparer des pistes entre cibles sans supposer a priori un modèle de mouvement. Récemment, [Dicle et al., 2013] modélisent la dynamique sous-jacente de chaque objet en mouvement avec des auto-régressions linéaires et formulent un cadre d'affectation linéaire généralisé qui fusionne les détections avec la complexité de mouvement la moins combinée. Ces mesures de proximité et d'affinité de mouvement prédites sont considérées comme complémentaires aux approches d'association de données basées sur la similitude d'apparence.

2.3.3. Mesures d'évaluation des Suivi

[Bernardin et Stiefelhagen , 008] propose une paire de mesures mesurant les performances de plusieurs suiveurs d'objets qui se concentrent sur la précision de l'estimation de l'emplacement des objets, la précision de la reconnaissance de la configuration des objets et la cohérence de la propagation des étiquettes d'objets dans le temps. La Précision de Suivi d'Objets Multiples (MOTA) combine tous les faux positifs, faux négatifs et erreurs de commutation d'étiquettes en un seul nombre, tandis que la Précision de Suivi d'Objets multiples (MOTP) mesure le déplacement moyen entre la position de la vérité du terrain et la sortie du tracker. Comme ces métriques combinent les erreurs d'association de plusieurs étiquettes en un seul nombre, avec un nombre distinct pour la précision de la position, elles sont largement utilisées pour évaluer les performances des systèmes de suivi multi-cibles [Jalal, 2012, Milan et al., 2013].

La métrique principale utilisée pour comparer les trackers devrait être la métrique MOTA car elle combine trois sources d'erreur courantes en une seule mesure. Le MOTA est calculé comme suit :

$$MOTA = 1 - \frac{\sum_t (fn_t + fp_t + ids_t)}{\sum_t gt} \quad (2.6)$$

Où t est l'indice de trame, fn_t et fp_t sont des erreurs de détection comme décrit dans l'équation (2.6) tandis que gt est le nombre total d'objets de vérité au sol dans la trame t. L'ids_t est toute erreur lors de la commutation de l'identité d'un objet cible dans la trame t. Remarque la MOTA est généralement représentée en pourcentage $[-inf, 100]$, ce qui peut également être négatif dans les cas où le nombre d'erreurs dépasse le nombre d'objets de la vérité du terrain.

Le score MOTA définit une détection avec un critère de chevauchement de 50% qui ne capture pas la précision de localisation du tracker. Pour une mesure plus détaillée de la façon dont un tracker localise chaque cible, le MOTP est utilisé :

$$MOTP = 1 - \frac{\sum_{t,i} d_{t,i}}{\sum_t g} \quad (2.7)$$

Où $d_{t,i}$ est le chevauchement de la boîte englobante avec l'objet vrai attribué i, alors que c_t représente le nombre des objets correspondants dans l'image t.

Un autre ensemble de mesures de suivi courantes qui capturent la cohérence du suivi sont les mesures de qualité principalement suivies (MT) et principalement perdues (non suivies) (ML) [Li et al., 2009, Wu et Nevatia, 2006]. La MT mesure la partie des objets réels qui sont suivis pendant au moins 80% de leur durée de vie. Alors que le ML désigne la partie des objets vrais qui sont récupérés pour moins de 20% de leur durée de vie. Il est souhaitable qu'un tracker ait une MT élevée et une ML faible.

2.4 Résumé et implications

La détection et le suivi sont deux sujets de recherche bien étudiés et les deux utilisent de plus en plus des représentations de modèles basées sur l'apprentissage automatique. Cependant, la majorité des méthodes utilisent des classificateurs supervisés comme modèle d'apparence qui sont généralement formés dans un processus par lots. En plus de nécessiter des quantités importantes de données sur la formation, des études antérieures ont montré que les modèles d'apprentissage de cette manière sont biaisés par rapport à la source initiale des données.

Les observations et lacunes suivantes identifiées dans la littérature permettent de résoudre la première question de recherche : « *Comment découvrir des objets dont les caractéristiques d'apparence peuvent être inconnues par le détecteur ?* »

- Les méthodes de détection basées sur le mouvement décrites dans la section (2.2.3) utilisent une caméra statique pour modéliser l'arrière-plan et segmenter les objets en mouvement de premier plan. Cette approche sans modèle d'apparence ne nécessite pas de formation préalable et est capable d'identifier des objets en mouvement inédits. Cependant, ces approches reposent sur diverses formes de soustraction d'arrière-plan, ce qui les rend inadaptées aux caméras avec un mouvement important tel qu'expérimenté sur un véhicule.
- La méthode récente de [Guizilini et Ramos, 2013] a utilisé des contraintes épipolaires pour tenir compte du mouvement de l'ego de la caméra et entraîner directement un classificateur pour apprendre l'apparence des parties dynamiques et statiques de la scène.

Grâce aux travaux de recherche existants, ces observations permettent à la réponse à la deuxième question de recherche : « *Comment adapter un détecteur basé sur l'apparence pour un déploiement dans des nouveaux environnements avec des caractéristiques de fond différentes aux données de formation ?* »

- Une caractéristique utile des méthodes de soustraction d'arrière-plan est qu'elles s'adaptent à l'environnement déployé, ce qui les rend robustes aux nouveaux motifs d'arrière-plan. Cependant, ces techniques sont sensibles aux mouvements de la caméra, ce qui les empêche de modéliser les apparences d'arrière-plan au-delà d'une seule scène statique. Pour surmonter ce problème, nous recherchons une représentation d'apparence qui n'est pas ancrée à un emplacement spécifique de l'image.
- Les propositions de détection visent à sélectionner des objets candidats susceptibles d'attirer l'attention sur la reconnaissance d'objets. Il a été constaté que certaines de ces techniques identifiaient systématiquement les régions candidates [Hosang et al., 2014]. Comme ces méthodes se concentrent sur des régions avec des apparences générales d'un objet, il devrait être possible d'apprendre un modèle d'apparence générale pour les distracteurs d'arrière-plan tout en améliorant l'efficacité par rapport à une approche par fenêtre coulissante.

Les observations suivantes recensées dans la documentation sur le suivi permettent de résoudre la troisième question de recherche : « *Comment adapter le coût d'affectation à l'aide des données collectées lors du déploiement* ? »

L'apprentissage en ligne a été utilisé pour le suivi d'un objet unique où un détecteur est formé pour identifier une instance unique d'un objet [Kalal et al., 2012]. Cependant, les contraintes utilisées pour collecter des échantillons d'entraînement lors du déploiement supposent qu'un seul objet d'intérêt existe dans l'image, ce qui rend non trivial l'extension aux problèmes de MOT.

Les contraintes géométriques appliquées à la segmentation de scène auto-supervisée dans la section 2.1.3. fournissent un niveau de robustesse au modèle. Bien que bon nombre de ces contraintes reposent sur une connaissance spécifique de la structure de la scène, certaines des contraintes utilisées dans le suivi visuel d'objets (par exemple, l'exclusion mutuelle [Milan et al., 2013] sont généralement applicables et pourraient être utilisées comme source d'auto-supervision pour adapter un tracker basé sur l'apparence lors du déploiement.

Chapitre 3
Detection d’objets en mouvement

CHAPITRE 3

DETECTION D'OBJETS EN MOUVEMENT

3.1. Introduction

La détection des piétons est nécessaire principalement lorsque l'objectif est de les considérer comme des obstacles à éviter. Dans ce cas d'exigence des méthodes plus générales peuvent être utilisées pour approximer les mouvements des objets. Les méthodes de détection des piétons utilisent des données temporelles et sont donc capables d'extraire des informations autres que celles extraites de détecteurs décrits dans le deuxième chapitre. La soustraction du fond et le flux optique sont deux outils de détection utiles, ils sont décrits dans cette section.

3.2. Soustraction d'arrière-plan

Si l'arrière-plan d'un certain nombre d'images est constant, le premier plan peut être segmenté de l'arrière-plan en calculant la différence entre l'image d'arrière-plan et l'image actuelle. Cependant, en raison des changements dans les conditions d'éclairage et des changements généraux dans la scène, l'hypothèse d'un arrière-plan statique ne tient souvent pas. en raison des conditions d'éclairage variables et des changements généraux de la scène. En permettant des mises à jour plus petites dans l'image d'arrière-plan, il est possible de s'adapter aux changements d'éclairage. Si l'arrière-plan est représenté sous la forme d'un Modèle de Mélange Gaussien (GMM), il est possible de définir des valeurs de pixels et des variances appropriées pour l'arrière-plan dans une scène variable [Z. Zivkovic, 2004]. Lorsque des ombres sont présentes, elles peuvent être classées comme premier plan, ceci est problématique lorsque l'on cherche à déterminer la position d'un objet car l'ombre dépend également de la position de la ou des sources lumineuses. Comme les ombres ne modifient que l'éclairage de l'arrière-plan, il est également possible de segmenter les ombres. Un exemple de soustraction d'arrière-plan est représenté dans la figure (3.1).

Figure (3.1) : Exemple de soustraction d'arrière-plan
[OpenCV. Background subtraction using MOG2, Online; accessed May 24, 2017]

La soustraction d'arrière-plan suppose que les conditions de l'éclairage ne changent pas trop vite, si un objet de premier plan partage la couleur de l'arrière-plan, il se peut qu'il ne soit pas complètement segmenté. Si l'on essaie de distinguer des objets séparés en utilisant des composants connectés sur la sortie, un post-traitement supplémentaire est utile pour s'assurer que les corps sont entièrement connectés, par exemple les composants de la tête et du corps ne sont pas séparés en raison du bruit. Étant donné que les objets ne sont pas classés, ils peuvent se présenter sous n'importe quelle forme, cela signifie que les objets adjacents peuvent être regroupés ensemble. Cela peut être déroutant à l'étape suivante lors du suivi des objets.

3.3. Flux optique

En comparant deux images adjacentes dans le temps, un humain peut souvent déterminer facilement le mouvement dans l'image, connu sous le nom de flux optique. Les ordinateurs sont capables, dans certaines conditionss, d'approximer le flux de pixels dans un ensemble d'images consécutives. Le flux optique des pixels qui composent l'objet peut donc être calculé, la position et la vélocité des pixels peuvent alors être extraites, contrairement aux détecteurs décrits dans le chapitre précédent qui ne fournissent que la position, un exemple de flux optique est donné par la figure (3.2).

Figure (3.2) : Exemple de flux optique extrait (les flèches représentent la vitesse du pixel)

Si l'intensité d'un pixel P situé en (x, y) au temps t est donnée par $I(x, y, t)$ et que le pixel est déplacé en $(x+\Delta x, y+\Delta y)$ au temps $t+\Delta t$, la contrainte de constance de luminosité est définie par :

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t) \quad (3.1)$$

En utilisant l'expansion de Taylor, l'intensité s'écrit sous la forme :

$$I(x, y, t) \approx I(x + \Delta x, y + \Delta y, t + \Delta t) = I(x, y, t) + \frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t \quad (3.2)$$

Supposant que le déplacement du pixel est faible, par simplification on obtient :

$$\frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t = 0 \quad (3.3)$$

En divisant par Δt on obtient :

$$\frac{\partial I}{\partial x}\frac{\Delta x}{\Delta t} + \frac{\partial I}{\partial y}\frac{\Delta y}{\Delta t} + \frac{\partial I}{\partial t}\frac{\Delta t}{\Delta t} = 0 \quad (3.4)$$

En utilisant les composantes de vitesse, V_x et V_y du pixel P, permet d'écrire :

$$\frac{\partial I}{\partial x}V_x + \frac{\partial I}{\partial y}V_y + \frac{\partial I}{\partial t} = 0 \quad (3.5)$$

Supposant que les pixels dans le voisinage local de (x, y) ont un petit déplacement approximativement constant, on peut utiliser la méthode de Lucas-Kanade [Bruce D et al.,1981] pour résoudre cette équation à deux inconnues. Le système d'équation est alors de la forme :

$$Av = b \quad (3.6)$$

$$\text{avec ; } A = \begin{bmatrix} I_x(x_1, y_1, t_1) & I_y(x_1, y_1, t_1) \\ I_x(x_2, y_2, t_2) & I_y(x_2, y_2, t_2) \\ \vdots & \vdots \\ I_x(x_n, y_n, t_n) & I_y(x_n, y_n, t_n) \end{bmatrix},\ v = \begin{bmatrix} V_x \\ V_y \end{bmatrix},\ b = \begin{bmatrix} -I_t(x_1, y_1, t_1) \\ -I_t(x_2, y_2, t_2) \\ \vdots \\ -I_t(x_n, y_n, t_n) \end{bmatrix}$$

Où n est le nombre de pixels dans la fenêtre locale utilisée pour les calculs. Le système d'équations (3.6) est généralement surdéterminé et peut être résolu en utilisant les moindres carrés, ce qui permet de déterminer la vitesse du pixel P. L'hypothèse d'un faible déplacement se vérifie si l'on choisit une échelle appropriée et par conséquent, on utilise des pyramides d'échelle.

3.4. Suivi des personnes

Il existe plusieurs façons de réaliser le suivi des personnes, dans ce travail nous avons principalement utilisés des algorithmes de suivi par détection, ces algorithmes utilisent les

informations des objets détectés dans chaque image ensuite ils associent et connectent les détections, cette association peut être utilisée pour extraire la vitesse et la direction des objets en mouvement et rendre le système plus robuste au bruit et aux détections manquées. La position et la vitesse des personnes suivis, sont décrites par ce qu'on appelle l'état de la cible. Il est souvent nécessaire d'estimer l'état et certains états peuvent être cachés, c'est-à-dire non mesurables, comme par exemple la vitesse.

Il existe de multiples façons d'associer les détections, dans ce chapitre trois méthodes sont présentées. Pour que le système ait une application en temps réel, seules les méthodes causales ou les adaptations causales des méthodes sont considérées, il s'agit de méthodes qui n'utilisent pas d'informations futures.

La première méthode utilisée dans notre travail est un tracker basé sur le filtre de Kalman avec association de données effectuée par l'algorithme Hongrois, voir section 3.4.4, a été utilisé comme tracker initial. L'association dans l'algorithme hongrois est seulement effectuée entre les ensembles de détections dans des images consécutives et les prédictions sont effectuées en utilisant un filtre de Kalman.

La deuxième méthode est une extension de l'approche précédente, elle consiste à prendre en compte une fenêtre temporelle plus large, c'est-à-dire l'utilisation un plus grand nombre d'images dans le calcul, ce qui permet d'obtenir une association plus complexe. Ce problème peut être formulé sous la forme d'un graphe et partitionné en trajectoires, par exemple l'optimisation en nombres binaires.

La Densité d'Hypothèses de Probabilité (PHD) est la troisième approche, elle présente des similitudes avec le filtrage de Kalman. Cependant, une association explicite de données n'est pas effectuée mais les probabilités postérieures sont utilisées pour évaluer quelles détections doivent être associées.

3.4.1. Maximum à posteriori

En utilisant les statistiques bayésiennes, c'est-à-dire la théorie qui dit que nous pouvons déduire la probabilité d'un état mesuré comme une probabilité basée sur ce qui s'est passé auparavant. Ceci est formulé dans la règle de Bayes :

$$P(A \mid B) = \frac{P(B|A) \cdot P(A)}{P(B)} \qquad (3.7)$$

A et B sont des résultats d'évènements, $P(A)$ et $P(B)$ sont appelés les probabilités antérieures tandis que $P(A|B)$ et $P(B|A)$ les probabilités postérieures.

Pour considérer cela comme un problème d'association de données, on peut considérer que la probabilité postérieure est la probabilité qu'une détection appartienne à un objet identifié compte tenu des détections déjà associées à sa piste. L'adaptation de ce problème peut prendre plusieurs formes qui se basent sur l'idée de connecter une nouvelle détection à d'anciennes détections/filtres en trouvant le maximum à posteriori.

3.4.2. Estimation bayésienne récursive

L'estimation bayésienne récursive, ou filtre de Bayes, peut être utilisée pour approcher récursivement une distribution de probabilité en utilisant des mesures dans le temps. L'algorithme contient deux étapes, la prédiction et la mise à jour, pour prédire l'état suivant à partir des estimations précédentes et pour corriger l'état interne à partir de la nouvelle mesure. Cette section commencera par la théorie du suivi d'un seul objet et aboutira au filtre de Kalman, plus largement utilisé.

On suppose que l'état du système x est un processus de Markov où l'état suivant ne dépend que de l'état précédent. Les mesures z sont les états observés d'un modèle de Markov caché, si l'état du système au temps k-1 est noté x_{k-1}, l'état suivant est donné par la transition de Markov :

$$x_k = t_k(x_{k-1}, v_{k-1}) \tag{3.8}$$

Où v_{k-1} représente le bruit du processus, la probabilité que le système soit dans l'état x_k conditionné par l'état précédent x_{k-1} peut être exprimée en utilisant la densité de transition de Markov : $f_{k|k-1}(x_k \mid x_{k-1})$

Les observations z_k au temps k du système auront une composante de bruit w_k :

$$z_k = h_k(x_k, w_k) \tag{3.9}$$

Cela signifie que les mesures peuvent contenir certaines erreurs dues au bruit, ceci peut également être exprimé par la fonction de vraisemblance : $g_k(z_k, x_k)$, qui fournit la probabilité de mesurer z_k si la cible est dans l'état x_k, la densité de filtrage (ou postérieure) s'exprime comme suit : $\pi_k(x_k \mid z_{1:k})$. La fonction de vraisemblance peut être utilisée pour extraire des informations sur l'état actuel à partir des mesures jusqu'au moment k. La prédiction de l'état actuel à partir des mesures passées est possible en utilisant la densité de transition et la densité de filtrage précédente :

$$\pi_{k|k-1}(x_k \mid z_{1:k-1}) = \int f_{k|k-1}(x_k \mid x_{k-1})\pi_{k-1}(x_{k-1} \mid z_{1:k-1})dx_{k-1} \tag{3.10}$$

Lorsqu'une nouvelle mesure arrive, la densité de filtrage peut être exprimée à l'aide de la fonction de vraisemblance :

$$\pi_k(x_k \mid z_{1:k}) = \frac{g_k(z_k|x_k)\pi_{k|k-1}(x_k|z_{1:k-1})}{\int g_k(z_k|x)\pi_{k|k-1}(x|z_{1:k-1})dx} \quad (3.11)$$

Les équations (3.10) et (3.11) décrivent la propagation récursive du postérieur en utilisant une densité initiale π_0 [Ba Tuong, 2008].

Le but de tracking un piéton est d'estimer son état, c'est-à-dire déterminer sa position et sa vitesse, afin de faire une prédiction qualifiée et c'est ce que fournit l'équation de la densité de transition de Markov $f_{k|k-1}(x_k \mid x_{k-1})$. L'état réel est inconnu mais en mesurant la position des piétons, par exemple la position du pixel à partir de la sortie du détecteur et en utilisant un modèle de transition qui couple la position et la vitesse, on peut estimer l'état en exploitant la relation de la densité de filtrage $\pi_k(x_k \mid z_{1:k})$. Le détecteur peut émettre plusieurs détections même lorsqu'il suit un seul objet et c'est là que l'étape de prédiction donnée par l'équation (3.10) est importante car elle fournit des informations sur la détection la plus probable.

- **Filtre de Kalman :**

Dans le cas d'un système linéaire gaussien, le filtre de Kalman représente une solution particulière du filtre de Bayes [Simo Srkk, 2013]. Ce filtre permet de décrire les transformations d'état et les observations comme un système linéaire et que le bruit est une distribution gaussienne indépendante de moyenne nulle. Cela permet des calculs rapides et donc une grande applicabilité, la dynamique du système est décrite par :

$$\begin{cases} x_k = F_{k-1}x_{k-1} + v_{k-1} \\ z_k = H_k x_k + w_k \end{cases} \quad (3.12)$$

Où F_{k-1} est la matrice de transition et H_k la matrice d'observation, v_{k-1} et w_k sont des variables de bruit gaussiennes indépendantes de moyennes nulles dont les covariances sont décrites par les matrices Q_{k-1} et R_k. La densité de transition équation (3.13) et la vraisemblance de mesure équation (3.14) sont calculées à partir de densités gaussiennes avec la notation $N(.; m, P)$, m étant la moyenne et P la covariance de la distribution :

$$f_{k|k-1}(x_k \mid x_{k-1})\& = \mathcal{N}(x_k; F_{k-1}x_{k_1}, Q_{k-1}) \quad (3.13)$$

$$g_k(z_k, x_k)\& = \mathcal{N}(z_k; H_k x_k, R_k) \quad (3.14)$$

Pour un mouvement à vitesse constante, à deux dimensions, l'état x est donné par :

$$x = \begin{bmatrix} p_x \\ p_y \\ v_x \\ v_y \end{bmatrix} \quad (3.15)$$

où p_x et p_y désignent les positions (x, y) et v_x et v_y désignent les vitesses dans les directions x et y.

La matrice de transition F pour une vitesse constante et la matrice d'observabilité, dont les états de position sont les seuls observables, sont données successivement par les équations suivantes :

$$F = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.16)$$

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad (3.17)$$

Si tous les états de la matrice H sont observables, H prend la dimension de la matrice d'identité de dimension quatre par quatre.

En décrivant le postérieur à l'aide d'une distribution gaussienne, la densité postérieure au moment $k-1$ est la suivante :

$$\pi_{k-1}(x_{k-1} \mid z_{1:k-1}) = \mathcal{N}(x_{k-1}; m_{k-1}, P_{k-1}) \quad (3.18)$$

Avec, x_{k-1} est l'état, m_{k-1} est la moyenne et P_{k-1} est la covariance. La densité de probabilité prédite équation (3.10) à l'instant k est également décrite avec la distribution gaussienne :

$$\pi_{k|k-1}(x_k \mid z_{1:k-1}) = \mathcal{N}(x_k; m_{k|k-1}, P_{k|k-1}) \quad (3.19)$$

Avec, la moyenne et la covariance prédites sont données par :

$$m_{k|k-1} = F_{k-1} x_{k-1} \quad (3.20)$$

$$P_{k|k-1} = Q_{k-1} + F_{k-1} P_{k-1} F_{k-1}^T \quad (3.21)$$

Lorsque de nouvelles mesures z_k sont disponibles au moment k l'équation (3.11), densité postérieure, est également une distribution gaussienne :

$$\pi_k(x_k \mid z_{1:k}) = \mathcal{N}(x_k; m_k, P_k) \quad (3.22)$$

où ;

$$m_k = m_{k|k-1} + K_k e \quad (3.23)$$

$$P_k = (I - K_k H_k) P_{k|k-1} \quad (3.24)$$

Avec, e, K_k et S_k sont, respectivement, l'innovation, le gain de Kalman et la covariance de l'innovation. Ils sont définis comme suit :

$$e = z_k - H_k x_{k|k-1} \quad (3.25)$$

$$K_k = P_{k|k-1} H_k^T S_k^{-1} \quad (3.26)$$

$$S_k = R_k + H_k P_{k|k-1} H_k^T \quad (3.27)$$

Face à des systèmes non linéaires, on peut former des approximations du système, par exemple par linéarisation. Les techniques existantes sont les filtres de Kalman étendus et non accentués [Simo Srkk, 2013].

Pour adapter le filtre de Kalman au suivi des piétons, les positions des pixels des détections disponibles sont utilisées comme mesures z. Un algorithme d'association, par exemple le plus proche voisin, utilise l'état (position) prédit $x_{k|k-1} = H_{k|k-1} x_{k-1}$ pour associer l'une des nouvelles détections à la cible actuelle. L'innovation (erreur), équations (3.15), (3.26) et (3.27). 3.16, est alors calculée et utilisée pour mettre à jour l'état de la cible. La covariance de la cible est déterminée par la covariance cible est déterminée par les matrices de covariance des bruits de mesure et de processus R_k et Q_k, ce dernier peut être réglé pour favoriser les nouvelles mesures par rapport à l'état actuel du modèle, ou vice versa. Si le système est supposé linéaire et invariant dans le temps, le gain de Kalman de la covariance K_k, convergera vers un état stable.

3.4.3. Algorithme Hongrois avec Kalman

En utilisant la théorie présentée dans la section 3.4.2, on peut déterminer dans quelle mesure une détection précédente correspond à une nouvelle détection et utiliser cette information pour suivre des objets. Lorsqu'il y a plusieurs objets, l'objectif est plutôt de trouver les meilleures associations possibles entre les détections précédentes et les nouvelles détections. Cette situation est analogue au problème classique d'un petit nombre de travailleurs ayant des salaires et des compétences différents pour accomplir un ensemble de tâches. Les mathématiciens introduisent souvent le sujet de l'optimisation avec ce problème. En supposant qu'il y a n travailleurs et une quantité égale de tâches, on peut commencer par tester toutes les combinaisons possibles de travailleurs et calculer comment le travail global est effectué. La résolution de ce problème est $O(n\,!)$, comme le montre la figure (3.8).

(a) Option d'attribution 1 (b) Option d'attribution 2

Figure (3.3) : Deux étapes dans la visualisation de toutes les affectations possibles de trois nœuds à trois autres nœuds (le lien rouge est verrouillé), les options pour les deux autres liens sont affichées, le reste des combinaisons peut être trouvé en déplaçant le lien rouge entre toutes les combinaisons possibles.

Un gros problème à résoudre lorsque la taille des deux ensembles augmentera. Le meilleur algorithme pour résoudre ce problème est l'algorithme de Kuhn-Munkres aussi appelé l'algorithme Hongrois [H. W. Kuhn et al., 1955]. Cet algorithme réduit la complexité temporelle à un temps polynomial $O(n4)$ et a ensuite été modifié pour fonctionner en O(n3) [Jack Edmonds et al., 1972].

La mise en œuvre réelle est plus facile à comprendre lorsqu'on utilise une matrice pour représenter les différents coûts d'utilisation d'un travailleur pour un travail particulier. Cela donne une matrice de coûts comme il est indiqué dans le tableau 3.1.

Tableau 3.1 : Matrice des coûts pour trois travailleurs (t_i) affectés à trois tâches (w_j), (c_{ij}) étant le coût de l'affectation, $i = 1,2,3$ et $j = 1,2,3$.

	t_1	t_2	t_3
w_1	c_{11}	c_{12}	c_{13}
w_2	c_{21}	c_{22}	c_{23}
w_3	c_{31}	c_{32}	c_{33}

Pour trouver l'affectation optimale, on applique les étapes suivantes sur la matrice indiquée dans le tableau précédent [H. W. Kuhn et al., 1955] :

1. Soustrayer la plus petite entrée de chaque ligne de toutes les entrées de cette ligne ;
2. Soustrayer la plus petite entrée de chaque colonne de toutes les entrées de cette colonne ;
3. Tracer des traits à travers les lignes et les colonnes de façon à couvrir chaque zéro en utilisant le moins de lignes verticales et horizontales possible ;

4. Evaluer si l'algorithme est terminé : Si la quantité minimale de lignes utilisées est la même que la quantité de tâches et de travailleurs, dans ce cas $n = 3$, on peut affecter de manière optimale chaque travailleur à une tâche qui correspond à une entrée nulle. Si la quantité minimale de lignes est inférieure à n, l'algorithme n'est pas terminé ;
5. Trouver la plus petite entrée non couverte par une ligne, soustraire ce nombre de chaque ligne non couverte et l'ajouter à chaque colonne couverte. Retournez à l'étape 3.

L'addition et la soustraction de chaque colonne/ligne ne changeront pas la solution optimale et l'algorithme est construit pour trouver des zéros dans chaque colonne/ligne. Chaque itération de l'algorithme donnera une matrice à coût réduit avec au moins un zéro dans chaque colonne. La logique derrière l'utilisation du plus petit nombre de traits pour couvrir les zéros est que si l'on utilise moins que le nombre de lignes et de colonnes, il peut y avoir une situation où l'ordre d'affectation affecte le résultat. Pour éviter cela, l'algorithme itère jusqu'à ce que le nombre minimal de traits requises soit égal au nombre de lignes. Cela garantit que chaque ligne peut être assignée indépendamment des autres assignations.

- **Adaptation du suivi :**

Afin d'utiliser cet algorithme pour le suivi, les travailleurs et les tâches de la section précédente sont remplacés par des détections dans différentes images. Le coût de l'affectation d'une détection à une autre est calculé dans cette thèse en utilisant la distance entre deux détections. La résolution de l'algorithme pour chaque image, les nouvelles détections étant assignées aux précédentes, fournira alors la meilleure correspondance en fonction de leurs positions respectives.

Pour utiliser efficacement l'algorithme Hongrois, il faut que de nouvelles personnes puissent entrer et sortir de la séquence vidéo. Pour résoudre ce problème, nous commençons par nommer chaque détection dans une image précédente une piste. Une nouvelle détection peut être assignée à une ancienne piste, ou elle peut initialiser une nouvelle piste. Il est possible qu'aucune détection ne soit attribuée à une piste. Pour ce faire, nous ajoutons des détections et des pistes fictives. Les détections fictives représentent une piste qui ne reçoit pas de nouveau nœud et une piste fictive représente un nœud qui commence une nouvelle piste. En supposant qu'il y a deux pistes existantes (t_1 et t_2) et trois nouvelles détections (d_1, d_2 et d_3), les nœuds fictifs (Dt_1 et Dt_2) se trouvent dans les colonnes les plus à droite et les pistes fictives (Td_1, Td_2 et Td_3) se trouvent dans les lignes inférieures du Tableau 3.2.

Tableau 3.2 : Matrice des coûts pour trois détections affectées à deux pistes, c_{ij} étant le coût de l'affectation, C_{ua} le coût de la non-affectation d'une piste et nt le coût de la création d'une nouvelle piste par une détection.

	d_1	d_2	d_3	Td_1	Td_2
t_1	c_{11}	c_{12}	c_{13}	C_{ua}	Inf
t_2	c_{21}	c_{22}	c_{23}	inf	C_{ua}
Td_1	C_{nt}	inf	inf	0	0
Td_2	inf	C_{nt}	inf	0	0
Td_3	inf	inf	C_{nt}	0	0

Il peut être avantageux de laisser les pistes continuer à vivre même si elles ne reçoivent pas de nouvelles détections. Cela peut être utilisé pour réduire l'effet de l'occlusion, des détections manquées et bruyantes. Pour ce faire, un filtre de Kalman est utilisé, comme décrit dans la section. 3.4.3, un filtre de Kalman peut être utilisé comme estimateur à la fois pour la position et la vitesse où nous mettons à jour le filtre avec les mesures assignées à la piste spécifique. Avec un réglage correct, ce filtre peut approximer le mouvement continu d'un piéton. Si la position prédite d'une piste est utilisée, il y a plus de chances de trouver les bonnes correspondances lorsque l'algorithme Hongrois est appliqué, voir figure (3.4).

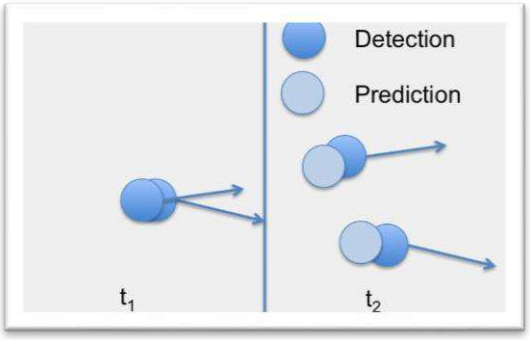


Figure (3.4) : Les détections en deux temps et les prédictions du filtre de Kalman

Nous pouvons constater qu'il est beaucoup plus facile de faire correspondre la détection en t2 à la prédiction en t_2 que de faire correspondre la détection en t_2 et la détection en t_1.

Si une piste ne reçoit plus de nouvelles détections, il est probable qu'il n'y a pas de piéton correspondant à cette piste. L'élimination de ces traces est appelée élagage et s'effectue en tenant

compte du moment où un filtre a été apparié pour la dernière fois à une détection et du nombre de détections auxquelles il a été apparié dans un nombre donné de trames. Il s'agit d'un paramètre de réglage qui est lié à la robustesse du détecteur et à la façon dont les filtres de Kalman sont réglés.

3.5. Conclusion

De nombreuses techniques de détection s’appuient sur le fait que la partie statique de la scène reste inchangée durant la prise de vue et que les changements observés proviennent d’un objet en mouvement, les techniques présentées dans ce chapitre sont des techniques de détection d’éléments mobiles optiques qui s’appuient sur des informations extraites du flux vidéo de la caméra ainsi que sur des informations calculées à priori. L’implémentation et les résultats des algorithmes de détection d’objets en mouvement seront représentés dans les chapitres ultérieurs.

Chapitre 4
Methodologie et Implementation

CHAPITRE 4

MÉTHODOLOGIE ET IMPLÉMENTATION

4.1. Introduction

Les principales étapes de notre méthode sont décrites dans la figure (4.2), mais avant de procéder à l'explication et au détail des différents algorithmes utilisés que ce soit pour la détection et le suivi des personnes ou l'évaluation métrique ainsi que les bases des données utilisées, nous devons tout d'abord revenir brièvement sur quelques recueils dans la littérature sur le suivi multi-objets.

4.2. Algorithme de suivi d'objet multiple (MOT)

La tâche décrite ici équivaut à développer une solution algorithmique fiable et efficace qui permet de suivre des objets se déplaçant dans une séquence vidéo image par image, les algorithmes de MOT doivent être en mesure de :

- Détecter les cibles présentes sur l'image. Dans notre cas, distinguer les piétons du fondde l'image et déterminer leur position exacte. En pratique, les cibles sont repérées par un cadre qui suit leur contours appelé Bounding Box.
- Attribuer un identifiant unique à chaque cible afin qu'elle puisse être suivie sur plusieurs images consécutives et reconstruire sa trajectoire.

Le suivi d'objets, en particulier de piétons, est une tâche complexe en raison des nombreuses difficultés que l'algorithme MOT doit résoudre.

Ces difficultés peuvent se résumer à des problèmes liés aux mouvements de la caméra (changement d'inclinaison, zoom, flou de bouger, etc.), à la diversité de l'environnement (changement de luminosité, reflets...), à la diversité des cibles à suivre (vêtements piétons, auto-masquage d'objets articulés, etc.) ou cohérence du tracker dans le temps (cacher entre objets

d'une même classe peut mettre fin au tracking ou échanger des identifiants entre cibles). La figure (4.1) montre comment le tracker suit un objet image par image.

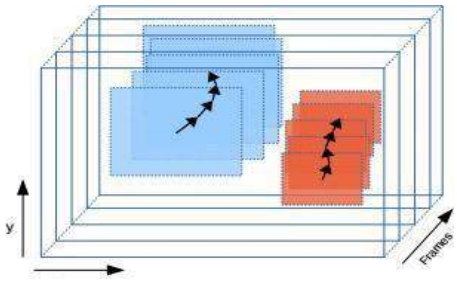


Figure (4.1) : Exemple de suivi dans une vidéo

4.2.1. Distinction entre suivi par identification et par association

Inspirés de la distinction faite dans le recueil d'articles sur le suivi multi-objets entre algorithmes de suivi par identification et de suivi par association (W. Luo et al., 2014). Dans le suivi par identification, la cible est reconnue dès sa première apparition et convertie en un vecteur caractéristique. Ensuite, l'étape de suivi lors de la récupération d'une nouvelle image consiste à placer la cible à proximité immédiate de la capture précédente. Le vecteur de caractéristique extrait de cette manière est comparé au vecteur de caractéristique cible. Ces méthodes ont l'avantage d'être très courtes en temps d'exécution et cohérentes avec la simple inférence de l'algorithme d'extraction de caractéristiques, mais elles nécessitent autant d'inférences que de cibles à suivre. Ceci est gênant dans le cadre d'applications où de nombreuses cibles simultanées sont attendues et où des contraintes de temps réel sont absolument nécessaires. Un exemple de méthode de suivi d'identification est le réseau dit « Siamois » [L. Bertinetto et al., 2016, A. He et al., 2018, L. Leal Taixé et al., 2016, B Avec Lee Al., 2018]. Dans le cas du suivi par association, à chaque image, un détecteur fournit l'ensemble des cibles en une unique inférence, un algorithme d'association se charge alors de coupler l'ensemble de la cible de l'image précédente à la cible de la nouvelle image en minimisant la métrique donnée.

Dans notre application, nous utiliserons des algorithmes de suivi de piétons qui appartiennent à un sous-ensemble d'algorithmes de suivi par association, avec le squelette de la chaîne de traitement représenté sur la figure (4.2). La chaîne de traitement peut être divisée en deux chaînes : « chaîne d'inférence » et « chaîne d'évaluation ». La chaîne d'inférence est linéaire, et les images de suivi des piétons (base de données) sont transmises au détecteur chargé de détecter tous les piétons. Le détecteur fournit à chaque objet une annotation, qui est une boîte englobante entourant l'objet associé à la classe(personne). Ces annotations, et des images

facultatives, sont ensuite transmises au tracker, qui associe un identifiant unique à chaque boîte englobante, formant la trajectoire du piéton. Ainsi, une itération de la chaîne d'inférence génère toutes les boîtes englobantes des objets présents sur l'image et les associe à des identifiants uniques. La chaîne d'inférence est donc la chaîne qui sera implémentée sur le système embarqué final. Dans nos travaux, nous cherchons à quantifier l'efficacité des algorithmes de suivi et à visualiser les résultats du traitement des images. Pour cela, nous ajoutons une chaîne d'évaluation composée de trois briques : l'étape de prétraitement, la métrique d'évaluation de l'étape de détection et la métrique d'évaluation de l'étape de suivi. Ci-après, nous décrivons l'algorithme choisi pour chaque bloc d'algorithme et la base de données d'images de suivi de piétons sélectionné.

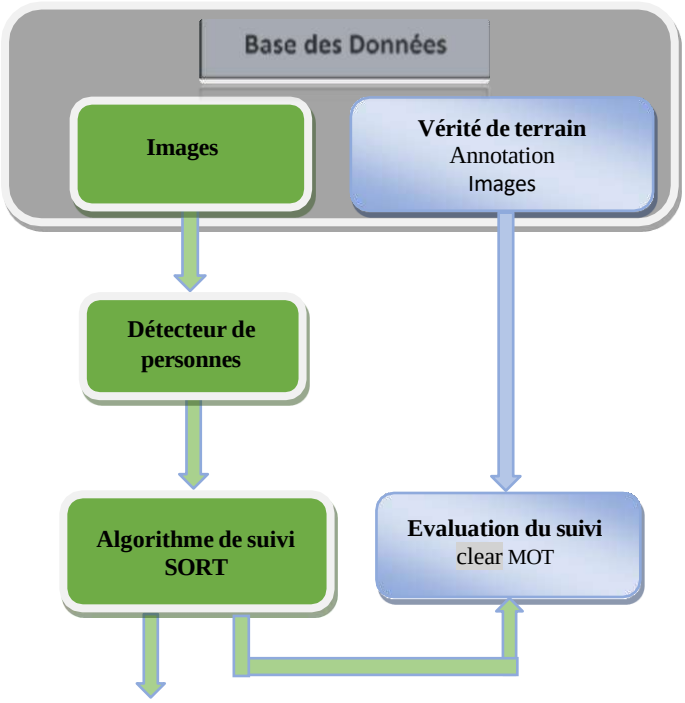


Figure (4.2) : Schéma récapitulatif de la chaîne de traitement implémentée.

4.2.2. Chaine de traitement

L'étape de suivi consiste à attribuer un identifiant unique à chaque objet précédemment détecté, effectué par l'algorithme Simple Online Real-Time Tracking (SORT) [Bewley et al, 2016]. La motivation pour utiliser le tracker SORT est de concevoir un tracker aussi simple que possible pour travailler en temps réel, là où la plupart des trackers développés se basent sur l'apparence des objets pour les retrouver sur différentes trames. L'itération du tracker SORT est divisée en 4 étapes distinctes, résumées dans la Figure (4.2), voir la Section 4.5.3 pour plus de détails.

4.2.3. Sélection de la métrique d'évaluation pour l'étape de tracking

Pour quantifier l'efficacité du tracker, nous avons utilisé les métriques CLEAR [R. Stiefelhagen et al., 2006] couramment utilisées dans le domaine du suivi multi-objets. Lors de la constitution de trajectoires, différentes erreurs d'association peuvent survenir. Ces erreurs se résument dans le cas des métriques CLEAR au taux de Faux Positifs (FP), Faux Négatifs (FN), échanges d'identifiants (ou ID switch, IDsw) et aux fragmentations de trajectoire (Frag) comme présenté dans la figure (4.3).

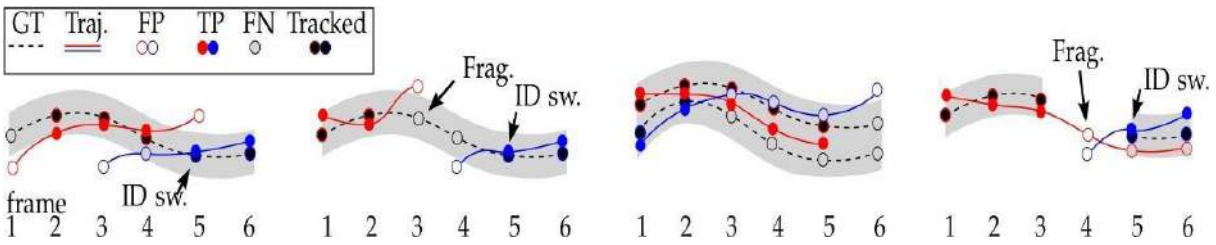


Figure (4.3) : Schéma récapitulatif des différentes erreurs d'association de bounding box à une trajectoire [A. Milan et al., 2016].

Les courbes en pointillé ou en trait plein représentent des trajectoires, véritables ou estimées composées de bounding boxes symbolisées par des points. Ces trajectoires sont représentées dans un espace dans lequel l'axe des abscisses est discret et représente les numéros de frame et les ordonnées représentent la similarité entre des bounding boxes. En lignes pointillés sur ce schéma figure la trajectoire véritable de la cible tel que présenté dans les annotations de la vérité terrain (ou Ground Truth, GT) de l'annexe [A]. Les zones grises caractérisent un ensemble de coordonnées de boîte englobante avec une IOU dont la boîte englobante réelle est supérieure au seuil associé. Les courbes rouges et bleues continues sont estimées par le tracker. Pour chaque image, vous pouvez connecter la boîte englobante à l'orbite (points pleins) ou non (points creux). Une boîte englobante orbitale GT qui n'est pas associée à une orbite estimée générera un faux négatif (FN). Inversement, la boîte englobante d'une orbite estimée qui n'est associée à aucune boîte englobante dans l'orbite GT est un faux positif (FP). Enfin, chaque discontinuité orbitale estimée produit une fragmentation (Frag) et chaque échange orbital produit un changement d'ID switch (IDsw). Le calcul du nombre de ces erreurs individuellement représente une métrique unique. Cependant, au lieu de traiter ces différentes métriques individuellement, il est recommandé d'utiliser une métrique composite appelée précision de suivi multi-objets (MOTA) donnée dans l'équation (4.1).

$$MOTA = 1 - \frac{\sum_t (fn_t + fp_t + ids_t)}{\sum_t g_t} \quad (4.1)$$

MOTA reflète la consistance du tracker dans le temps, elle diminue lorsque le nombre de faux positifs, faux négatifs ou échanges d'identifiants augmente et peut prendre des valeurs négatives. Dans notre cas, nous la multiplions par 100 afin d'obtenir un résultat maximal de 100. À cela s'ajoute la Multi-Object Tracking Précision (MOTP) définie comme l'IOU moyen des bounding boxes traquées et permettant de rendre compte de la précision moyenne des contours des bounding boxes. De plus, on retrouve parmi les métriques CLEAR les notions de rappel et précision comme définies pour les métriques COCO. La raison étant qu'un tracker, comme nous aurons l'occasion de le voir, peut générer ou supprimer de fausses détections et ainsi influencer sur ces deux valeurs.

Par la suite, nous utilisons en supplément la moyenne harmonique de la précision et du rappel, appelée métrique F1, afin de rendre compte de l'évolution moyenne de ces deux critères. L'ensemble de ces métriques sont fournies par la librairie Pymotmetrics GitHub Repository [S1] qui est compatible avec les résultats du concours MOT. Enfin, nous rajoutons à ces métriques le taux de Frames Per Seconds du tracker (FPS tracker) pour rendre compte du temps d'inférence par image du tracker.

4.3. Détecteurs visuels de personnes

Cette section présente les différents détecteurs visuels de personnes qui ont fait l'objet d'une enquête dans le cadre de ce travail. Comme présenté dans le deuxième chapitre, l'état de l'art dans le détecteur visuel de personnes comprend plusieurs détecteurs qui ont des performances de détection, un temps de calcul et une abstraction de modèle différents. Dans notre travail, nous avons sélectionné quatre détecteurs, à savoir : HOG-SVM [N. Dalal, et al., 2005], DPM [P. F. Felzenszwalb et al., 2010], ACF [P. Dollár et al., 2014] et LDCF [W. Nam et al., 2014] Annexe [B].

Pour effectuer des évaluations de suivi par détection planifiées, les caractéristiques pertinentes de ces détecteurs sont résumées dans le tableau 4.1

Tableau 4.1 : Caractéristiques des détecteurs utilisés

Détecteur	Type	Modèle	Classification
HOG-SVM [N. Dalal, et al., 2005]	HOG	Holistic	Linear SVM
ACF [P. Dollár et al., 2014]	Channel Features	Holistic	AdaBoost
LDCF [W. Nam et al., 2014]	Channel Features	Holistic	AdaBoost
DPM [P. F. Felzenszwalb et al., 2010]	HOG	Parts-based	Linear SVM

4.3.1 Histogramme des gradients orientés (HOG-SVM)

Ce détecteur, proposé par Dalal et Triggs [N. Dalal, et al., 2005], est l'un des détecteurs classiques et les plus anciens, ce détecteur calcule des histogrammes locaux de l'orientation du gradient sur une grille dense et utilise la machine à vecteurs de support linéaire (SVM) comme classifieur. Le modèle appris est basé sur une abstraction holistique (corps entier) entraîné sur l'ensemble de données publiques de formation des personnes Inria [N. Dalal, et al., 2005]. Les sorties de détection sont filtrées par une technique de suppression non maximale (NMS) par paires max (PM) qui supprime la confiance sans réponse de chaque paire de détections qui se chevauchent suffisamment. Bien que HOG-SVM ne soit pas actuellement le meilleur détecteur, ses caractéristiques HOG constitutives sont les caractéristiques les plus discriminantes à ce jour [P. Dollár et al., 2012].

4.3.2. Modèle des parties déformables (DPM)

DPM [P. F. Felzenszwalb et al., 2010] est un détecteur basé sur des pièces qui fonctionne en agrégeant les preuves de différentes parties d'un corps pour détecter une personne dans une image. Le détecteur utilise un mélange de modèles basés sur des pièces déformables et une version modifiée des fonctionnalités HOG, il se compose d'un filtre racine (un qui caractérise le corps entier) et de plusieurs filtres partiels ; son score sur une fenêtre candidate est déterminé comme le score du filtre racine plus la somme des scores de chaque filtre de pièce, en prenant le maximum sur les emplacements des pièces, moins un coût de déformation qui pénalise l'écart par rapport aux emplacements idéaux des pièces par rapport au filtre racine. Il a appris à l'aide de données partiellement étiquetées avec une SVM latente, sur le jeu de données
INRIA personne.

La boîte englobante de détection finale est déterminée avec une fonction de mappage apprise qui utilise les positions des pièces détectées. Il utilise une technique NMS basée sur PM. Étant donné que ce détecteur repose sur des parties, il détecte bien les personnes partiellement occluses et conduit à une meilleure localisation.

4.3.3. Caractéristique du canal agrégé (ACF)

Il s'agit d'un détecteur rapide de personne basé sur la l'idée des caractéristiques du canal qui a surpassé plusieurs détecteurs dans divers l'analyse comparative des ensembles de données [N. Dalal, et al., 2005]. Il est basé sur des agrégats de fonctionnalités représentés sous forme de canaux, une variante du classifieur Boosted et une abstraction holistique de la personne. Un canal est une entité par pixel calculée à partir d'un patch correspondant de pixels d'entrée. Il

peut être, par exemple, le composant L de la couleur LUV de l'image d'entrée transformée, ou même un histogramme de chaque quantifié orientation du gradient (un canal par orientation) de l'image d'entrée. ACF utilise dix canaux-magnitude du gradient, HOG (6 canaux) et canaux de couleur LUV. Chaque canal est agrégé sur des blocs pour créer des canaux de résolution inférieure. Le classificateur final a appris à l'aide d'AdaBoost et de la profondeur deux arbres de décision sur ces entités de canal. Le détecteur considéré dans ce travail est entraîné sur l'ensemble de données de personnes INRIA et utilise des NMS à base de PM.

4.3.4. Caractéristiques de canal décorrélées localement (LDCF)

Le détecteur de personnes LDCF [W. Nam et al., 2014] est un détecteur qui se trouve également sur des caractéristiques de canal comme ACF, mais, au lieu de former un classifieur sur les entités directement, il applique une étape de décorrélation au préalable. Le point clé est l'observation que les arbres de décision utilisés dans Boosting, qui utilisent des fractionnements orthogonaux (entité unique), peuvent mieux se généraliser si la corrélation entre les fonctionnalités de canal est réduite. Par conséquent, LDCF modifie ACF en appliquant des filtres de décorrélation par canal. Les filtres sont déterminés comme les vecteurs propres d'une matrice de covariance spécifique au canal calculé à partir d'une grande collection d'images naturelles.

4.4. Suivi d'objets avec le tracker SORT (Simple Online Realtime Tracking)

SORT (Simple Online Realtime Tracking) est une implémentation de suivi par détection qui est orienté vers les problèmes de suivi d'objets multiples (MOT) dans les vidéos. Ce tracker, à l'aide d'un modèle de détection d'objets, détecte les objets dans chaque image et les représente par des boîtes englobantes. Une fois la détection des objets effectuée par le modèle, pour suivre les objets de l'image actuelle, le tracker utilise les objets détectés dans l'image précédente ainsi que dans l'image actuelle. La qualité de la détection a un impact important sur les performances du suivi en temps réel et c'est pourquoi le tracker SORT doit être très bien combiné avec le modèle de détection d'objets. Au cours de cette section, les différentes techniques utilisées pour le suivi en temps réel seront expliquées, notamment le filtre de Kalman et l'algorithme Hongrois. Le tableau 4.2 résume les caractéristiques du tracker SORT

4.4.1. Simple online and real time tracker (SORT)

Le tracker SORT, proposé par [A. Bewley et al., 2016], est un tracker multi-objets léger qui dépend uniquement des détections et du modèle de mouvement dyn-amique sans utiliser de modèle d'apparence cible pour le suivi. Les trackers se concentrent sur une gestion efficace et

fiable des associations frame-frame courantes. En outre, il utilise deux méthodes classiques extrêmement efficaces, le filtre de Kalman et la méthode Hongroise [H. W. Kuhn et al., 1955], pour gérer respectivement les composants de prédiction de mouvement et d'association de données du problème de suivi, le tracker suit chaque cible indépendamment et se rapproche des déplacements inter-images de chaque cible avec un modèle de vitesse constante linéaire qui est indépendant des autres objets et du mouvement de la caméra. Lors de l'affectation de détections à des cibles existantes, la géométrie de chaque zone englobante de cibles est estimée en prédisant son nouvel emplacement dans l'image actuelle.

La matrice du cout d'affectation est ensuite calculée comme la distance intersection-sur-union (IOU) entre chaque détection et toutes les boîtes englobantes prédites à partir des cibles existantes. L'affectation est résolue de manière optimale à l'aide de l'algorithme Hongrois. Dans nos expériences, l'efficacité du tracker est évaluée en utilisant chacun des six détecteurs présentés, chaque variante est précédée du nom du détecteur associé, par exemple SORT-ACF.

Le tableau 4.2 : Caractéristiques du tracker utilisé

Tracker	Échantillonnage	Modèle d'apparence	Modèle dynamique	Association de données	Estimation ponctuelle
SORT (10)	MH	Multi-modèle	Vitesse linéaire (KF)	Hongrois	Moyenne

Avec ;
MH : Echantillonnage Metropolis-Hastings ;
KF : Filtre de Kalman.

4.4.2 Association des données

Pour affecter un objet détecté à une cible précédente, le tracker SORT prédit son nouvel emplacement dans la frame par une géométrie de boîte englobante estimée. La détection de l'objet et la prédiction de la cible précédente sont indépendantes l'une de l'autre, cela signifie qu'elles ont besoin d'une métrique pour associer les deux. La distance IOU entre chaque détection et toutes les boîtes de délimitation prédites des cibles est la métrique utilisée par SORT pour quantifier l'association.

La matrice des coûts d'affectation est une autre métrique chargée d'associer les données et elle est calculée en appliquant la IOU (métrique pour quantifier l'association) mentionnée précédemment. Une fois cette matrice calculée, l'algorithme Hongrois est utilisé pour résoudre

l'affectation entre l'observation prédite. L'algorithme Hongrois est principalement utilisé pour les problèmes d'affectation.

4.4.3 Fonctionnement de l'algorithme Hongrois et du filtre de Kalman

Le tracker SORT ne considère que la taille et la position de la boîte englobante entre deux trames consécutives pour effectuer sa surveillance. Chaque boîte englobante est décrite par un vecteur d'état :

$$x = [u, v, s, r, u'v', s']^T \quad (4.2)$$

Le vecteur (u, v) représente le centre de la boîte englobante, $s = w \times h$ représente son aire, r est le rapport de w/h , w et h représentent respectivement sa largeur et sa hauteur, u′, v′ et s’ sont à leur tour les dérivées temporelles de u, v et s. Pour toutes les boîtes englobantes d'une trame, la vitesse de l'objet dans la trame est supposée constante entre deux trames consécutives, et le bruit sur la variable du vecteur d'état est supposé être gaussien.

Le principe de fonctionnement du SORT se base sur un filtre de Kalman prédictif et en particulier vecteurs d’état des bouding boxes avec hypothèse de vitesse constante des cibles dans le repère image.

Comme mentionné précédemment, le tracker SORT passe par quatre étapes distinctes, et nous considérons un état initial où un ensemble de boîtes englobantes conservatrices est attaché à la trajectoire d'un seul objet.

- La première étape de SORT consiste en une réception des bounding boxes nouvellement détectées par le détecteur sur la nouvelle frame (désignées par la suite par bounding boxes actuelles), ce qui correspond à l’étape d’observation du filtre de Kalman prédictif.
- Par la suite, en supposant que la vitesse des cibles reste inchangée entre les frames précédentes et actuelles, la bouding box estimée de chaque cible est prédite dans la nouvelle frame à l’aide d’un schéma d’Euler. Ce qui correspond à l’étape de prédiction du filtre de Kalman :

  $$\hat{X}_k^- = x_{k-1} + Bu_k \quad (4.3)$$

  $$P_k^- = AP_{k-1}A^T + Q \quad (4.4)$$

  D’où ; k représente l’instant discret, x_k est l’état, A et B sont les matrices de transition d’état, u signifie une entrée (qui n’est pas prise en compte dans le système de suivi d’objet actuel), P est la matrice de covariance d’erreur et Q est le bruit du système.

- L'étape d'association entre les bounding boxes estimées et actuelles est effectuée à l'aide de la méthode Hongroise [OpenCV, Background subtraction using MOG2, Online; accessed May 24, 2017] qui consiste en une maximisation du poids de l'ensemble des couples des bouding boxes estimées et actuelles par une sélection gloutonne des meilleurs couples de bounding boxes. Le poids pour un couple de bounding boxes étant défini comme, leur intersection divisée par leur union (métrique couramment appelée Intersection Over Union (IOU), voir plus loin). Enfin, les couples résultants de cette association possédant une IOU inférieure à une valeur seuil sont rejetés car considérés comme non-concluants.

- Vient enfin l'étape de correction des bounding boxes. Chaque couple de bounding boxes représente théoriquement une cible identique. Leurs vecteurs d'état peuvent alors être moyennés pour obtenir une meilleure estimation de la cible. Les bounding boxes corrigées ainsi calculées sont alors concaténées aux trajectoires de leurs cibles respectives. L'étape de mise à jour est la suivante :

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \quad (4.5)$$

$$\hat{x}_k = \hat{x}_k^- + K_k (z_k - H\hat{x}_k^-) \quad (4.6)$$

$$P_k = (I - K_k H) P_k^- \quad (4.7)$$

Où K est le gain de Kalman, H est la matrice de mesure, R est la matrice de covariance du bruit- de mesure et z est la valeur de mesure.

Plus de détails sur les paramètres du filtre de Kalman utilisé dans les étapes de SORT sont disponibles dans l'annexe [C].

Au cours de ce cycle, plusieurs situations particulières peuvent apparaître :

- Dans le cas où une bounding box estimée ne possède pas d'équivalent parmi les bounding boxes actuelles, la cible est considérée comme perdue et la bouding box estimée devient la bouding box corrigée.

- Dans le cas où une bounding box actuelle ne possède pas d'équivalent parmi les bounding boxes estimées, la bounding box actuelle constitue le début de la trajectoire d'une nouvelle cible. La bouding box actuelle devient alors la bounding box corrigée et est ajoutée à une trajectoire vierge.

Deux paramètres sont introduits pour gérer ces cas particuliers :

- L'âge maximum d'une trajectoire perdue (MAXAGE) qui correspond à la durée maximale de prédiction d'une trajectoire lorsque celle-ci est perdue avant de la considérer comme définitivement terminée.
- L'âge minimal d'une trajectoire (MINHITS) qui correspond au nombre minimal de bounding boxes consécutives nécessaire avant de considérer une trajectoire comme commencée.

La figure (4.4) montre à quelle étape le filtre de Kalman est appliqué dans un algorithme de suivi d'objets multiples et la figure 4.5 montre comment le filtre de Kalman calcule la position future d'un objet détecté.

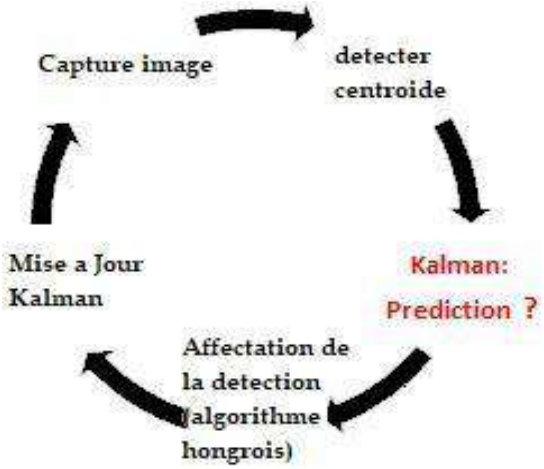


Figure (4.4) : Étapes d'un algorithme de détection multi-objets

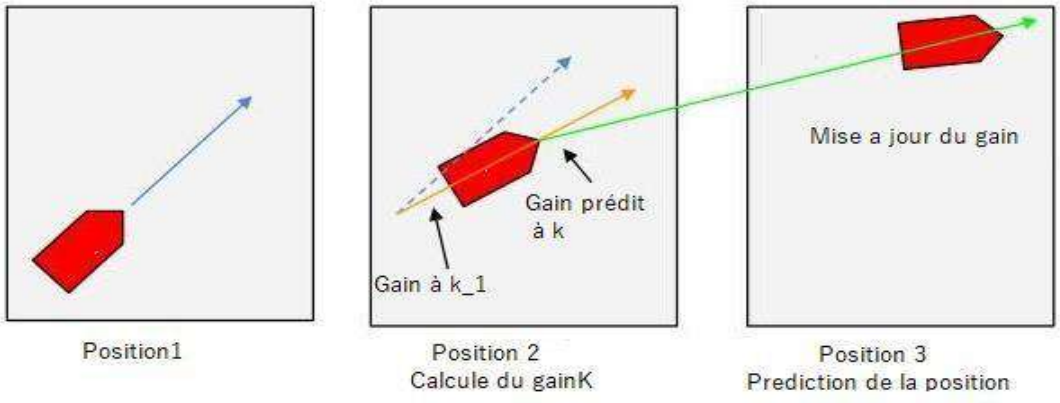


Figure (4.5) : Calcul de la position future d'une prédiction

4.4.4. Création et suppression des identités de piste

Au cours de ce cycle, plusieurs cas particuliers peuvent survenir : dans le cas où un cadre de sélection estimée n'a pas d'équivalent parmi les cadres englobants actuels, la cible est considérée comme perdue et le cadre de sélection estimé devient le cadre englobant mis à jour. En outre, dans le cas où un cadre de sélection actuel n'a pas d'équivalent parmi les cadres de sélection estimés, le cadre de sélection actuel constitue le point de départ de la trajectoire d'une nouvelle cible. Le cadre englobant actuel devient alors le cadre englobant mis à jour et est ajouté à une nouvelle piste.

4.5. Ensemble de données

Pour démontrer la généralité de notre algorithme de suivi, ce dernier est validé sur une variété de séquences vidéo publiques pour l'évaluation, nous avons utilisé sept ensembles de données accessibles au public résumés dans le tableau 4.3. Ces ensembles de données sont sélectionnés de manière à englober des caractéristiques cibles, des contextes environnementaux et des configurations de capteurs variables. Ils comprennent : une caméra fixe et une autre mobile, différentes résolutions de cadre d'image, des réglages intérieurs/extérieurs, des arrière-plans encombrés et dépouillés, des occlusions de cibles répétées et plusieurs interactions de cibles, voir les figures (4.6)-(4.12) et le tableau 4.3.

Tableau 4.3 : Ensembles des données utilisées

Ensemble de données	Camera	Résolution	Fps	#Frames	#Ids
AVIAR-EnterExit [CAVIAR-Project, 2004]	statique	384 × 288	25	383	4
CAVIAR-OneShop [CAVIAR-Project, 2004]	statique	384 × 288	25	1377	7
PETS-S2L1 [J. Ferryman et al., 2009]	statique	768 × 576	7	795	20
TUD-Crossing [M. Andriluka et al., 2010]	statique	640 × 480	25	200	13
ETH-Bahnhof [A. Ess et al., 2009]	mobile	640 × 480	14	1000	222
ETH-Jelmoli [A. Ess et al., 2009]	mobile	640 × 480	14	440	75
ETH-Sunnyday [A. Ess et al., 2009]	mobile	640 × 480	14	354	31

L'ensemble de données PETS-S2L1, figure (4.10), présente une scène extérieure capturée à l'aide d'une caméra de surveillance avec une vue en perspective inclinée, il y a plusieurs occlusions et interactions inter-cibles. L'ensemble de données CAVIAR-OneShop, figure (4.7), présente des occlusions de cibles intermittentes dans un scénario intérieur et les

vitesses des cibles varient, certaines d'entre elles restant statiques pendant un certain temps. De même, CAVIAR-EnterExit propose le même environnement que CAVIAR-OneShop avec des directions de mouvements plus diverses et plusieurs encombrements en arrière-plan. TUD-Crossing, figure (4.12), propose des piétons traversant la route à partir d'une vue latérale (caméra statique) avec cible horizontale bidirectionnelle avec mouvements dans une foule dense. Il est considéré comme l'ensemble de donné le plus sévère dans les cas d'occultation. L'ETH-Bahnhof, l'ETH-Jelmoli et l'ETHSunnyday, figures (4.8), (4.11) et (4.9), ces ensembles de données sont tous acquis à l'aide d'une caméra mobile. Dans ETH-Bahnhof la caméra est montée à hauteur de hanche et la plupart les cibles s'approchent ou s'éloignent de la caméra sur un passage pour piétons. A ETH-Jelmoli, la caméra montre un mouvement erratique au milieu une foule de personnes se déplaçant dans des directions différentes sur une place, la scène devient très complexe bien que la foule reste clairsemée. Dans ETH-Sunnyday, la caméra avance sur un passage pour piétons dans une foule dense. Les cibles se rapprochent et s'éloignent de la caméra.

La figure (4.6) montre des exemples de trames aléatoires prises de chaque ensemble de données.

(a) : ETH-Jelmoli **(b) :** ETH-Sunnday **(c) :** ETH-Bahnhof **(d) :** TUD-Crossing

(e) : CAVIAREnterExit **(f) :** CAVIAROneShop **(g) :** PETS-S2L1

Figure (4.6) : Images prises à partir des sept ensembles de données utilisées.

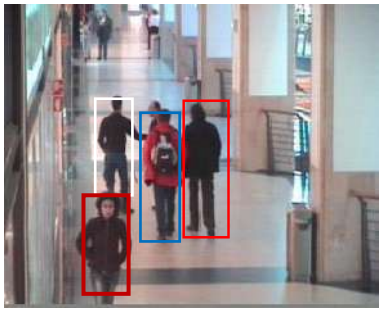
- Quelques challenges de la base de données :

Les séquences vidéo posent des problèmes particulièrement difficiles, parmi ces difficultés soulevées dans les bases de données utilisées est que le nombre des objets cibles est grand, ceci augmente le nombre des fausses assignations lors de l'opération d'association des données. Par ailleurs, le point de vue de la caméra est loin par rapport à la scène ce qui se reflète sur la taille des objets en mouvements (changement d'échelle). Un autre défi pour certaines séquences vidéo est que, la plupart des objets cibles ont la même taille, des vêtements similaires et ont des mouvements similaires (même direction et même vitesse) et parallèles. Même si le nombre des objets cibles est limité, il y a plusieurs occultations multiples (occultations entre plusieurs objets en même temps) et totales (un objet cible est totalement caché par d'autres objets). Aussi les occultations entre les objets sont longues figure (4.12).

Frame 149

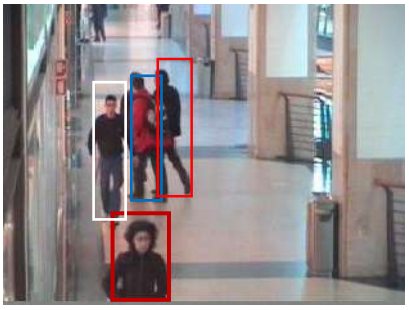
Frame 191

Frame 223



Frame 271

Frame 295



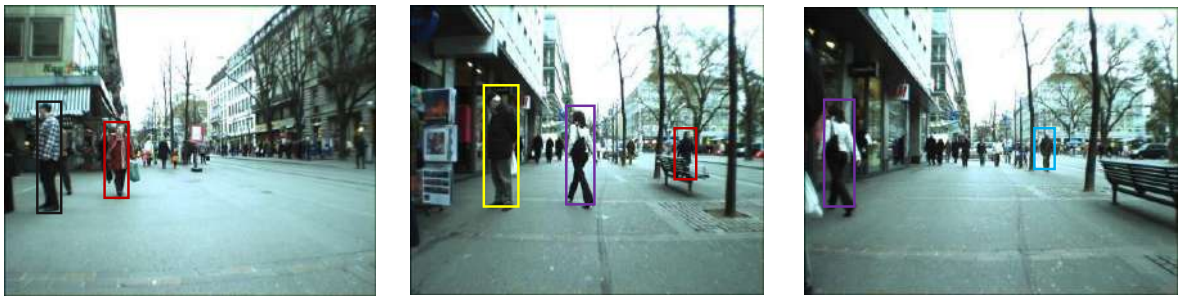
Frame 334

Figure (4.7) : Résultats pour CAVIAR-EnterExit [CAVIAR-Project, 2004].

Nous pouvons résumer les différents problèmes rencontrés lors de l'utilisation des séquences vidéo choisis par les détections manquantes des objets c'est-à-dire la qualité des détecteurs d'objets présente un défi pour le suivi multi objets, figures (4.7) et (4.11). Les occultations, suivre plusieurs objets cibles augmente les cas d'occultation entre les objets, figures (4.7), (4.10) et (4.12) montrent des cas d'occultation multiple (plusieurs objets entrent en occultations en même temps). Variations d'échelle, la distance entre les objets cibles et la position de la caméra varie (à cause des mouvements des objets cibles), ce qui entraîne un

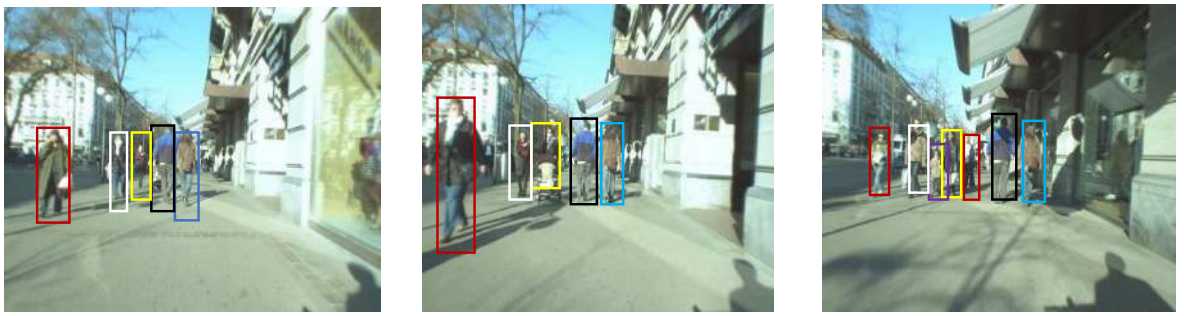
changement d'échelle (la taille d'un objet cible change en s'éloignant ou en s'approchant de la caméra). La dynamique des objets à cause du mouvement arbitraire des objets cibles (mouvements dans toutes les directions), figures (4.7) et (4.10).

Frame 01 Frame 23 Frame 89

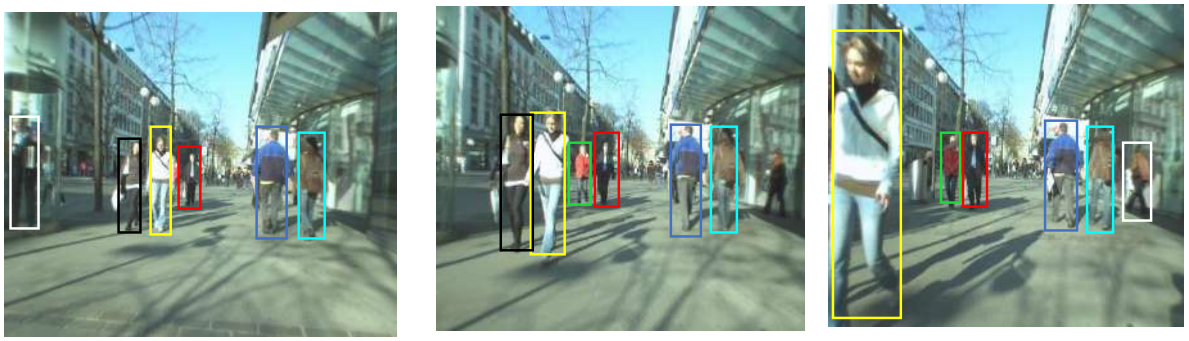


Frame 333 Frame 852 Frame 877

Figure (4.8) : ETH-Bahnhof [A. Ess et al., 2009]

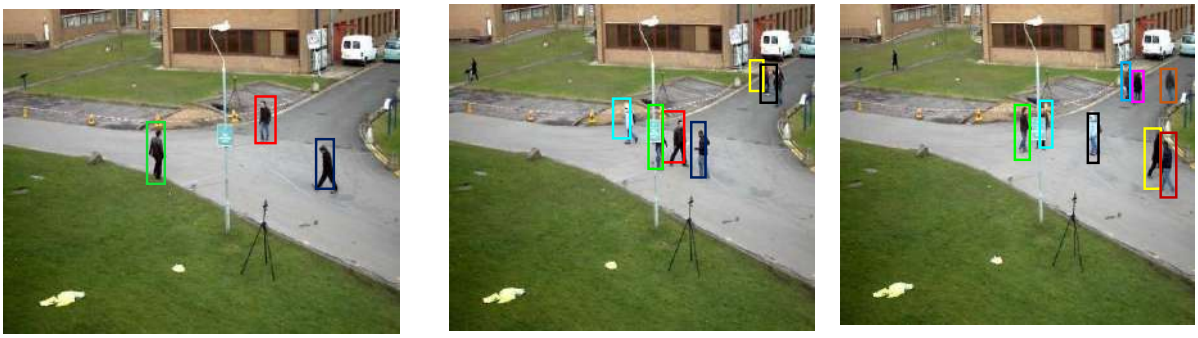


Frame 01 Frame 26 Frame 112

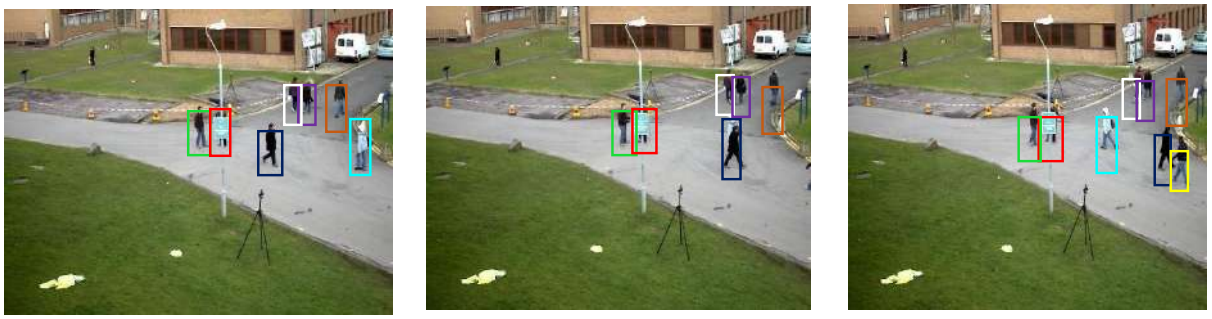


Frame 330 Frame 342 Frame 352

Figure (4. 9) : ETH-Sunnyday [A. Ess et al., 2009]



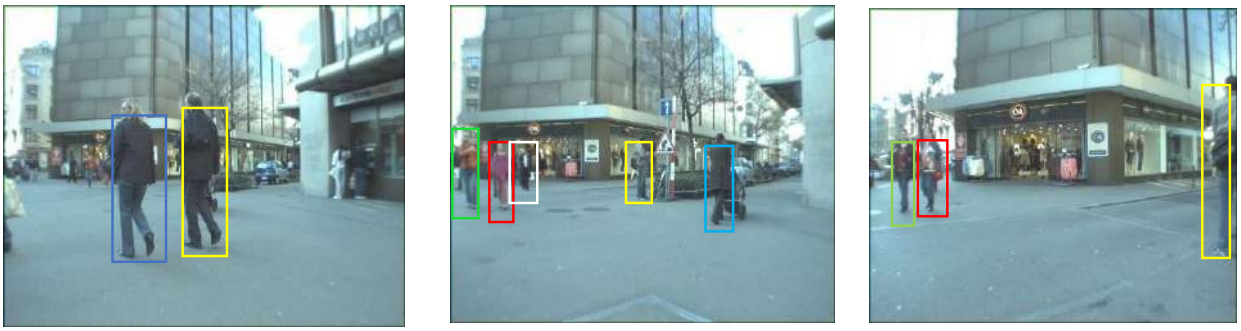
Frame 06 Frame 57 Frame 88



Frame 121 Frame 131 Frame 139

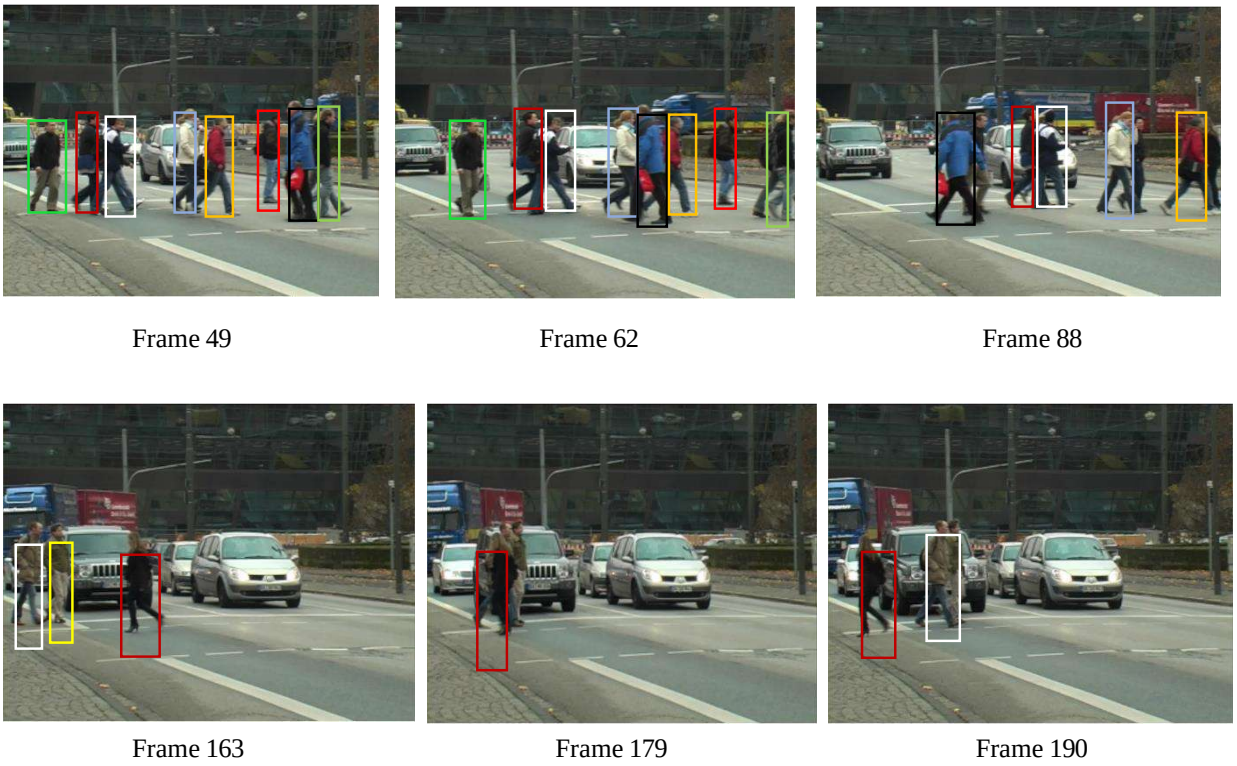
Figure (4.10) : PETS2009-S2L1

Frame 89 Frame 121 Frame 128



Frame 273 Frame 321 Frame 393

Figure (4.11) : RETH-Jelmoli [A. Ess et al., 2009]



Frame 49 Frame 62 Frame 88

Frame 163 Frame 179 Frame 190

Figure (4.12) : TUD-Crossing [M. Andriluka et al., 2010]

4.6. Paramètres d'évaluation

Les performances du détecteur sont mesurées en termes de précision et de rappel qui sont définis selon l'équation (4.8). La précision caractérise la proportion de détections qui sont effectivement de vraies cibles, tandis que le rappel indique la proportion des cibles correctement détectées. TP signifie vrais positifs, FP pour faux positifs et FN pour faux négatifs.

$$Precision = \frac{TP}{TP+FP} \quad ; \quad Recall = \frac{TP}{TP+FN} \tag{4.8}$$

Les performances du tracker sont quantifiées à l'aide des métriques CLEAR-MOT courantes [K. Bernardin et al., 2008]. Les métriques CLEAR-MOT sont principalement basées sur le calcul de deux grandeurs : la précision de suivi multi-objets (MOTA) et la précision de suivi multi-objets (MOTP), équations (4.9) et (4.10).

$$MOTA = 1 - \frac{\sum_t (fn_t + fp_t + ids_t)}{\sum_t g_t} \tag{4.9}$$

$$MOTP = \frac{\sum_{j,t} s(x_j, g_t)}{\sum_t c_t} \tag{4.10}$$

Où, $F_p = \sum_t \frac{FP_t}{g_t}$ dénote le nombre total de faux positifs, $F_N = \sum_t \frac{FN_t}{g_t}$ dénote le nombre total de faux négatifs et $Id_{sw} = \sum_t \frac{Id_{sw,t}}{g_t}$ dénote le nombre total de commutateurs $Id_{sw,t}$, divisé

par le nombre total de cibles de vérité de terrain g_t additionnées sur l'ensemble du jeu de données. Même s'il n'est pas directement calculé, le taux de suivi réel (vrai positif) peut être exprimé comme $T_p = \sum_t \frac{FP_i}{g_t}$. MOTP est le chevauchement de boîte de connexion moyenne (intersection sur union) entre la position cible estimée et les annotations de vérité de terrain sur les cibles correctement suivies. La position $R(x_j)$ rectangulaire estimée du tracker est considérée comme une piste correcte si son score de zone de chevauchement $S(x_j, g_t) = \frac{R(x_j) \cap R(g_t)}{R(x_j) \cup R(g_t)}$ avec la vérité du terrain $g_t$est au-dessus d'un seuil S_0.

4.7. Conclusion

Dans ce chapitre nous avons essayé d'expliquer et en détail les différents outils utilisés tels que les techniques et les algorithmes pour la conception de la chaine de traitement afin de concevoir un système de suivi par détection des personnes dans des multiples scènes dont le but est de démontrer l'impact du choix du détecteur dans la performance de la qualité du suivi. Les résultats ainsi que les discussions dans le chapitre suivant vont confirmer ce qu'nous avons évoqué précédemment.

Chapitre 5
Resultats et Discussion

CHAPITRE 5

RÉSULTATS ET DISCUSSION

5.1. Introduction

Ce chapitre présentera en détail les différentes expériences qui ont été menées, l'objectif est d'évaluer différentes approches de détection-suivi sur des jeux de données généraux afin de recueillir des informations utiles sur la sélection des détecteurs-suivi et plus précisément sur le comportement de chaque approche dans différents contextes d'application. Nous évaluerons d'abord les performances des détecteurs dans chaque ensemble de données, puis effectuerons une évaluation de suivi par détection.

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

- Estimer la position de chaque objet cible d'une façon précise ;
- Maintenir la même identité pour un objet cible au cours du temps.

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

Nous commençons par une présentation de l'implémentation spécifique utilisée ensuite les métriques d'évaluation suivi des ensembles de données utilisés, nous continuerons par une description des paramètres expérimentaux et de mises en œuvre spécifiques ainsi que les résultats obtenus et nous terminerons avec une discussion détaillée sur les résultats obtenus ainsi que le système mis en place.

5.2. Détails de la mise en œuvre des algorithmes

Plusieurs choix d'implémentation et configurations expérimentales sont discutés ci-dessous. Pour ajuster les différents paramètres libres liés aux détecteurs et au tracker, par exemple : les seuils des détecteurs et les paramètres du tracker utilisé, un ensemble de données

de réglage est utilisé. Pour chaque jeu de données d'évaluation, un jeu de données distinct acquis le même paramètre est utilisé pour le réglage. Pour les deux jeux de données CAVIAR, le CAVIAR-Shop [CAVIAR-Project, 2004] est utilisé pour le réglage, par contre le jeu de données PETS-S1L1 [J. Ferryman et al., 2009] correspondant est employé pour PETS-S2L1. Les paramètres utilisés pour les trois ensembles de données de l'ETH sont réglés sur la base de l'ensemble de données ETH-Crossing [A. Ess et al., 2009]

Les paramètres de SORT sont réglés, conformément à l'approche décrite dans leurs publications originales à l'aide des ensembles de données de réglage correspondants [A. Bewley et al., 2016].

a. Détecteurs :

Les détecteurs utilisés dans les expériences sont basés sur des implémentations open source accessibles au public : LDCF et ACF, basés sur la boîte à outils Matlab de Dollar [P. Dollár, 2014] ; DPM basé sur l'implémentation Matlab publiée par [R. B. Girshick et al., 2010], et le détecteur HOG-SVM basé sur OpenCV [S2].

b. Trackeur :

Pour le tracker SORT [A. Bewley et al., 2016], les implémentations Python et Matlab sont utilisées.

c. Temps de calcul:

L'objectif principal de ce travail est d'évaluer systématiquement l'exactitude (MOTA) et la précision (MOTP) des performances de suivi par détection pour mettre en évidence les compromis de choix de détecteur et de tracker. Selon le choix du détecteur et du tracker, la fréquence d'images de l'algorithme complet de suivi par détection varie. Il convient de noter que les détecteurs et le tracker sélectionnés pour l'évaluation sont implémentés dans des différents langages, par exemple Python, Matlab, processeurs hétérogènes, CPU et GPU.

5.3. Ensembles de données

Afin de démontrer la généralité de l'algorithme de suivi, il est validé sur une variété de séquences vidéo publiques pour l'évaluation, nous avons utilisé sept ensembles de données accessibles au public résumés dans le tableau 5.1. Ces ensembles de données sont sélectionnés de manière à englober les caractéristiques cibles, contextes environnementaux et configurations de capteurs. Ils incluent : une caméra fixe/mobile, différentes résolutions de cadre d'image, réglages intérieurs/extérieurs, des arrière-plans encombrés et dépouillés, des occlusions de cibles répétées et plusieurs interactions de cibles.

Comme montre la figure (5.1) et le tableau 5.1, PETS-S2L1 présente une scène extérieure capturée à l'aide d'une caméra de surveillance avec une vue en perspective inclinée ; il a plusieurs occlusions et interactions inter-cibles. L'ensemble de données CAVIAR-OneShop montre des occlusions de cibles intermittentes dans un scénario intérieur et les vitesses des cibles varient, certaines d'entre elles restent statiques pendant un certain temps. De même, CAVIAR-EnterExit propose le même environnement que CAVIAR-OneShop avec des directions de mouvements plus diverses et plusieurs encombrements en arrière-plan. Le TUD-Crossing se caractérise par des piétons traversant la route en vue latérale, (caméra statique), avec des mouvements cibles horizontaux bidirectionnels dans une foule dense. Il est considéré comme l'ensemble de donné le plus sévère (difficile) dans les cas d'occultation. Les ensembles de données ETH-Bannhof, ETH-Jelmoli et ETHSunnyday sont acquis à l'aide d'une caméra mobile. Dans ETH-Bahnhof, la caméra est montée à hauteur de hanche et la plupart les cibles s'approchent ou s'éloignent de la caméra sur un passage pour piétons. A ETH-Jelmoli, la caméra montre un mouvement erratique dans une foule de personnes se déplaçant dans des directions différentes sur une place. La scène devient très compliquée bien si la foule reste clairsemé. Dans ETH-Sunnyday, la caméra se déplace tout en long d'un passage pour piétons dans une foule dense. Les cibles se rapprochent et s'éloignent de la caméra. La figure (5.1) illustre des exemples de trames aléatoires prises de chaque ensemble de données.

(a) ETH-Jelmoli

(b) ETH-Sunnyday

(c) ETH-Bahnhof

(d) TUD-Crossing

(e) CAVIAREnterExit

(f) CAVIAROneShop

(g) PETS-S2L1

Figure (5.1) : Images prises à partir des sept ensembles de données utilisées.

Tableau 5.1 : Ensembles de données utilisées

Ensemble de données	Camera	Résolution	Fps	#Frames	#Ids	Variation Eclairage	Occlusion	Interaction
CAVIAR-EnterExit [CAVIAR-Project, 2004]	Statique Intérieur	384 × 288	25	383	4	++	+	+
CAVIAR-OneShop [CAVIAR-Project, 2004]	Statique Intérieur	384 × 288	25	1377	7	++	++	++
PETS-S2L1 [J. Ferryman et al., 2009]	Statique Extérieur	768 × 576	7	795	20	+	++++	++++
TUD-Crossing [M. Andriluka et al., 2010]	Statique Extérieur	640 × 480	25	200	13	+	+++++	+
ETH-Bahnhof [A. Ess et al., 2009]	Mobile Extérieur	640 × 480	14	1000	222	++	+++	+++
ETH-Jelmol [A. Ess et al., 2009]	Mobile Extérieur	640 × 480	14	440	75	++	+++	+
ETH-Sunnyday [A. Ess et al., 2009]	Mobile Extérieur	640 × 480	14	354	31	++++	+++	++

5.4. Résultats de simulation

Chaque combinaison détecteur-tracker est évaluée sur les sept ensembles de données publics décrits à l'aide des paramètres décrits dans la section 4.6. Les résultats de l'évaluation sont présentés en fonction des paramètres MOTA et MOTP. Il est important de mentionner qu'un seuil de chevauchement, $S_0 = 0{,}5$, est utilisé selon le protocole d'évaluation établi [L.Leal-Taixé et al., 2015], [R. Stiefelhagen et al., 2006].

Plusieurs tableaux récapitulatifs sont présentés qui mettent en évidence certaines caractéristiques, ils sont catégorisés pour faciliter les comparaisons spécifiques. Les catégories tentent de répondre aux questions suivantes :

1. Quel détecteur fonctionne le mieux ?
2. Quelle combinaison de suivi par détection fonctionne le mieux ?
3. Comment fonctionnent les différentes méthodes de suivi par détection sur les ensembles de données de caméras fixes et mobiles ?

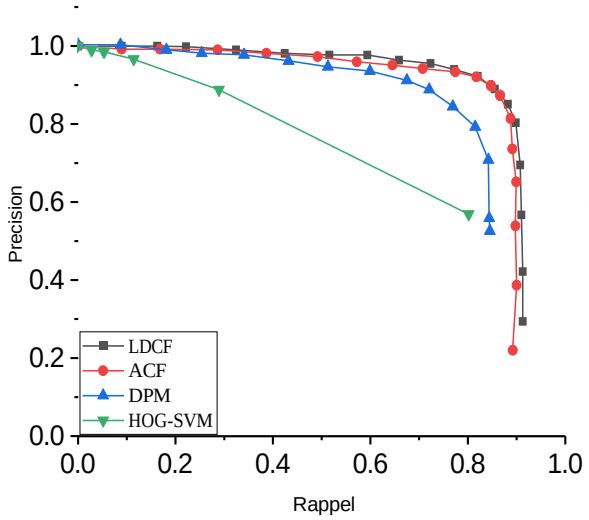
5.4.1. Evaluation des détecteurs

Les quatre détecteurs sont évalués sur tous les ensembles de données présentés en utilisant les métriques rappel et précision. La figure (5.2) montre les courbes de rappel et de précision résultant de la modification du seuil de détection final.

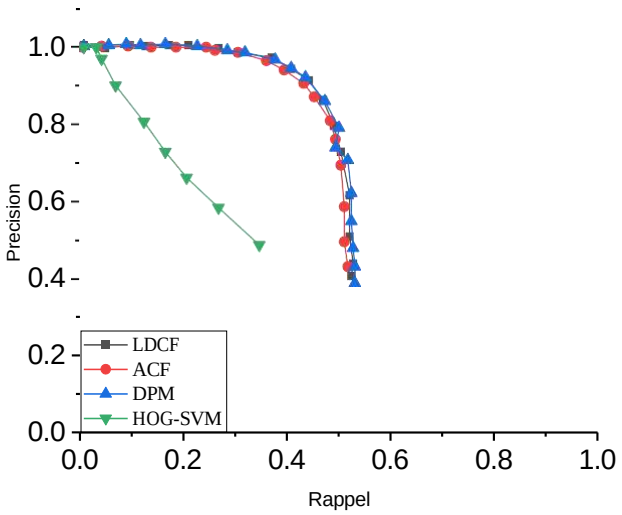
De manière générale, tous les détecteurs, à l'exception du HOG-SVM, fonctionnent bien sur presque tous les ensembles de donnés utilisés.

Le point de fonctionnement de chaque détecteur sur ces ensembles de données est déterminé en réglant globalement la valeur de seuil de sortie du détecteur à un point qui maximise le score $F1_{score} = 2\frac{precision \otimes recall}{precision + recall}$.

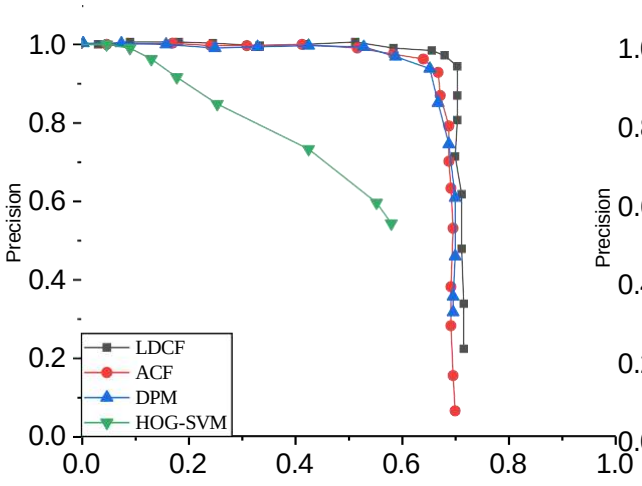
Les rappels et précisions obtenus pour tous les ensembles de données utilisées sont présentés dans le tableau 5.2. Ce tableau permet de mettre en évidence les performances de chaque détecteur sur les différents jeux de données. Plus tard, il servira comme une base pour comparer les différentes performances de chaque détecteur tout seul et en même temps avec un tracker incorporé. Selon la performance des détecteurs sur ces ensembles de données, le tableau 5.2, nous notons ce qui suit : le détecteur LDCF enregistre le rappel et la précision les plus élevés, à l'exception des jeux d'ensembles de Caviar-Oneshop, ETH-Jelmoli et ETH-Sunnyday où le détecteur DPM affiche le rappel le plus élevé, alors que le détecteur HOG-SVM a enregistré le plus mauvais taux de rappel ($rappel = 34\%$ pour la séquence vidéo CAVIAR-OneShop).



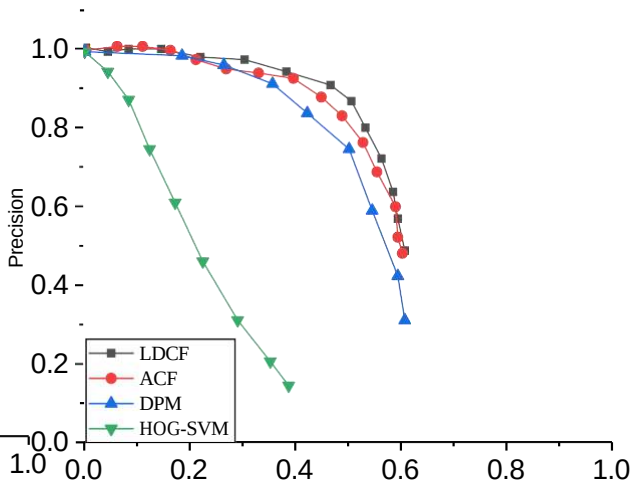
(a): PETS-S2L1



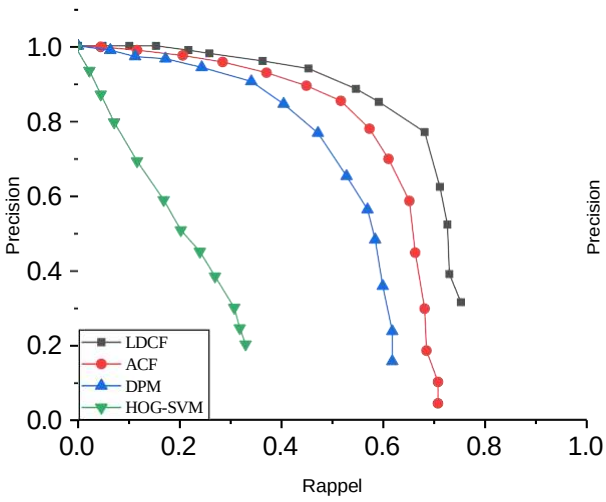
(b): Caviar-One Shop



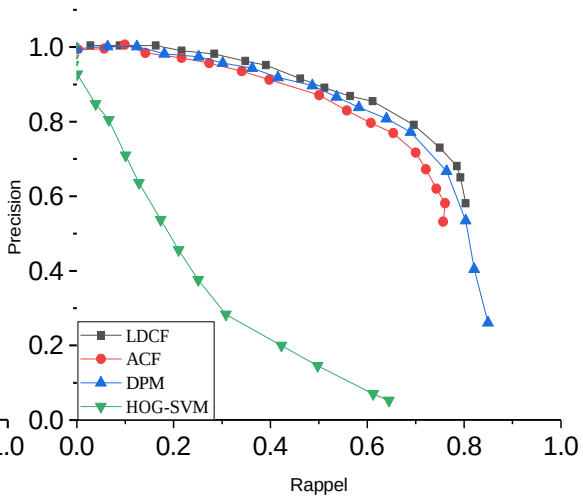
(c) : Caviar Enter-Exit



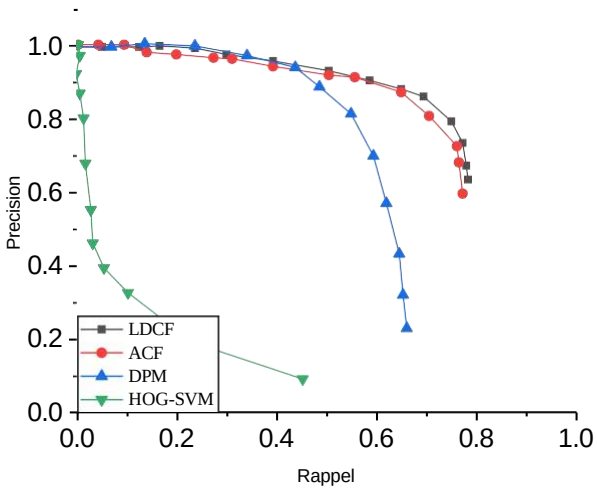
(d) : ETH-Jelmoli



(e) : ETH-Banhoff



(f) : ETH Sunnyday



(g) : TUD-CROSSING

Figure (5.2) : Evaluation des performances du détecteur en termes de métriques de rappel et deprécision

Tableau 5.2 : Ensembles de données utilisés avec les performances de chaque détecteur en termes de métriques de précision et de rappel.

Ensembles de données	**Détecteur** **Precision % / Rappel %**			
	LDCF	**ACF**	**DPM**	**HOG**
CAVIAR-EnterExit [CAVIAR-Project, 2004]	**93/67**	**90/65**	**89/65**	**82/59**
CAVIAR-OneShop [CAVIAR-Project, 2004]	**88/46**	**86/43**	**86/48**	**74/34**
PETS-S2L1 [J. Ferryman et al., 2009]	**93/90**	**90/89**	**92/84**	**87/80**
TUD-Crossing [M. Andriluka et al., 2010]	**90/70**	**89/69**	**85/60**	**63/51**
ETH-Bahnhof [M. Andriluka et al., 2010]	**73/60**	**69/59**	**70/50**	**45/50**
ETH-Jelmoli [M. Andriluka et al., 2010]	**86/48**	**82/46**	**75/50**	**49/39**
ETH-Sunnyday [M. Andriluka et al., 2010]	**88/68**	**87/60**	**80/71**	**59/66**

5.4.2. Évaluation du tracker

Comme mentionné précédemment, les résultats de l'évaluation rapportés sont basés sur les métriques MOTA et MOTP avec un seuil de chevauchement $S_0 = 0{,}5$ selon le protocole d'évaluation établi [L.Leal-Taixé et al., 2015, R. Stiefelhagen et al., 2006].

Pour déterminer la performance globale de chaque approche de suivi par détection (combinaisons détecteur- tracker), les résultats obtenus en exécutant ces combinaisons sur les sept ensembles de données publics sont présentés en détail dans les tableaux 4.3-4.9 pour tous les ensembles de données.

Tableau 5.3 : Résultats CLEAR-MOT sur le PETS-S2L1

	MOTA%	**MOTP%**	**FP%**	**FN%**	**IDws**
LDCF	77.3	71.11	6	8	6
ACF	72.6	72.2	07	13	8
DPM	65.5	72.6	20	17	9
HOG	62.9	72.5	23.6	15	10

Le tableau 5.3 détaille les résultats obtenus sur l'ensemble de données PETS2L1. Dans cet ensemble de données, tous les détecteurs ont fourni des taux de détection et de précision élevés. En conséquence, un MOTA de 77,3% est obtenu par la combinaison LDCF-SORT. Alors qu'un mauvais résultat avec MOTA de 62,9% est obtenu avec la combinaison détecteur-tracker de HOG-SORT.

Tableau 5.4 : Résultats CLEAR-MOT sur le CAVIAR- EnterExit

	MOTA%	**MOTP%**	**FP%**	**FN%**	**IDws**
LDCF	63.6	74.1	20	17	7
ACF	58.3	77.2	23.4	15	17
DPM	58.7	77.3	22	14	16
HOG	54.2	59.2	25	17	18

Pour le jeu de données CAVIAR-EnterExit, il est clair que tous les détecteurs ne fonctionnent pas bien, ce qui est confirmé par les résultats présentés dans le tableau 5.4. Dans ce jeu de données, les meilleurs résultats (précision et exactitude) sont obtenus par le détecteur LDCF, avec un (MOTA) de 63.6% et un (MOTP) de 74.1%. Les performances peuvent atteindre des résultats très moyens avec un MOTA de 54.2% en utilisant le détecteur SVM-HOG.

Tableau 5.5 : Résultats CLEAR-MOT sur le TUD-Crossing

	MOTA%	**MOTP%**	**FP%**	**FN%**	**IDws**
LDCF	**61.6**	**71**	21	18	9
ACF	**41.3**	**73.2**	25	28	18
DPM	36.4	70.	27	26	26
HOG	01.2	50.7	**53**	**30**	**75**

Le tableau 5.5 montre les résultats détaillés de l'ensemble de données TUD-Crossing. A noter que la combinaison SORT-LCDF affiche le meilleur MOTA de 61.6%, par contre, on

constate que la combinaison SORT-HOG rencontre des difficultés avec cet ensemble de données avec un MOTA de 01.2%.

Tableau 5.6 : Résultats CLEAR-MOT sur le ETH-Jelmoli

	MOTA%	MOTP%	FP%	FN%	IDws
LDCF	**40.2**	**75.2**	23	25	16
ACF	**20.2**	**76.4**	21	26	32
DPM	20	73.1	25	**24**	33
HOG	18.1	61.3	**45**	30	**40**

Pour le jeu de données du tableau 5.6 ETH-Jelmoli, il est clair que tous les détecteurs fonctionnent mal, ce qui est confirmé par les résultats présentés dans le tableau 5.6. Dans ce jeu de données, les meilleurs résultats (précision et exactitude) concernent le détecteur LDCF, avec un (MOTA) de 40.2% et un (MOTP) de 75.2%, qui sont obtenus avec le tracker SORT-LDCF. Les performances peuvent être très médiocres avec le détecteur SVM-HO. A noter que la combinaison SORT-LCDF a le meilleur résultat en termes d'exactitude suivie par ACF-SORT avec une différence de 20% pour MOTA, alors que pour MOTP on trouve que ACF-SORT devance SORT-LDCF par une différence de 1.2%. La combinaison HOG-SORT a obtenu un score de 18.1 % pour le MOTA et 61.3 % pour le MOTP.

Tableau 5.7 : Résultats CLEAR-MOT sur le ETH-Banhof

	MOTA%	**MOTP%**	**FP%**	**FN%**	**IDws**
LDCF	**58.1**	**74.7**	13	12	11
ACF	38.7	74	22	20	23
DPM	32.5	73.8	23	**25**	26
HOG	13.1	58.6	**50**	32	**46**

D'après les résultats détaillés de l'ensemble de données ETH-Banhof présentés dans le tableau 4.7, nous remarquons que l'ensemble SORT-LCDF a le meilleur résultat global, en termes d'exactitude et de précision, suivi par l'ensemble ACF-SORT mais avec une différence de 19,4 % pour MOTA et 0,7% pour MOTP. La combinaison HOG-SORT a obtenu un score de 13,1 % pour MOTA et un score de 58,6 % pour MOTP.

Tableau 5.8 : Résultats CLEAR-MOT sur le CAVIAR-OneShop

	MOTA%	**MOTP%**	**FP%**	**FN%**	**IDws**
LDCF	39.1	70.2	20	19	22
ACF	26.9	71.6	25	24	32
DPM	36.1	72.33	25	20	26
HOG	22.3	70.7	22	30	36

Pour la base de données CAVIAR-OneShop, l'ensemble de tous les détecteurs fonctionne mal, comme le montrent les résultats présentés dans le tableau 5.8. Dans cet ensemble de données, les meilleurs résultats (précision et rappel) concernent le détecteur LDCF, voir tableau 5.2.

Les meilleurs résultats (MOTA) de 39,1 % et (MOTP) de 72,33 % sont obtenus en utilisant respectivement les traceurs SORT-LDCF et SORT-DPM. Les performances les plus faibles sont obtenues par le détecteur SVM-HOG avec un MOTA de 22,3%.

Tableau 5.9 : Résultats CLEAR-MOT sur le ETH-Sunnyday

	MOTA%	**MOTP%**	**FP%**	**FN%**	**IDws**
LDCF	**64.6**	**77.6**	16	17	7
ACF	**20.3**	**77**	22	26	39
DPM	54.1	76	20	**19**	13
HOG	18.2	64.3	**29**	36	**41**

Le tableau 5.9 montre les résultats détaillés de l'ensemble de données ETH-Sunnyday. Nous constatons que la combinaison SORT-LCDF a le meilleur résultat global, à la fois en termes d'exactitude et de précision, suivie par DPM-SORT mais avec une différence de 10,5% pour MOTA et ACF-SORT avec 0,6% pour MOTP. La combinaison HOG-SORT a obtenu un score de 18,2 % pour MOTA et 64,3 % pour MOTP.

5.5. Performances sur les ensembles de données de caméra statique et mobile

Le tableau 5.10 présente les résultats moyens de MOTA et MOTP sur l'ensemble de données basés sur une caméra statique (fixe) et une caméra mobile. La meilleure précision de suivi sur les ensembles de données de caméras statiques et mobiles est obtenue par SORT-

LDCF. En moyenne, la meilleure précision de suivi de caméra statique est de 6.1%, est supérieure à la précision de la caméra mobile. La meilleur précision de suivi sur les ensembles de données statiques est obtenue à la fois par SORT combiné avec le détecteur LDCF. Même sur les ensembles de données mobiles, SORT-LDCF a réalisé le meilleur résultat. Sur les ensembles de données statiques la marge de précision de suivi du meilleur détecteur est plus élevée de 25.25% que la précision de suivi du mauvais détecteur. De même pour les ensembles de données mobiles la marge est très importante à 37.84% entre le meilleur et le mauvais détecteur.

D'après les résultats moyens illustrés dans le tableau 5.10, il existe une différence de 15.46% entre la précision du suivi MOTA de la base de données (caméra statique) et la base de données (caméra mobile).

Tableau 5.10 : Comparaison des performances de suivi par détection basés sur des caméras mobiles et statiques

Type de caméra / **Détecteurs**		**Caméra statique**	**Caméra mobile**
LDCF	**MOTA%**	60.4	54.3
	MOTP%	71.6	75.88
ACF	**MOTA%**	49.77	26.4
	MOTP%	73.55	75.8
DPM	**MOTA%**	49.17	35.5
	MOTP%	73.05	35.5
HOG	**MOTA%**	35.15	74.3
	MOTP%	63.17	16.46
MOYENNE	**MOTA%**	48.62	33.16
	MOTP%	70.34	71.84

5.6. Discussion

Nos discussions sont basées sur les résultats présentés dans les tableaux précédents qui fournissent des informations très riches sur les performances de différentes combinaisons détecteur-tracker sur différents ensembles de données/contextes. Sans doute, la performance de

chaque suivi par détection est considérablement influencée par le détecteur, compte rendu des performances globales du suivi par détection sur chaque ensemble de données sur les deux jeux de données CAVIAR (EnterExit et OneShop), il existe une différence significative dans la précision de suivi en raison du choix du détecteur et également de l'environnement avec un maximum de six cibles et légère variation d'éclairage due aux ombres des colonnes verticales intérieures. Les performances globales sur CAVIAR-OneShop sont très faibles en raison d'un rappel et d'une précision faibles du détecteur. Dans ces deux ensembles de données, étant donné que l'environnement d'arrière-plan ne change pas, toute occurrence de faux positif est susceptible de se reproduire affectant les performances globales.

D'autre part, la base de données PETS-S2L1 est un ensemble de données plus simple malgré la présence de plusieurs occlusions et interactions cibles. L'encombrement en l'arrière-plan est minime et en raison de la position de la caméra, le déplacement de la cible correspondant sur le plan image est petit.

Le quatrième ensemble de données, TUDCrossing, présente une occlusion inter-cible significative. Le Tracker SORT atteint les meilleures précisions de suivi. Les trois jeux de données ETH – ETH-Bahnhof, ETH-Sunnyday, ETH-Jelmoli ont tous des caractéristiques similaires.

Les jeux de données SORT-LDCF aboutissent à la meilleure précision de suivi. D'après les résultats obtenus, il peut être a fait valoir qu'aucune approche de suivi par détection n'a les meilleures performances dans tous les jeux de données. Selon la nature de l'ensemble de données, un changement de détecteur peut aider à améliorer les résultats du suivi (tracking).

Le choix du détecteur est très important. Même une petite différence dans le taux de rappel des détecteurs peut entraîner une différence significative dans la précision du suivi. Par exemple, en TUD-Crossing, la meilleure précision de suivi (MOTA) est obtenue avec SORT-LDCF (61,6 %) et la deuxième meilleure avec SORT-ACF (41,3 %), une différence de précision de 20,3 % même s'il n'y a que 1 % de différence de rappel et de précision entre les deux détecteurs. Les mauvaises précisions pour le tracker SORT sont principalement dues au nombre élevé de commutateurs d'identification obtenues lors de l'utilisation de HOG-SVM.

HOG-SVM a une précision de suivi moyenne inférieure à celle de tous les autres détecteurs. Les résultats obtenus indiquent que des meilleures précisions de suivi sont obtenues lors de l'utilisation des détecteurs LDCF, ACF et DPM.

Compte tenu des performances des approches de suivi par détection sur les ensembles de données de contextes statiques et mobiles, il est évident que les performances varient en fonction de l'ensemble de données, une meilleure qualité de suivi moyenne est observée lorsqu'il s'agit de caméras fixes plutôt que de caméras mobiles. Ceci est clair car la combinaison de la caméra et du mouvement cible pose plus de défis pour le tracker.

Dans les ensembles de données de caméras statiques et mobiles, la combinaison SORT-LDCF se distingue en moyenne comme le meilleur tracker. Les détecteurs LDCF et ACF offrent les meilleures précisions de suivi sur les ensembles de données de caméras statiques et mobiles, respectivement. Détecteur LDCF combiné avec le détecteur SORT est plus robuste à la diversité et à la variabilité des ensembles de données. Les performances de suivi par détection sur les ensembles de données mobiles sont inférieures à celles des jeux de données statiques. Cela est dû aux faibles performances du détecteur R/P et aux mouvements couplés de la caméra et des cibles.

La sélection du détecteur a un impact plus important sur la précision du suivi sur les ensembles de données mobiles que sur les jeux de données statiques. Comme indiqué, la précision de suivi (MOTA) est une mesure plus pertinente que la précision de suivi (MOTP). Ceci est également démontré dans la littérature [L. Leal-Taixé et al., 2015]. De plus, nos évaluations mettent en évidence le rôle clé du détecteur dans le suivi par détection.

Sur la base des résultats et des discussions ci-dessus, les observations importantes suivantes peuvent être décrites :

- La combinaison LCDF-SORT s'avère être le meilleur trackers (suivi des performances) sur les sept jeux de données. En revanche, le classement des détecteurs est assez clair : LDCF, ACF, DPM et enfin HOG-SVM comme mauvais détecteur utilisé
- Nos résultats expérimentaux démontrent qu'une différence de 1 % dans le rappel des détecteurs peut entraîner une réduction de 20 % du MOTA. Par conséquent, le détecteur joue un rôle clé dans la performance globale et il convient d'y accorder plus d'attention.
- Les détecteurs LDCF, ACF et DPM combinés avec SORT sont plus robustes à la diversité et à la variabilité des ensembles de données traités. Par conséquent, il est préférable d'utiliser ces détecteurs pour une application de suivi par détection, sans aucune information à priori sur la nature de la scène (l'environnement).
- Les performances du suivi par détection sur les jeux de données mobiles sont inférieures à celles des ensembles de données statiques. Cela est dû aux performances inférieures

(faibles performances) du détecteur Rappel/Précision et aux mouvements couplés (aux doubles mouvements) de la caméra et des cibles. Le choix du détecteur a une influence plus élevée sur la précision du suivi sur les ensembles de données mobiles que sur les ensembles de données statiques.

- Dans nos expériences et évaluations, nous nous sommes concentrés sur le rôle du détecteur dans le suivi de la détection, il a été constaté que la précision du suivi (MOTA) est plus pertinente que la précision du suivi (MOTP). Étant donné que les détecteurs sont caractérisés par rappel/précision, ces mesures peuvent également être utilisées comme indicateurs de performance de suivi par détection et, plus important encore, doivent être utilisées lors du prototypage de système de suivi par détection.

5.7. Conclusion

Dans ce travail, nous avons présenté plusieurs évaluations comparatives de suivi par détection en utilisant une combinaison d'un tracker spécifique et de quatre détecteurs sur sept ensembles de données publiques. Notre objectif n'était pas de développer une approche de suivi par détection avec les meilleures performances absolues, mais plutôt d'étudier l'influence des choix de détecteur et de tracker sur les performances globales de suivi.

Les résultats montrent que la performance globale dépend de la difficulté de l'ensemble de données, de la performance du détecteur sur l'ensemble de données spécifique et de la combinaison tracker-détecteur. Un tracker est plus sensible à la sélection du détecteur, une attention particulière doit être accordée au choix du détecteur exact à utiliser dans le suivi par détection et de vérifier ses performances sur un ensemble de validation avant de se connecter au tracker si possible.

Chapitre 6
Conclusion Générale et Perspctives

CHAPITRE 6

CONCLUSION GÉNÉRALE ET PERSPECTIVES

6.1. Conclusion générale

Les travaux menés dans cette thèse portent principalement sur la détection et le suivi d'objets dans un environnement dynamique interne et extérieur. En effet ce travail s'est articulé autour de six chapitres, le premier visait à présenter la problématique du thème et une introduction générale sur la détection et le suivi visuels d'objets, dans le deuxième chapitre nous avons présenté un état d'art sur les travaux liés à la détection et le suivi visuels d'objets ainsi qu'une classification des algorithmes de suivi multi-objets. Dans le troisième chapitre, nous avons exposé les approches élaborées du le suivi des personnes qui impliquent l'association de détections dans différentes images, à savoir la soustraction du fond et le flux optique. L'implémentation des algorithmes de nos approches pour la détection et suivi d'objets ainsi que les différents résultats obtenus avec discussion sont exposés dans le cinquième chapitre.

Le but de ce travail est de considérer uniquement les détecteurs d'objets et les associations de données comme des outils essentiels de la solution algorithmique proposée. Sur cette base, nous avons abordé dans le quatrième chapitre notre méthode de suivi multi-objets validée sur des séquences vidéo publiques et évaluée à l'aide du jeu de paramètres d'évaluation ClearMOTA. L'algorithme de suivi par détection adopté s'appuie sur des détecteurs d'objets pour initialiser, terminer et mettre à jour les trajectoires. Il est important de mentionner ici tout suivi dans ce travail est effectué sur le plan de l'image et la sortie du tracker décrit un rectangle englobant délimitant l'objet. Les détections et les trajectoires sont mises en correspondance par le module d'association de données, qui identifie les détections comme : une des cibles, ou elle est utilisée pour mettre à jour la trajectoire, ou comme une nouvelle cible, ou elle est utilisée pour créer une trajectoire potentielle qui attend d'autres associations avec de futures détections pour devenir une trajectoire. Le tracker ou le filtre lui-même traite alors la façon dont la

trajectoire se propage sur l'image actuelle compte tenu du modèle dynamique et estime la boîte englobante la plus probable de l'objet dans l'image actuelle.

L'étape de suivi consiste à attribuer un identifiant unique à chaque objet précédemment détecté, qui est mis en œuvre par un simple algorithme de suivi en temps réel et en ligne (SORT). SORT fonctionne sur la base d'un filtre de Kalman prédit sur le vecteur d'état de la boîte englobante, en supposant une vitesse constante de l'objet dans le cadre de l'image. Les itérations SORT sont divisées en quatre étapes distinctes, qui sont brièvement décrites ci-dessous :

- Nous considérons un état initial dans lequel toutes les boîtes englobantes retenues appartiennent à la trajectoire d'un seul objet, la première étape de SORT consiste à recevoir la boîte englobante nouvellement détectée par le détecteur sur une nouvelle trame, correspondant à l'étape d'observation du filtre de Kalman prédit.
- Dans l'étape suivante, le carré englobant estimé de chaque objet dans la nouvelle image est prédit à l'aide d'une carte d'Euler, en supposant que la vitesse de l'objet reste constante entre l'image précédente et l'image actuelle, cela correspond à la phase de prédiction du filtre de Kalman.
- Vient ensuite l'étape d'association entre la boite englobante estimée et la boîte englobante actuelle. Cette étape est réalisée à l'aide de la méthode Hongroise, qui consiste à maximiser les poids de toutes les paires de boîtes englobantes estimées et actuelles en sélectionnant avidement la meilleure paire de boîtes englobantes. Le poids de plusieurs boîtes englobantes est défini comme leur intersection divisée par leur union (IOU). Enfin, les couples issues de cette association dans lesquelles l'IOU est inférieure à la valeur seuil sont rejetées car considérées comme indéterminés.
- La dernière étape consiste à corriger la boîte englobante. Chaque paire de boîtes englobantes représente théoriquement un objet identique. Leurs vecteurs d'état peuvent ensuite être moyennés pour obtenir une meilleure estimation de la cible. Les boîtes englobantes corrigées ainsi calculées sont ensuite reliées aux trajectoires de leurs cibles respectives. Si la boîte englobante estimée n'a pas d'équivalent dans la boîte englobante actuelle, la cible est considérée comme manquante et la boîte englobante estimée devient la boîte englobante corrigée. Si la boîte englobante courante n'a pas d'équivalent dans la boîte englobante estimée, alors la boîte englobante actuelle constitue le point de départ de la nouvelle trajectoire de l'objet et elle devient alors la boîte englobante corrigée et est ajoutée à la trajectoire vierge.

Nos principales contributions peuvent être résumées comme suit :

- Évaluer les méthodes de détection et de suivi basées sur des combinaisons corrélées de trckers et de détecteurs dans différents environnements d'application.
- Sur la base des résultats expérimentaux obtenus à l'aide des détecteurs et trackers choisis, nous avons essayés de présenter des idées générales et des discussions qui mettent en évidence l'impact des choix de détecteurs sur les performances du suivi et cela est confirmé par les résultats expérimentaux qui montrent que le suivi est sensible au choix du détecteur et doit être appliqué après une évaluation minutieuse. Même une différence de 1% dans le rappel du détecteur peut entraîner une baisse de 20% de la précision du suivi (MOTA).

6.2. Perspectives

Plusieurs points ressortent de cette thèse. Malgré les résultats encourageants, il reste encore des pistes à explorer.

Dans ce travail, nous avons montré que la qualité de la détection a un impact significatif sur les performances des méthodes de suivi multi-objets par détection. En fait, dans le chapitre résultats et discussion, on voit que les performances de suivi atteignent un pourcentage très intéressant lorsque la meilleure précision de détection est utilisée. Il serait intéressant d'améliorer la qualité de la détection pour réduire le taux de faux positifs. En particulier, nous pouvons appliquer un détecteur d'objet qui fournit des détections plus fiables.

Pour améliorer les performances et la rapidité des détecteurs utilisés dans nos travaux, plusieurs solutions sont possibles : premièrement, changer le modèle du détecteur utilisé par un autre modèle plus rapide. Par exemple, ils peuvent être remplacés par des modèles de détecteurs à une seule étape tels que les variantes de YOLO [J. Redmon et al., 2016] ou SSD [W. Liu et al., 2016].

Le tracker SORT simple en temps réel et en ligne relègue la qualité de la détection au détecteur et uniquement les coordonnées de la boîte englobante pour corréler les annotations à chaque nouvelle image. L'avantage de cette stratégie est que le temps d'inférence du tracker est fortement réduit en contrepartie de forts échanges d'identifiants entre trajectoires, notamment lorsque deux cibles ont des trajectoires proches dans le plan image.

Pour corriger ce problème, il existe deux solutions possibles : d'après [S. Murray, 2017], les résultats du suivi peuvent être améliorés en modifiant les métriques similitudes utilisées par le tracker. La mesure de similarité est la fonction utilisée pour évaluer la similarité de deux

annotations, le résultat de cette fonction est appliquée pour chaque paire d'annotations de la trame précédente et de la trame courante forme la matrice de pondération pour l'application de la méthode Hongroise. La mesure de similarité utilisée dans notre travail est l'IOU entre les boîtes englobantes cibles, mais d'autres mesures sont également possibles, comme le coût linéaire ou le coût exponentiel proposé par [R. Sanchez-Matilla et al., 2016] et [F. Yu et al., 2016]. Selon les résultats présentés dans [S. Murray, 2017], ces métriques améliorent la cohérence du tracker dans le temps, en particulier à faible FPS. L'apparence des objets peut également être utilisée pour améliorer leur reconnaissance, avec cette logique, DeepSORT [N. Wojke et al., 2017] est conçu, il combine un tracker SORT avec des métriques d'association d'annotations. Cette métrique d'association, appelée « cosine softmax loss », est basée sur la l'identification de la cible après occultation complète. Cet indicateur est un ensemble de données réidentifiées pour être le plus discriminant possible entre les cibles.

Annexes

ANNEXE A

A.1. Mesures de performances pour le suivi multi-objets

Pour mieux comprendre les métriques proposées, nous expliquons d'abord la qualité que nous attendons d'un tracker multi-objets idéal. Il doit trouver le nombre correct d'objets à tout moment et estimer la position de chaque objet aussi précisément que possible (noter que les attributs tels que la silhouette, l'orientation ou la vitesse des objets ne sont pas explicitement pris en compte ici). Il doit également maintenir un suivi cohérent de chaque objet dans le temps : il doit attribuer un identifiant de suivi unique à chaque objet qui reste constant tout au long de la séquence (même après une occlusion temporaire, etc.). Cela conduit à des critères de conception pour les métriques de performance suivantes, ils doivent permettre de juger de la précision du tracker dans la détermination des positions précises des objets. Ils doivent refléter sa capacité à suivre de manière cohérente les configurations des objets au fil du temps, c'est-à-dire à suivre correctement les trajectoires des objets, chacun produisant exactement une trajectoire. De plus, nous voulons que les métriques utiles aient le moins de paramètres libres possible, des seuils ajustables, etc., pour aider à simplifier l'évaluation et maintenir des résultats comparables, clair, facile à comprendre et se comporte selon l'intuition humaine, notamment en présence d'erreurs multiples de natures différentes ou en cas de répartition inégale des erreurs tout au long de la séquence assez général pour comparer la plupart des types de trackers. (Trackers 2D, 3D, trackers de centroïdes d'objets ou trackers de régions d'objets, etc.), petits en nombre mais expressifs, ils peuvent donc être utilisés, par exemple, dans de grandes évaluations comparant de nombreux systèmes. Sur la base des critères ci-dessus, nous proposons une procédure pour évaluer systématiquement et objectivement les propriétés des trackers.

En supposant que pour chaque période t, le tracker multi-objets génère un ensemble d'hypothèses $\{h_1, ..., h_m\}$ pour un ensemble d'objets visibles $\{o_1, ..., o_n\}$, le processus d'évaluation consiste en ce qui ne suivre

Pour chaque instant t, Établir la meilleure correspondance possible entre l'hypothèse h_j et l'objet o_i. Pour chaque correspondance trouvée, on calcule l'erreur dans l'estimation de la position de l'objet et on cumule toutes les erreurs de communication :

- Compter tous les objets sans sortie hypothétique comme des ratés ;

- Compter toutes les hypothèses de suivi où aucun objet réel n'existe comme faux positifs, tous les événements où l'hypothèse de suivi d'un objet a changé par rapport à la trame précédente sont comptés comme des erreurs de non-concordance. Cela peut se produire, par exemple, lorsque deux objets ou plus sont échangés alors qu'ils sont proches l'un de l'autre, ou lorsqu'une piste d'objet est réinitialisée avec un ID (identificateur) de piste différent après avoir été précédemment perdue en raison d'une occlusion.

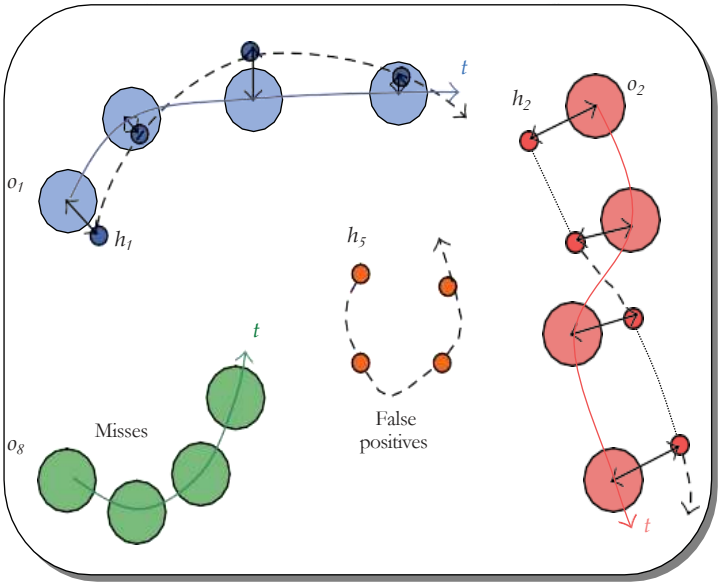


Figure (A.1) : Mappage des hypothèses de suivi aux objets. Dans le cas le plus simple, il suffit de faire correspondre la paire objet-hypothèse la plus proche pour chaque période t [K. Bernardin et al., 2008]

Les performances de suivi peuvent alors être exprimées visuellement en deux nombres : "exactitude du suivi" qui indique à quel point l'emplacement exact de la personne est estimé, "précision du suivi" qui indique la quantité d'erreurs commises par le tracker en termes d'échecs, de faux positifs et d'absence de correspondance, restaurer les échecs de piste, etc. Ces mesures sont détaillées dans la section suivante.

A.2. Etablissement de la correspondance entre les objets et les hypothèses de suivi

Comme mentionné ci-dessus, la première étape de l'évaluation des performances d'un tracker multi-objets consiste à trouver la séquence d'hypothèses d'objets $\{h_1, ..., h_m\}$ que le tracker génère dans chaque image par rapport aux objets réels $\{o_1, \dots, o_m\}$, figure (A.1), correspond naïvement à la paire objet-hypothèse la plus proche et traite tous les objets restants comme des échecs et toutes les hypothèses restantes comme des faux positifs. Cependant, il y a quelques points importants à considérer qui rendent la procédure moins simple.

Premièrement, si la distance $dist_{i,j}$ de l'objet o_i et de l'hypothèse h_h dépasse un certain seuil T, sa correspondance ne doit pas être établie. Il existe une limite conceptuelle au-delà de laquelle nous ne pouvons plus parler d'erreurs d'estimation de position, mais nous devons plutôt affirmer que le tracker a raté l'objet et suit autre chose, ceci est illustré sur la figure (A.2(a)). Pour les trackers de zone d'objet (c'est-à-dire les trackers qui estiment également la taille de l'objet ou la zone qu'ils occupent), la distance peut être exprimée en termes de chevauchement entre l'objet et l'hypothèse.

Deuxièmement, afin de mesurer la capacité du tracker à étiqueter les objets de manière cohérente, il est nécessaire de détecter le moment où les objets sont en conflit au fil du temps. La figure (A.2(b)) illustre ce problème. Ici, une piste a été affectée par erreur à trois objets différents au fil du temps, des incompatibilités peuvent se produire lorsque des objets sont proches les uns des autres et que les trackers échangent par erreur leurs identités. Cela se produit également lorsqu'une piste est perdue et réinitialisée avec une identité différente. Une façon de mesurer ces erreurs pourrait être de déterminer la « meilleure » cartographie (o_i, h_j) pour chaque objet o_i et hypothèse h_j, par exemple sur la base de la correspondance initiale à o_i , ou la plus correspondante (o_i, h_j) souvent faite tout au long de la séquence. Toutes les correspondances qui violent ce mappage sont alors comptées comme des erreurs. Cependant, dans certains cas, cette mesure peut devenir peu intuitive. Comme le montre la figure (A.2(c)), Par exemple, si les identités de l'objet o_i ne sont échangées qu'une seule fois au cours de la séquence de suivi, le laps de temps dans lequel l'échange se produit peut grandement affecter la sortie de valeur d'une telle mesure d'erreur. C'est pourquoi nous adoptons une approche différente : calcul de l'erreur de non-concordance une seule fois dans la période où le mappage objet-hypothétique change, et considérer que la correspondance dans le segment du milieu est correcte. Surtout lorsque le suivi de nombreux objets et que les inadéquations sont fréquentes, cela nous donne une mesure d'erreur plus intuitive et expressive. Pour détecter le moment où une erreur de non-concordance se produit, une liste de mappages objet-hypothétique est construite. Soit $M_t = \{(o_i, h_j)\}$ l'ensemble des applications au temps t, et soit $M_0 = \{\}$. Ensuite, si une nouvelle correspondance est établie entre o_i et h_k au temps $t + 1$, ce qui contredit l'application (o_i, h_j) dans M_t, calculer l'erreur de non-concordance et remplacer (o_i, h_j) par

Mt (o_i, h_k) dans le $M_t + 1$.

La liste des applications M_t ainsi construite peut maintenant aider La meilleure correspondance entre objets et hypothèses est établie à l'instant $t+1$ lorsqu'il y a plusieurs choix valides, a figure (A.2(d)) illustre cette situation.

Lorsqu'il n'est pas clair quelle hypothèse correspond à l'objet o_i, h_0 avec $(o_i, h_0) \in M_t$ est préféré, car il s'agit probablement de l'orbite correcte. D'autres hypothèses sont considérées comme des faux positifs, et peuvent être dues au fait que le tracker émet plusieurs hypothèses pour l'o_i, ou parce que les hypothèses qui suivaient auparavant un autre objet ont accidentellement traversé l'o_i.

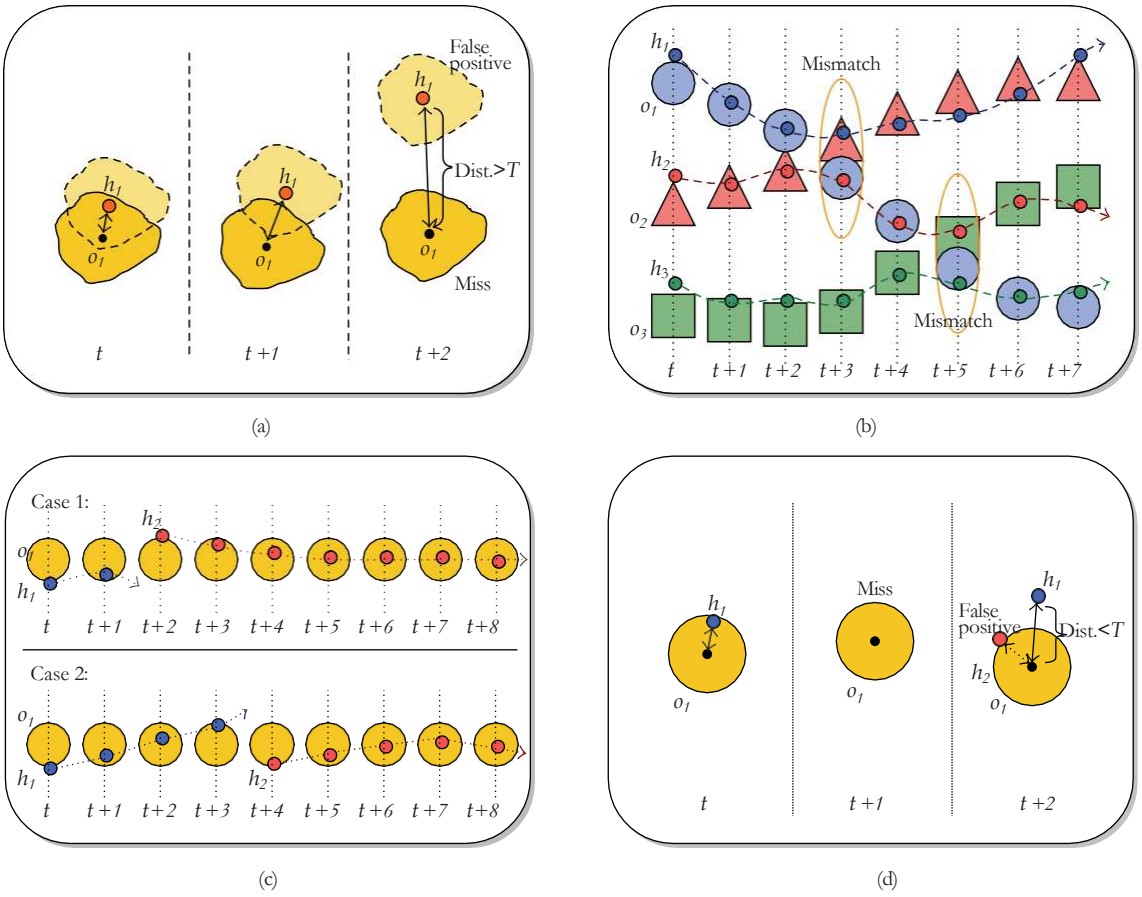


Figure (A.2) : Meilleures métriques de correspondance et d'erreur [K. Bernardin et al., 2008]

Dans La figure (A.2(a)) la distance entre o_1 et h_1 dépasse un certain seuil T, plus aucune correspondance ne peut être établie. Au lieu de cela, o_1 est considéré comme une omission et h_1 comme un faux positif.

La figure (A.2(b)) : Orbitales incompatibles, h_2 est d'abord mappé sur o_2, cependant, après quelques images, o_1 et o_2 se croisent et h_2 suit le mauvais objet, plus tard, il est à nouveau passé à o_3 de manière incorrecte.

Dans La figure (A.2(c)) : Problèmes lors de l'utilisation d'un mappage objet-hypothèse "meilleur" au niveau de la séquence basé sur les correspondances les plus fréquentes. Dans le premier cas, h_1 ne suit que o_1 pendant 2 trames, puis h_2 prend en charge le suivi. Dans le second cas, h_1 suit presque la moitié de la séquence de o_1. Dans les deux cas, le "meilleur" mappage associerait h_2 et o_1, cependant, cela se traduit par le calcul de deux erreurs de non-concordance pour le premier cas ; le deuxième cas comporte quatre erreurs, bien qu'une seule erreur du même type se produise dans les deux cas.

La figure (A.2(d)) : Réinitialisez correctement la piste. A l'instant t, o_1 est suivi par h_1. A $t+1$, la trajectoire est perdue. A $t+2$, il y a deux hypothèses valables. Bien que h_2 soit plus proche de o_1, la correspondance avec h_1 est basée sur la connaissance du mappage précédent au temps $t+1$.

Supposons que l'on fasse correspondre un objet o_i, la priorité est h_0 avec $(o_i, h_0) \in M_t$, puisque c'est probablement la bonne piste. D'autres hypothèses sont considérées comme des faux positifs, et peuvent être dues au fait que le tracker émet plusieurs hypothèses pour l'o_i, ou parce que les hypothèses qui suivaient auparavant un autre objet ont accidentellement traversé l'o_i.

A.3. Indicateurs de performance

Sur la base de la stratégie d'appariement ci-dessus, deux indicateurs très intuitifs peuvent être définis.

- Précision de suivi multi-cibles (MOPT) : Il s'agit de l'erreur totale dans les positions estimées des paires objet-hypothétique appariées sur toutes les trames, moyennée par le nombre total d'appariements effectués. Il montre la capacité du tracker à estimer des emplacements précis d'objets indépendamment des compétences en matière de reconnaissance des configurations d'objets, de maintien de trajectoires cohérentes, etc.
- Précision de suivi multi-cibles (MOTA) : MOTA prend en compte toutes les erreurs de configuration d'objets, les faux positifs, les ratés, les décalages produits par le tracker sur toutes les images. Il est similaire à des métriques largement utilisées dans d'autres domaines (comme le taux d'erreur sur les mots (WER), couramment utilisé dans la reconnaissance vocale), et fournit une mesure très intuitive de la performance d'un tracker dans la détection

d'objets et le maintien de leurs trajectoires, indépendamment de la précision d'estimation de l'emplacement d'objets.

Remarque sur le calcul de la moyenne : notons que pour MOTP et MOTA, il est important de d'abord additionner toutes les erreurs dans la trame avant de calculer la moyenne ou le rapport final. Moyenne $(1/n)$ pour toutes les trames n, par exemple pour les mesures FP et FN dans [K. Smith et al., 2005], peut conduire à des résultats peu intuitifs, ceci est illustré à la figure (A.3). Bien que le tracker manque systématiquement la plupart des objets de la séquence, le calcul pour chaque image, puis le calcul de la moyenne ne donne toujours qu'un taux d'échec de 50 %. D'autre part, agréger d'abord tous les échecs et calculer un seul ratio global donne un résultat plus intuitif, un échec de 80 %.

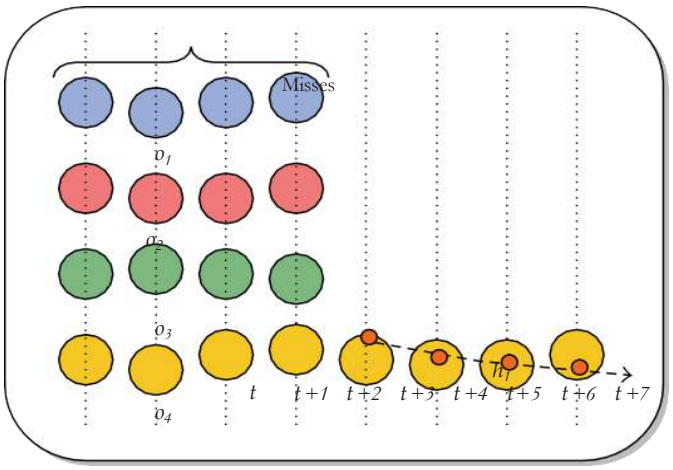


Figure (A.3) : Taux d'erreur calculé. Source [K. Bernardin et al., 2008]

Supposons que la longueur de la séquence soit de 8 images. Pour les trames t_1 à t_4, quatre objets $o_1, \ldots, o_4$ sont visibles, mais aucun n'est suivi. Pour les trames t5 à t8, seul o_4 est encore visible et a été suivi par h_1. Dans chaque trame $t_1 \ldots t_4$, quatre objets sont manqués, ce qui donne un taux d'échec de 100 %. Dans chaque trame $t_5,\ldots,t_8$, le taux d'échec est de 0 %. La moyenne de ces taux d'erreur au niveau de la trame donne un résultat global de (1/8) (4,100 + 4,0) = taux d'échec de 50 % En autres cas, en additionnant d'abord toutes les erreurs, puis en calculant le rapport global, donne un résultat plus intuitif, 16 ratés/20 objets = 80 %.

ANNEXE B

B.1. Détection des piétons

La détection des piétons est un sous-domaine spécialisé de la détection d'objets, actuellement piloté par les industries de l'automobile, de la robotique, de la surveillance et de la sécurité. L'ensemble de données Caltech est une référence populaire pour de nombreux tests de détection, et les réseaux de neurones produisent actuellement les meilleurs résultats. Cependant, il existe de nombreux algorithmes disponibles pour la détection des piétons qui sont moins exigeants en termes de calcul, tels que ceux basés sur le modèle de partie déformable (DPM) et les fonctionnalités de canal agrégées (ACF).

B.1.1. Détecteur basé sur les caractéristiques des canaux agrégés (ACF)

Les caractéristiques de canaux agrégées sont des caractéristiques appropriées pour la détection des piétons. Le cadre de détection dans l'implémentation d'origine est formé sur des patchs d'image de taille 128 x 64 à l'aide de 10 canaux de fonctionnalités. Les caractéristiques sont extraites et 2048 arbres de décision binaires profonds sont entraînés à l'aide d'Adaboost Dans ce travail les détecteurs qui utilisent ces fonctionnalités sont appelés détecteurs ACF.

Les canayx utilisé dans l'implémentation d'origine sont :

- Amplitude du gradient normalisé.
- Histogramme de gradient directionnel à 6 canaux (HOG).
- Image dans l'espace colorimétrique LUV.

Les dix canaux sont sous-échantillonnés par un facteur de quatre puis lissés avant d'alimenter l'arbre de décision.

La détection d'objets est effectuée à l'aide de fenêtres coulissantes à plusieurs échelles. D'autres façons d'accélérer le calcul dans le cadre ACF consistent à calculer uniquement les entités à des échelles différentes et à les approximer à la plupart des échelles. Dans l'implémentation d'origine, le calcul était effectué une fois par octave (une fois par moitié de l'échelle de l'image) et se rapprochait de sept échelles intermédiaires en utilisant la caractéristique calculée la plus proche en utilisant la loi de puissance d'échelle trouvée dans l'article original.

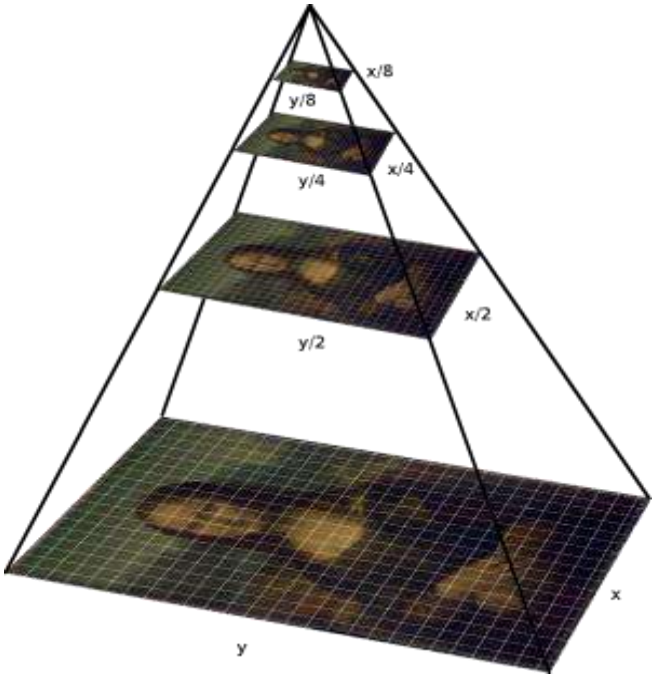


Figure (B.1) : Pyramide à l'échelle : ensemble de versions de la même image à différentes résolutions. [Scalepyramid, Online; accessed June 5, 2017]]

B.1.2. Détecteur basé sur un modèle de parties déformable (DPM)

L'approche du modèle de partie déformable est essentiellement un classificateur qui utilise les mêmes caractéristiques HOG avec la dimension supplémentaire de traiter une personne comme un ensemble de parties distinctes. Ceux-ci sont façonnés de manière à avoir des parties caractéristiques, qui sont généralement la poitrine, les pieds, la tête et les épaules pour les personnes en érection. Chaque partie de l'image est notée en fonction de sa ressemblance avec la partie du corps et de sa position par rapport au corps. Le filtre complet du corps, le filtre racine, est appliqué à un niveau supérieur de la pyramide d'échelle, voir figure (B.1), tandis que les filtres partiels sont appliqués plus à un niveau inférieur de la pyramide, c'est-à-dire avec des résolutions plus élevées. Après avoir calculé individuellement les scores de chaque filtre, les scores du filtre racine peuvent être combinés en un seul score final pour classer les objets, voir figure (B.2). Pour entraîner ce classifieur, une machine à vecteurs de support latents est généralement utilisée.

Cela signifie que ce qui est inclus dans un SVM standard sont des variables latentes inconnues qui sont la position de la partie par rapport au filtre racine. Ces parties ne sont pas apprises en observant leur position dans l'image comme le filtre racine, mais sans aucune annotation de leurs positions. Ce sont des variables latentes. L'emplacement du filtre partiel est

utilisé dans la figure (B.2). Notant la réponse du filtre au filtre racine, cela permet au classificateur de trouver les piétons de manière plus fiable dans des poses différentes de celles du filtre racine.

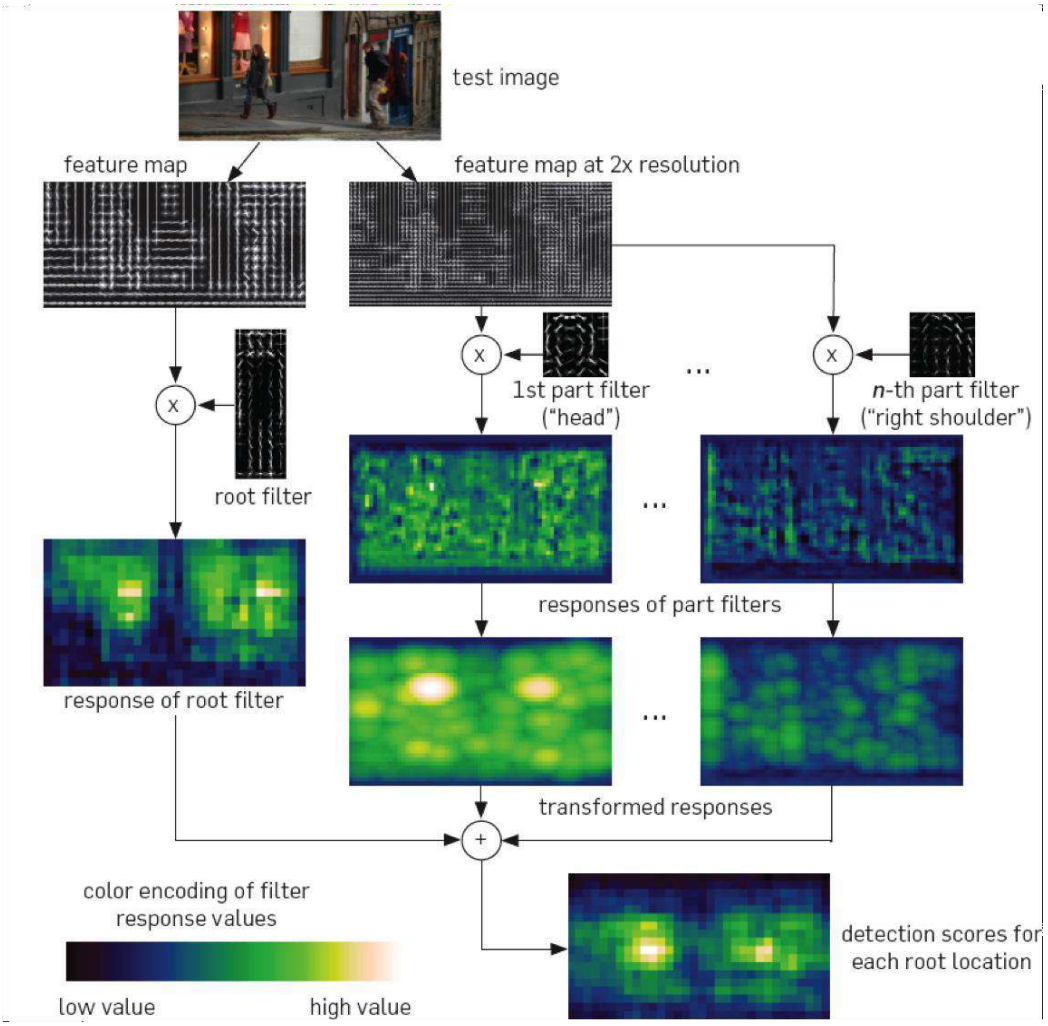


Figure (B.2) : Pipeline de classification DPM où chaque filtre balaie une image différente dans la pyramide d'échelle et s'additionne pour former le score de détection final. La réponse du filtre partiel est également pondérée avec le coût de déformation. [Pedro Felzenszwalb. *Dpmpic*, Online; accessed June 6, 2017]

B.1.3. Histogramme des Gradients Orientés (HOG)

Lors de l'utilisation de différents types d'algorithmes de vision par ordinateur, le plus important est le type de caractéristiques qui peuvent être extraites des images pertinentes. Une caractéristique est une information sur une image ou une séquence de données utilisée pour le calcul. La détection humaine nécessite souvent de nombreuses caractéristiques différentes, car les humains se présentent sous de nombreuses tailles et formes, portent des vêtements différents

et sont visibles sous différentes lumières. Un ensemble populaire de fonctionnalités est l'histogramme des gradients orientés, comme le montrent Dalal et Triggs, qui peut être utilisé efficacement pour la détection des piétons. L'une des raisons pour lesquelles HOG est un bon extracteur de caractéristiques est qu'il utilise des informations sur les bords, c'est-à-dire la zone de contenu à haute fréquence. Ceci est intéressant car c'est souvent le bord de l'image qui fournit les informations nécessaires pour comprendre ce que les gens regardent.

Figure (B.3) : Exemple d'image du descripteur HOG avec image d'entrée [Stefan van der Walt. Hogpicture, Online; accessed May 29, 2017].

ANNEXE C

Précisions sur le Filtre de Kalman Implémenté dans le Tracker SORT

Le but d'un tracker SORT est de suivre une cible dans le temps. L'objectif est résumé dans une boîte englobante décrite par quatre paramètres : centre (u, v), aire s et rapport r. À partir de ces coordonnées et ces dérivées temporelles, le vecteur d'état de la boîte englobante est définit par l'équation (C.1), dont le but est d'évaluer son évolution dans le temps.

$$x = [u, v, s, r, u', v', s']^T \quad \text{(C1)}$$

$$s = w * T \quad \text{(C.2)}$$

$$r = w/h \quad \text{(C.3)}$$

$$\begin{cases} X_{k+1} = F_k X_k + B_k U_k + \alpha_k \\ Z_k = H_k X_k + \beta_k \end{cases} \quad \text{(C.4)}$$

Le filtre de Kalman implémenté dans le tracker SORT est basé sue un système d'équations (C.4), dont la première équation est appelée équation d'évolution, le vecteur d'état futur X_{k+1} peut être obtenu à partir de X_k vecteur d'état courant, U_k vecteur de commande et α_k bruit d'état. La deuxième équation du système, dite équation d'observation, le vecteur d'observation Z_k est obtenu à partir du vecteur d'état courant X_k et du bruit d'observation β_k.

Remarques :

Tout d'abord, nous ne sommes intéressés que par le suivi la boîte englobante, ce qui signifie que nous n'avons aucun contrôle sur les coordonnées de la boîte englobante. Cela signifie $U_k = 0$.

On suppose ici que la vitesse de la cible est constante dans l'espace image entre deux trames consécutives, ceci explique pourquoi le vecteur d'état X_k ne contient que la dérivée première des coordonnées de la boîte englobante. De plus, étant donné la forme de la matrice F_k, on remarque que l'équation (C.1) dans (C.4)) peut être réduite au diagramme d'Euler plus en le bruit d'état α_k.

Les bruits d'état α_k et d'observation β_k sont supposés être des bruits gaussiens avec leurs matrices de covariance Q_k et R_k, respectivement. On remarque qu'une plus grande incertitude en observation est donnée pour l'aire et le rapport des boîtes englobantes. De plus, les dérivées des coordonnées de la boîte englobante ont une plus d'incertitude initiale, comme on peut le voir dans la matrice d'incertitude initiale pour le vecteur d'état P_0.

Les Matrices utilisées dans le filtre de Kalman du traqueur SORT sont :

$$R_k = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 0.1 \end{pmatrix} \quad \text{(C.5)}$$

$$P_0 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 10^{-2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10^{-2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 10^{-3} \end{pmatrix} \quad \text{(C.6)}$$

$$F_k = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{(C.7)}$$

$$Q_k = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 10^{-2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10^{-2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 10^{-4} \end{pmatrix} \quad \text{(C.8}$$

$$B_k = 0 \quad \text{(C.9)}$$

$$H_k = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \quad \text{(C.10)}$$

L'application du filtre de Kalman s'effectue en quatre étapes. Les coordonnées de la boîte englobante de la trame précédente dans la trame courante sont prédites en appliquant la première équation du système d'équations (C.4). Par conséquent, nous avons généré la boîte englobante estimée indiquée par les symboles bleus sur la figure (C.1).

La nouvelle boîte englobante fournie par le détecteur en appliquant la deuxième équation du système d'équations. Ainsi, la boîte englobante observée marquée par le symbole rouge sur la figure (C.1) est générée.

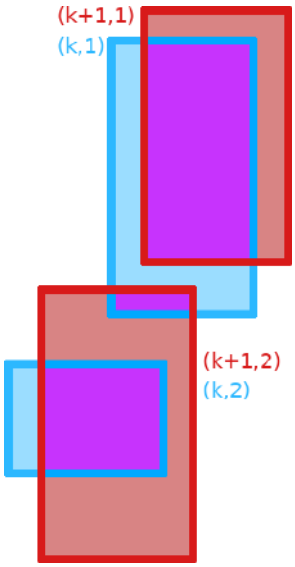


Figure (C.1) : Exemple de deux paires de boîtes englobantes estimées (bleues) et observées (rouges). La boîte englobante est représentée par une paire (X,y), où X est l'indice du cadre à partir duquel la boîte englobante a été obtenue, et y est l'indice de la boîte englobante dans la boîte englobante du cadre avec l'indice X

Corrélation de la boîte englobante estimée à la boîte englobante observée, pour ce faire, nous calculons l'IOU [boîte englobante estimée, boîte englobante observée] pour chaque paire. Ainsi, nous obtenons la matrice de poids représentée sur la figure (C.2). Ensuite, nous appliquons la méthode hongroise à cette matrice de poids, ce qui nous donne les paires de boîtes englobantes $[(k,1),(k+1,1)]$ et $[(k,2),(k+1,2)]$. Enfin, nous calculons un seuil à la valeur IOU. Par exemple, si nous sélectionnons uniquement des annotations avec des IOU supérieures à un seuil de 0,5, seule une paire de $[(k,1),(k+1,1)]$ sera sélectionnée dans l'exemple de la figure (C.2).

	(k+1,1)	(k+1,2)
(k,1)	0,7	0,1
(k,2)	0	0,3

Figure (C.2). Matrice de pondération générée en calculant l'IOU entre chaque paire de boîte englobante estimée, boîte englobante observée

Bibliographie

BIBLIOGRAPHIE

Nguyen, T.L.A., Bremond, F., & Trojanova, J. (2016). *Multi-Object Tracking of Pedestrian Driven by Context. In 2016 13th IEEE International Conference on Advanced Video and Signal based Surveillance (AVSS) (pp. 23-29), IEEE.*

Daniel Castro, Urbano Nunes, and Antonio Ruano. *Feature Extraction for Moving Objects Tracking System in Indoor Environments. Transform, 2004.*

Marta Marron, Miguel Angel Sotelo, Juan Carlos Garcia, David Fernandez, and Ignacio Parra. *3D-Visual Detection of Multiple Objects and Structural Features in Complex and Dynamic Indoor Environments. In IECON 2006 - 32nd Annual Conference on IEEE Industrial Electronics, pages 3373–3378. IEEE, nov 2006.*

Nico Cornelis, Bastian Leibe, Kurt Cornelis, and Luc Van Gool. *3D Urban Scene Modeling Integrating Recognition and Reconstruction. International Journal of Computer Vision, 78 (2-3):121–141, jul 2008.*

Shao-Wen Yang and Chieh-Chih Wang. Simultaneous egomotion estimation, segmentation, and moving object detection. Journal of Field Robotics, 28(4) :565–588, 2011.

Dave Ferguson, Michael Darms, Chris Urmson, and Sascha Kolski. Robotics Institute: Detection, Prediction, and Avoidance of Dynamic Obstacles in Urban Environments. In Intelligent Vehicles Symposium, pages 1149–1154. IEEE, 2008.

Jiajun Zhu, Michael Steven Montemerlo, Christopher Paul Urmson, and Andrew Chatham. Object Detection and Classification for Autonomous Vehicles, 2012.

N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), volume 1, pages 886–893. Ieee, 2005.

Paul Viola and Michael J Jones. Detecting pedestrians using patterns of motion and appearance. In Proceedings Ninth IEEE International Conference on Computer Vision, pages 734–741 vol.2. Ieee, 2003.

R. Benenson, M. Mathias, T. Tuytelaars, and L. Van Gool. Seeking the strongest rigid detector. In Computer Vision and Pattern Recognition (CVPR), 2013.

Rodrigo Benenson, Markus Mathias, Radu Timofte, and L. Van Gool. Pedestrian detection at 100 frames per second. In 2012 IEEE Conference on Computer Vision and Pattern Recognition, pages 2903–2910. IEEE, jun 2012.

Paul Viola and Michael J. Jones. Robust Real-Time Face Detection. International Journal of Computer Vision, 57(2) :137–154, may 2004.

N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), volume 1, pages 886–893. Ieee, 2005.

Paul Viola and Michael J Jones. Detecting pedestrians using patterns of motion and appearance. In Proceedings Ninth IEEE International Conference on Computer Vision, pages 734–741 vol.2. Ieee, 2003.

Christos Tzomakas and Werner von Seelen. Vehicle Detection in Traffic Scenes Using Shadows. Technical report, Institut fur Neuroinformatik, Ruhr-Universitat Bochum, 1998.

Jai Deng, Wei Dong, Richard Socher, Li-Jai Li, Kai Li, and Li Fei-Fei. ImageNet: A large scale hierarchical image database. 2009 IEEE Conference on Computer Vision and Pattern Recognition, pages 248–255, jun 2009.

Xi Li, Weiming Hu, Chunhua Shen, Zhongfei Zhang, Anthony Dick, and Anton Van Den Hengel. A survey of appearance models in visual object tracking. ACM Transactions on Intelligent Systems and Technology, 4(4): Article 58, sep 2013.

Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. Online Object Tracking: A Benchmark. In 2013 IEEE Conference on Computer Vision and Pattern Recognition, pages 2411–2418. IEEE, jun 2013.

Hanxuan Yang, Ling Shao, Feng Zheng, Liang Wang, and Zhan Song. Recent advances and trends in visual tracking: A review. Neurocomputing, 74(18):3823–3831, nov 2011.

Alper Yilmaz, Omar Javed, and Mubarak Shah. Object Tracking: A Suvey. ACM Computing Surveys, 38(4):13–es, dec 2006.

Boris Babenko, Ming-Hsuan Yang, and Serge Belongie. Visual Tracking with Online Multiple Instance Learning. IEEE transactions on pattern analysis and machine intelligence, dec 2010.

Sam Hare, Amir Saffari, and Philip H. S. Torr. Struck: Structured output tracking with kernels. In International Conference on Computer Vision (ICCV), pages 263–270. IEEE, nov 2011.

Zdenek Kalal, Krystian Mikolajczyk, and Jiri Matas. Tracking-Learning-Detection. IEEE transactions on pattern analysis and machine intelligence, 34(7):1409–1422, dec 2012.

Federico Pernici and Alberto Del Bimbo. Object Tracking by Oversampling Local Features. IEEE Transactions on Pattern Analysis and Machine Intelligence, 36(12):2538–2551, dec 2014.

Caglayan Dicle, Mario Sznaier, and Octavia Camps. The way they move: Tracking multiple targets with similar appearance. In International Conference on Computer Vision (ICCV), 2013.

Hamed Pirsiavash, Deva Ramanan, and Charless C. Fowlkes. Globally-optimal greedy algorithms for tracking a variable number of objects. In Computer Vision and Pattern Recognition (CVPR), pages 1201–1208. IEEE, jun 2011.

Amir Roshan Zamir, Afshin Dehghan, and Mubarak Shah. Gmcp-tracker: Global multi-object tracking using generalized minimum clique graphs. In European Conference on Computer Vision (ECCV), 2012.

Chang Huang, Bo Wu, and Ramakant Nevatia. Robust Object Tracking by Hierarchical Association of Detection Responses. In European Conference on Computer Vision (ECCV), pages 788–801, 2008.

Zdenek Kalal, Jiri Matas, and Krystian Mikolajczyk. P-N learning: Bootstrapping binary classifiers by structural constraints. In 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pages 49–56. Ieee, jun 2010.

Yoav Freund and Robert E Schapire. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. Journal of Computer and System Sciences, 55(1):119–139, aug 1997.

Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories. Computer Vision and Image Understanding, 106(1):59–70, apr 2007.

Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In Computer Vision and Pattern Recognition (CVPR), pages 1521–1528. IEEE, jun 2011.

A.G.a. Perera, C. Srinivas, A. Hoogs, and G. Brooksby. Multi-Object Tracking Through Simultaneous Long Occlusions and Split-Merge Conditions. In Computer Vision and Pattern Recognition (CVPR). IEEE, 2006.

Chang Huang, Bo Wu, and Ramakant Nevatia. Robust Object Tracking by Hierarchical Association of Detection Responses. In European Conference on Computer Vision (ECCV), pages 788–801, 2008.

Nicolas Pinto, David D. Cox, and James J. DiCarlo. Why is Real-World Visual Object Recognition Hard? PLoS Computational Biology, 4(1), 2008.

H. W. Kuhn. The Hungarian method for the assignment problem. Naval Research Logistics Quarterly, 2 :83–97, 1955.

Anton Milan, Stefan Roth, and Konrad Schindler. Continuous energy minimization for multitarget tracking. IEEE transactions on pattern analysis and machine intelligence, 36 (1):58–72, jan 2014.

Andreas Geiger, Martin Lauer, Christian Wojek, Christoph Stiller, and Raquel Urtasun. 3D Traffic Scene Understanding from Movable Platforms. Pattern Analysis and Machine Intelligence (PAMI), 2014.

L. Leal-Taixe, A. Milan, I. Reid, S. Roth, K. Schindler, ´ MOTChallenge 2015: Towards a benchmark for multi-target tracking, arXiv:1504.01942 [cs]ArXiv: 1504.01942. URL http://arxiv.org/abs/1504.01942.2015.

A. Bewley, Z. Ge, L. Ott, F. Ramos, B. Upcroft, Simple online and realtime tracking, in: IEEE International Conference on Image Processing (ICIP'16), 2016, pp. 3464–3468.

N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 2005.

P. F. Felzenszwalb, R. B. Girshick, D. McAllester, D. Ramanan, Object detection with discriminatively trained part-based models, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (9) (2010) 1627–1645.

P. Dollar, R. Appel, S. Belongie, P. Perona, Fast feature pyramids fo ´ r object detection, IEEE Trans. Pattern Anal. Mach. Intell. 36 (8) (2014) 1532–1545.

W. Nam, P. Dollar, J. H. Han, Local decorrelation for improved pedestrian detection, in: Advances in Neural Information Processing Systems (NIPS'14), 2014, pp. 424–432.

CL Zitnick and P Dollar. Edge Boxes: Locating Object Proposals from Edges. In European Conference on Computer Vision (ECCV), 2014.

D. M. Gavrila and S. Munder. Multi-cue Pedestrian Detection and Tracking from a Moving Vehicle. International Journal of Computer Vision, 73(1):41–59, jul 2006.

Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. Measuring the objectness of image windows. IEEE transactions on pattern analysis and machine intelligence, 34(11):2189–202, nov 2012.

J. Carreira and C. Sminchisescu. CPMC : Automatic Object Segmentation Using Constrained Parametric Min-cuts. TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, 34(7):1312–1328, 2012.

Santiago Manen, Matthieu Guillaumin, and Luc Van Gool. Prime Object Proposals with Randomized Prim's Algorithm. In International Conference on Computer Vision (ICCV), 2013.

Pekka Rantalankila, Juho Kannala, and Esa Rahtu. Generating Object Segmentation Proposals Using Global and Local Search. In 2014 IEEE Conference on Computer Vision and Pattern Recognition, pages 2417–2424. IEEE, jun 2014.

J. R. R. Uijlings, K. E. A. Sande, T. Gevers, and A. W. M. Smeulders. Selective Search for Object Recognition. International Journal of Computer Vision, 104(2):154–171, apr 2013.

Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient Graph-Based Image Segmentation. International Journal of Computer Vision, 59(2):167–181, sep 2004.

Ming-ming Cheng Ziming, Zhang Wen-yan Lin, and Philip Torr. BING: Binarized Normed Gradients for Objectness Estimation at 300fps. In Computer Vision and Pattern Recognition (CVPR), 2014.

Jan Hosang, Rodrigo Benenson, and B Schiele. How good are detection proposals, really? In British Machine Vision Conference (BMVC), 2014.

Ross Girshick. Fast R-CNN. In International Conference on Computer Vision, pages 1440– 1448, 2015.

Piotr Dollar, Christian Wojek, Bernt Schiele, and Pietro Perona. Pedestrian detection: A ´ benchmark. 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009b.

Jian Zhang, Sakrapee Paisitkriangkrai, and Chunhua Shen. An overview of fast pedestrian detection: Feature selection and cascade framework of boosted features. In International Conference on Multimedia, pages 1566–1567. IEEE, jun 2009.

Oncel Tuzel, Fatih Porikli, and Peter Meer. Human Detection via Classification on Riemannian Manifolds. In computer Vision and Pattern Recognition, pages 1–8. IEEE, jun 2007.

Oswaldo Ludwig Junior, David Delgado, Valter Goncalves, and Urbano Nunes. Trainable classifier-fusion schemes: An application to pedestrian detection. In 2009 12th International IEEE Conference on Intelligent Transportation Systems, number 1. IEEE, oct 2009.

O. Javed, S. Ali, and M Shah. Online Detection and Classification of Moving Objects Using Progressively Improving Detectors. In 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, volume 1, pages 696–701. IEEE, 2005.

Paul E. Rybski, Daniel Huber, Daniel D. Morris, and Regis Hoffman. Visual classification of coarse vehicle orientation using Histogram of Oriented Gradients features. In 2010 IEEE Intelligent Vehicles Symposium, pages 921–928. IEEE, jun 2010.

S. Agarwal, A. Awan, and D. Roth. Learning to detect objects in images via a sparse, part-based representation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 26 (11):1475–1490, nov 2004.

Mykhaylo Andriluka, Stefan Roth, and Bernt Schiele. People-tracking-by-detection and people detection-by-tracking. 2008 IEEE Conference on Computer Vision and Pattern Recognition, pages 1–8, jun 2008.

Hossein Azizpour and Ivan Laptev. Object detection using strongly-supervised deformable part models. In European Conference on Computer Vision (ECCV), 2012.

Guillaume Bouchard and Bill Triggs. Hierarchical Part-Based Visual Object Categorization. In Computer Vision and Pattern Recognition (CVPR), volume 1, pages 710–715. IEEE, 2005.

Pedro F. Felzenszwalb, David McAllester, and Deva Ramanan. A discriminatively trained, multiscale, deformable part model. In 2008 IEEE Conference on Computer Vision and Pattern Recognition, pages 1–8. IEEE, jun 2008.

Pedro F. Felzenszwalb, Ross B. Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. IEEE transactions on pattern analysis and machine intelligence, 32(9):1627–45, sep 2010.

Bastian Leibe, Ales Leonardis, and Bernt Schiele. Combined object categorization and segmentation with an implicit shape model. In ECCV'04 Workshop on Statistical Learning in Computer Vision, number May, pages 1–16, 2004.

Bo Wu and Ram Nevatia. Detection and Tracking of Multiple, Partially Occluded Humans by Bayesian Combination of Edgelet based Part Detectors. International Journal of Computer Vision, 75(2):247–266, jan 2007.

Long (Leo) Zhu, Yuanhao Chen, Alan Yuille, and William Freeman. Latent hierarchical structural learning for object detection. In 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pages 1062–1069. Ieee, jun 2010.

M. Bajracharya, B. Moghaddam, a. Howard, S. Brennan, and L. H. Matthies. A Fast Stereo-based System for Detecting and Tracking Pedestrians from a Moving Vehicle. The International Journal of Robotics Research, 28(11-12):1466–1485, jul 2009.

A Howard, L H Matthies, A Huertas, M Bajracharya, and A Rankin. Detecting pedestrians with stereo vision: safe operation of autonomous ground vehicles in dynamic environments. In Int. Symp. Of Robotics Research, 2007.

W. Eric L. Grimson, Chris Stauffer, Raquel Romano, and Lily Lee. Using adaptive tracking to classify and monitor activities in a site. In Computer Vision and Pattern Recognition (CVPR), 1998.

M. Heikkila; M. Pietik ainen, and J. Heikkil a. A Texture-based Method for Detecting Moving Objects. Procedings of the British Machine Vision Conference 2004, pages 21.1–21.10, 2004.

Marko Heikkila and Matti Pietik ainen. A texture-based method for modeling the background and detecting moving objects. IEEE transactions on pattern analysis and machine intelligence, 28(4):657–62, apr 2006.

M. Piccardi. Background subtraction techniques: a review. In 2004 IEEE International Conference on Systems, Man and Cybernetics, pages 3099–3104. Ieee, 2004.

A. Singh, S. Sawan, M. Hanmandlu, V.K. Madasu, and B.C. Lovell. An Abandoned Object Detection System Based on Dual Background Segmentation. In International Conference on Advanced Video and Signal Based Surveillance, pages 352–357. IEEE, sep 2009.

Thierry Bouwmans. Recent advanced statistical background modeling for foreground detection: A systematic survey. Recent Patents on Computer Science, 4(3):147–176, 2011.

Paulo Vinicius Koerich Borges. Pedestrian Detection Based on Blob Motion Statistics. IEEE Transactions on Circuits and Systems for Video Technology, 23(2):224–235, feb 2013.

A Elgammal, R. Duraiswami, D. Harwood, and L.S. Davis. Background and foreground modeling using nonparametric kernel density estimation for visual surveillance. Proceedings of the IEEE, 90(7):1151–1163, jul 2002.

Vikas Reddy, Conrad Sanderson, and Brian C. Lovell. Improved Foreground Detection via Block-Based Classifier Cascade With Probabilistic Decision Integration. IEEE Transactions on Circuits and Systems for Video Technology, 23(1):83–93, jan 2013.

Zoran Zivkovic and Ferdinand van der Heijden. Efficient adaptive density estimation per image pixel for the task of background subtraction. Pattern Recognition Letters, 27(7):773–780, may 2006.

Sijong Kim, Jungwon Kang, and MJ Chung. Monocular vision based independently moving feature detection using image correspondences. In 2012 12th International Conference on Control, Automation and Systems, pages 2181–2184, 2012.

Philip Lenz, Julius Ziegler, Andreas Geiger, and Martin Roser. Sparse scene flow segmentation for moving object detection in urban environments. In 2011 IEEE Intelligent Vehicles Symposium (IV), pages 926–932, Baden-Baden, Germany, jun 2011. IEEE.

M.D. Breitenstein, F. Reichlin, B. Leibe, E. KollerMeier et L. van Gool. Online Multiperson Tracking-by-Detection from a Single, Uncalibrated Camera. Dans : Pattern Analysis and Machine Intelligence, IEEE Transactions on 33.9 (2011).

M. Zhang et R. Alhajj. Content-Based Image Retrieval : From the Object Detection/Recognition Point of View. Dans : Artificial Intelligence for Maximizing Content Based Image Retrieval. Sous la dir. de Z. Ma. PA : Information Science Reference. Hershey, 2009.

A. Ess, K. Schindler, B. Leibe et L. Van Gool. Object Detection and Tracking for Autonomous Navigation in Dynamic Environments. Dans : The International Journal of Robotics Research 29.14 (2010).

D. Gerónimo, A. M. López, A. D. Sappa et T. Graf. Survey of Pedestrian Detection for Advanced Driver Assistance Systems. Dans: IEEE Transactions on Pattern Analysis and Machine Intelligence 32.7 (juil. 2010).

Xin Zhang, Yee-Hong Yang, Zhiguang Han, Hui Wang et Chao Gao. Object Class Detection: A Survey. Dans : ACM Comput. Surv. 46.1 (2013).

M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn et A. Zisserman. The Pascal Visual Object Classes (VOC) Challenge . Dans : International Journal of Computer Vision 88.2 (2010).

Mark Everingham, S. M. Ali Eslami, Luc Van Gool, Christopher K. I. Williams, John Winn et Andrew Zisserman. The Pascal Visual Object Classes Challenge: A Retrospective. Dans: International Journal of Computer Vision 111.1 (jan. 2015).

S. Walk, N. Majer, K. Schindler et B. Schiele. New features and insights for pedestrian detection. Dans : 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Juin 2010.

Christian Wojek et Bernt Schiele. A Performance Evaluation of Single and Multi-feature People Detection. Dans: Pattern Recognition: 30th DAGM Symposium Munich, Germany, June 10-13, 2008 Proceedings. Sous la dir. de Gerhard Rigoll. Berlin, Heidelberg : Springer Berlin Heidelberg, 2008.

Jai Deng, Wei Dong, Richard Socher, Li-Jai Li, Kai Li, and Li Fei-Fei. Image Net: A large scale hierarchical image database. 2009 IEEE Conference on Computer Vision and Pattern Recognition, pages 248–255, jun 2009.

Jesse Davis and Mark Goadrich. The relationship between Precision-Recall and ROC curves. Proceedings of the 23rd international conference on Machine learning - ICML'06, pages 233–240, 2006.

Piotr Dollar, Zhuowen Tu, Pietro Perona, and Serge Belongie. Integral Channel Features. BMVC 2009 London England, pages 1–11, 2009a.

Piotr Dollar, Christian Wojek, Bernt Schiele, and Pietro Perona. Pedestrian detection: A benchmark. 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009b.

Anand Singh Jalal. The State-of-the-Art in Visual Object Tracking. Informatica, 36 :227–248, 2012.

Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. Online Object Tracking: A Benchmark. In 2013 IEEE Conference on Computer Vision and Pattern Recognition, pages 2411–2418. IEEE, jun 2013.

Sam Hare, Amir Saffari, and Philip H. S. Torr. Struck: Structured output tracking with kernels. In International Conference on Computer Vision (ICCV), pages 263–270. IEEE, nov 2011.

Je rome Berclaz, Franc ois Fleuret, Engin Tu retken, and Pascal Fua. Multiple Object Tracking Using K-Shortest Paths Optimization. IEEE Transactions on Pattern Analysis and Machine Intelligence, 33(9):1806–1819, 2011.

Hingorani J Cox and Sunita L Hingorani. An Efficient Implementation of Reid's Multiple Hypothesis Tracking Algorithm and Its Evaluation for the Purpose of Visual Tracking. Pattern Analysis and Machine Intelligence, 18(2):138–150, feb 1996.

Anand Singh Jalal. The State-of-the-Art in Visual Object Tracking. Informatica, 36:227–248, 2012.

Chanho Kim, Fuxin Li, Arridhana Ciptadi, and James M. Rehg. Multiple Hypothesis Tracking Revisited. In International Conference on Computer Vision (ICCV), 2015.

Songhwai Oh Songhwai Oh, S. Russell, and S. Sastry. Markov chain Monte Carlo data association for general multiple-target tracking problems. Conference on Decision and Control (CDC), 1:735–742, 2004.

R E Kalman. A New Approach to Linear Filtering and Prediction Problems. Journal of Basic Engineering, 82(Series D):35–45, 1960.

Keni Bernardin and Rainer Stiefelhagen. Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics. Image and Video Processing, (May), 2008.

Yuan Li, Chang Huang, and Ram Nevatia. Learning to associate: HybridBoosted multi-target tracker for crowded scene. In Computer Vision and Pattern Recognition (CVPR), pages 2953– 2960. IEEE, jun 2009.

Jan Hosang, Rodrigo Benenson, and B Schiele. How good are detection proposals, really? In British Machine Vision Conference (BMVC), 2014.

Z. Zivkovic. Improved adaptive Gaussian mixture model for background subtraction. In: Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004. Vol. 2. 2004, 28–31 Vol.2. DOI: 10.1109/ICPR.2004. 1333992.

Bruce D. Lucas and Takeo Kanade. An Iterative Image Registration Technique with an Application to Stereo Vision. In: Proceedings of the 7th International Joint Conference on Artificial Intelligence - Volume 2. IJCAI'81. Vancouver, BC, Canada: Morgan Kaufmann

Publishers Inc., 1981, pp. 674–679. URL : http : //dl.acm.org/citation.cfm?id=1623264.1623280.

Simo Srkk. Bayesian Filtering and Smoothing. New York, NY, USA: Cambridge University Press, 2013. ISBN: 1107619289, 9781107619289.

Jack Edmonds and Richard M. Karp. Theoretical Improvements in Algorithmic Efficiency for Network Flow Problems. In: J. ACM 19.2 (Apr. 1972), pp. 248– 264. ISSN: 0004-5411. DOI: 10.1145/321694.321699. URL: http://doi. acm.org/10.1145/321694.321699.

Wenhan Luo, Junliang Xing, Anton Milan, Xiaoqin Zhang, Wei Liu, Xiaowei Zhao, and Tae-Kyun Kim. Multiple object tracking : A literature review. arXiv preprint arXiv :1409.7618, 2014.

Bo Li, Junjie Yan, Wei Wu, Zheng Zhu, and Xiaolin Hu. High performance visual tracking with siamese region proposal network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 8971–8980, 2018.

Luca Bertinetto, Jack Valmadre, Joao F Henriques, Andrea Vedaldi, and Philip HS Torr. Fully-convolutional siamese networks for object tracking. In European conference on computer vision, pages 850–865. Springer, 2016.

Laura Leal-Taixé, Cristian Canton-Ferrer, and Konrad Schindler. Learning by tracking: Siamese cnn for robust target association. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pages 33–40, 2016.

David Held, Sebastian Thrun, and Silvio Savarese. Learning to track at 100 fps with deep regression networks. In European Conference on Computer Vision, pages 749– 765. Springer, 2016.

CAVIAR-Project, CAVIAR test case scenarios (2004). http://groups.inf.ed.ac.uk/vision/CAVIAR/CAVIARDATA1/

J. Ferryman, A. Shahrokni, Pets2009 : Dataset and challenge, in: IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, 2009, pp. 1–6.

M. Andriluka, S. Roth, B. Schiele, Monocular 3d pose estimation and tracking by detection, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR'10), 2010.

A. Ess, B. Leibe, K. Schindler, L. van Gool, Robust multi person tracking from a mobile platform, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (10) (2009) 1831–1846.

M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, A. Zisserman, The pascal visual object classes (VOC) challenge, International Journal of Computer Vision 88 (2) (2010) 303–338.

Rainer Stiefelhagen, Keni Bernardin, Rachel Bowers, John Garofolo, Djamel Mostefa, and Padmanabhan Soundararajan. The clear 2006 evaluation. In International evaluation workshop on classification of events, activities and relationships, pages 1–44. Springer, 2006.

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 779–788, 2016.

Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng Yang Fu, and Alexander C Berg. Ssd : Single shot multibox detector. In European conference on computer vision, pages 21–37. Springer, 2016.

Samuel Murray. Real-time multiple object tracking-a study on the importance of speed. arXiv preprint arXiv :1709.03572, 2017.

Ricardo Sanchez-Matilla, Fabio Poiesi, and Andrea Cavallaro. Online multi-target tracking with strong and weak detections. In European Conference on Computer Vision, pages 84–99. Springer, 2016.

Fengwei Yu, Wenbo Li, Quanquan Li, Yu Liu, Xiaohua Shi, and Junjie Yan. Poi: Multiple object tracking with high performance detection and appearance feature. In European Conference on Computer Vision, pages 36–42. Springer, 2016.

Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In 2017 IEEE International Conference on Image Processing (ICIP), pages 3645–3649. IEEE, 2017.

[S1] : Pymotmetrics GitHub Repository. Disponible : https://github.com/cheind/py-motmetrics.

[S2] : Méthode Hongroise. Disponible : https://fr.wikipedia.org/wiki/Algorithme_hongrois.