Apprentissage semi-supervisé pour la détection multi-objets dans des séquences vidéos : Application à l'analyse de flux urbains

Houda Maâmatou

N° d'ordre : D. U : 2805
E D S P I C : 796

UNIVERSITE CLERMONT AUVERGNE

ECOLE DOCTORALE
SCIENCES POUR L'INGENIEUR DE CLERMONT-FERRAND

T h è s e

Présentée par

HOUDA MAAMATOU

Ingénieur en Informatique temps réel & Master en Systèmes Intelligents et communicants

pour obtenir le grade de

DOCTEUR D'UNIVERSITÉ

SPECIALITE : INFORMATIQUE

Apprentissage semi-supervisé pour la détection multi-objets dans des séquences vidéos: Application à l'analyse de flux urbains

Soutenue publiquement le 05 avril 2017 devant le jury :

M. Vincent CHARVILLAT	Président et examinateur
M. Michel DHOME	Examinateur
M. Nabil DERBEL	Rapporteur
M. Serge MIGUET	Rapporteur
M. Sami GAZZAH	Invité
M. Yann GOYAT	Invité
M. Thierry CHATEAU	Directeur de thèse
Mme. Najoua ESSOUKRI BEN AMARA	Directrice de thèse

N° d'ordre : D.U : 2805
EDSPIC : 796

N° d'ordre : 2017-xxxx
EDST :

Université Clermont Auvergne
Clermont-Ferrand II
Ecole Doctorale Sciences Pour l'Ingénieur

Université de Sfax
École Nationale d'Ingénieurs de Sfax
Ecole Doctorale Sciences et Technologies

Thèse
présentée par :
Houda MAAMATOU

pour obtenir les grades de

Docteur de l'Université Clermont Auvergne

Spécialité : Informatique

et

Docteur de l'Université de Sfax

Spécialité : Ingénierie des Systèmes Informatiques

Apprentissage semi-supervisé pour la détection multi-objets dans des séquences vidéos : Application à l'analyse de flux urbains

---

Soutenue publiquement le mercredi 05 avril 2017 devant le jury :

M. Vincent CHARVILLAT	Pr Univ. de Toulouse	Président
M. Michel DHOME	Directeur de recherche CNRS	Examinateur
M. Nabil DERBEL	Pr ENIS, Univ. de Sfax	Rapporteur
M. Serge MIGUET	Pr Univ. Lumière Lyon 2	Rapporteur
M. Sami GAZZAH	Dr ISITCom, Univ. de Sousse	Invité
M. Yann GOYAT	Dr Logiroad	Invité
M. Thierry CHATEAU	Pr Univ. Clermont Auvergne	Directeur de thèse
Mme. Najoua ESSOUKRI BEN AMARA	Pr ENISo, Univ. de Sousse	Directrice de thèse

Thèse préparée dans un cadre de collaboration entre l'Institut Pascal (Clermont-Ferrand), le laboratoire LATIS : Laboratory of Advanced Technology and Intelligent Systems (Sousse) et l'entreprise Logiroad (Nantes).

"The only true wisdom is in knowing you know nothing."
Socrates

Remerciements

Je voudrais tout d'abord remercier tous les membres du jury pour avoir accepté d'évaluer mes travaux de thèse et pour avoir souligné les contributions et les efforts apportés à la réalisation de cette thèse. Merci à Mr. Vincent CHARVILLAT d'avoir présidé ce jury et examiné mes travaux. Merci également à Mr. Michel DHOME d'avoir examiné cette thèse. Je tiens également à remercier les deux rapporteurs, Mr. Nabil DERBEL et Mr. Serge MIGUET, pour la lecture de mon manuscrit et pour les commentaires concernant mes travaux de thèse.

Je souhaite aussi adresser mes remerciements les plus profonds à Mme. Najoua ESSOUKRI BEN AMARA et Mr. Sami GAZZAH, d'avoir dirigé et encadré mes travaux au sein du laboratoire LATIS. Je les remercie pour m'avoir facilité mes travaux, m'offrant à la fois de bonnes conditions de travail et une liberté de mener mes recherches. Je désire également exprimer toute ma gratitude à Mr. Thierry CHATEAU qui a dirigé la majeure partie de mes travaux de thèse au sein de l'Institut Pascal. Je le remercie pour son soutien et ses compétences scientifiques. Il m'a laissé beaucoup de liberté dans mon travail et en même temps il est toujours disponible à l'écoute et efficace avec ses conseils. Je voudrais également remercier Mr. Yann GOYAT le président de Logiroad d'avoir financé cette thèse avec les moindres exigences industrielles. Je le remercie pour sa prise de risque, sa bonne humeur et ses encouragements.

Cette thèse s'est déroulée dans le cadre d'une collaboration entre l'Institut Pascal, le laboratoire LATIS (unité de recherche SAGE précédemment) et l'entreprise Logiroad. Je tiens à remercier Mr. Jean-Pierre DERUTIN qui a été à l'origine de la création de cette collaboration. Je remercie également toutes les personnes rencontrées dans les trois établissements partenaires de cette collaboration.

Je remercie Céline TEULIERE avec laquelle j'ai eu l'occasion d'échanger, pour m'avoir conseillée et pour avoir corrigé les productions scientifiques de ma thèse. Je remercie également Eliane DE DEA pour son amitié et pour la correction des fautes d'orthographes de ce manuscrit de thèse.

Je remercie aussi tous ceux qui ont aidée et simplifiée la préparation d'une ou de plusieurs parties de mes travaux de recherche. Merci en particulier à Pierre BOUGES, Olivier PITARD, Najla RAGOUBI, Ala MHALLA, Datta RAMADASAN, Alexandre BOYER, Christelle BALLUT, Ghina ELNATOUR.

Mes remerciements vont aussi à toutes mes amies avec lesquelles j'ai passé des bons moments dans mon studio (studios des cézeaux, bat 22 Log 021) : Kaoutar, Chaima, Nawel, Essia, Amel, Lobna, Zeinabou, Mariem, Ibtissem, Sahar MAAMAR, Nadia, Akila, Yosra, khouloud, Rawan, Karima, Salsabil, Soumaya. Je remercie également tous et toutes mes ami(e)s qui m'ont aidé d'une sorte ou d'une autre : Houda, Samira, Khaoula, Nejla MANSOUR, Sawssen, Abir, Amira, Zeineb, Ala, Mohamed Mehdi, Lotfi et Mariem BEN ALI.

Mes derniers mots vont à ma précieuse famille. Un grand merci à mon père qui m'a toujours entourée par son amour, son sacrifice, sa confiance qui m'a été davantage une cause pour avancer et m'a donnée l'énergie pour progresser... Je remercie chaleureusement ma mère la fleur de ma vie, pour ses mots qui m'approvisionnent d'espoir, son amour qui ne cessera jamais, sa patience sans égale... Mes chers parents, je vous déclare mes hommages et mon respect et j'espère que dieu vous offre la vie et la santé.

Je remercie mon frère Hatem pour ses encouragements, ses motivations et ses soutiens. Cher frère, je te souhaite une belle vie pleine d'entente avec ton épouse et ton adorable fils Jassem. Je remercie mon frère Kais pour son respect, son assistance, pour ses conseils dans les moments les plus difficiles de ma vie. Cher frère, je te souhaite une vie pleine d'amour avec ton épouse et tes magnifiques enfants Bayrem et Loujaine. Mes chers frères, je vous aime sans limite.

Un grand merci à mon affectueuse sœur Hayet, pour son amour incomparable qui m'a fait preuve depuis longtemps. Chère sœur, je te souhaite une vie pleine de bénédiction et de joie avec ton mari, tes petites anges Amna, Nour El-Houda, Tasnim et Habiba. Un grand merci à ma mignonne sœur Wafa qui m'a partagé les détails de ma vie, m'a été toujours à l'écoute et m'a été la sœur et l'amie. Chère sœur, je te souhaite une vie pleine d'amour avec ton mari et ton admirable fils Fadi. Mes chères sœurs, je vous adore de tout mon cœur.

Résumé

Depuis les années 2000, un progrès significatif est enregistré dans les travaux de recherche qui proposent l'apprentissage de détecteurs d'objets sur des grandes bases de données étiquetées manuellement et disponibles publiquement. Cependant, lorsqu'un détecteur générique d'objets est appliqué sur des images issues d'une scène spécifique les performances de détection diminuent considérablement. Cette diminution peut être expliquée par les différences entre les échantillons de test et ceux d'apprentissage au niveau des points de vues prises par la(les) caméra(s), de la résolution, de l'éclairage et du fond des images.

De plus, l'évolution de la capacité de stockage des systèmes informatiques, la démocratisation de la "vidéo-surveillance" et le développement d'outils d'analyse automatique des données vidéos encouragent la recherche dans le domaine du trafic routier. Les buts ultimes sont l'évaluation des demandes de gestion du trafic actuelles et futures, le développement des infrastructures routières en se basant sur les besoins réels, l'intervention pour une maintenance à temps et la surveillance des routes en continu. Par ailleurs, l'analyse de trafic est une problématique dans laquelle plusieurs verrous scientifiques restent à lever. Ces derniers sont dus à une grande variété dans la fluidité de trafic, aux différents types d'usagers, ainsi qu'aux multiples conditions météorologiques et lumineuses.

Ainsi le développement d'outils automatiques et temps réel pour l'analyse vidéo de trafic routier est devenu indispensable. Ces outils doivent permettre la récupération d'informations riches sur le trafic à partir de la séquence vidéo et doivent être précis et faciles à utiliser. C'est dans ce contexte que s'insèrent nos travaux de thèse qui proposent d'utiliser les connaissances antérieurement acquises et de les combiner avec des informations provenant de la nouvelle scène pour spécialiser un détecteur d'objet aux nouvelles situations de la scène cible.

Dans cette thèse, nous proposons de spécialiser automatiquement un classifieur/détecteur générique d'objets à une scène de trafic routier surveillée par une caméra fixe. Nous présentons principalement deux contributions. La première est une formalisation originale de transfert d'apprentissage transductif à base d'un filtre séquentiel de type Monte Carlo pour la spécialisation automatique d'un classifieur. Cette formalisation approxime itérativement la distribution cible inconnue au départ, comme étant un ensemble d'échantillons de la base spécialisée à la scène cible. Les échantillons de cette dernière sont sélectionnés à la fois à partir de la base source et de la scène cible moyennant une pondération qui utilise certaines informations *a priori* sur la scène. La base spécialisée obtenue permet d'entrainer un classifieur spécialisé à la scène cible sans intervention humaine. La deuxième contribution consiste à proposer deux stratégies d'observation pour l'étape mise à jour du filtre SMC. Ces stratégies sont à la base d'un ensemble d'indices spatio-temporels spécifiques à la scène de vidéo-surveillance. Elles sont utilisées pour la pondération des échantillons cibles.

Les différentes expérimentations réalisées ont montré que l'approche de spécialisation proposée est performante et générique. Nous avons pu y intégrer de multiples stratégies d'observation. Elle peut être aussi appliquée à tout type de classifieur. De plus, nous avons implémenté dans le logiciel OD SOFT de Logiroad les possibilités de chargement et d'utilisation d'un détecteur fourni par notre approche. Nous avons montré également les avantages des détecteurs spécialisés en comparant leurs résultats avec celui de la méthode Vu-mètre de Logiroad.

Mots-clés : Apprentissage semi-supervisé, transfert d'apprentissage, spécialisation, filtre Séquentiel de Monte Carlo, classification/détection d'objets de trafic urbain, vidéo-surveillance.

Abstract

Title : Semi-supervised learning for multi-object detection in video sequences : Application to the analysis of urban flows

Since 2000, a significant progress has been recorded in research work which has proposed to learn object detectors using large manually labeled and publicly available databases. However, when a generic object detector is applied on images of a specific scene, the detection performances will decrease considerably. This decrease may be explained by the differences between the test samples and the learning ones at viewpoints taken by camera(s), resolution, illumination and background images.

In addition, the storage capacity evolution of computer systems, the "video surveillance" democratization and the development of automatic video-data analysis tools have encouraged research into the road-traffic domain. The ultimate aims are the management evaluation of current and future traffic requests, the road infrastructures development based on real necessities, the intervention of maintenance task in time and the continuous road surveillance. Moreover, traffic analysis is a problematicness where several scientific locks should be lifted. These latter are due to a great variety of traffic fluidity, various types of users, as well multiple weather and lighting conditions.

Thus, developing automatic and real-time tools to analyse road-traffic videos has become an indispensable task. These tools should allow retrieving rich data concerning the traffic from the video sequence and they must be precise and easy to use. This is the context of our thesis work which proposes to use previous knowledges and to combine it with information extracted from the new scene to specialize an object detector to the new situations of the target scene.

In this thesis, we propose to automatically specialize a generic object classifier/detector to a road traffic scene surveilled by a fixed camera. We mainly present two contributions. The first one is an original formalization of Transductive Transfer Learning based on a sequential Monte Carlo filter for automatic classifier specialization. This formalization approximates iteratively the previously unknown target distribution as a set of samples composing the specialized dataset of the target scene. The samples of this dataset are selected from both source dataset and target scene further to a weighting step using some prior information on the scene. The obtained specialized dataset allows training a specialized classifier to the target scene without human intervention. The second contribution consists in proposing two observation strategies to be used in the SMC filter's update step. These strategies are based on a set of specific spatio-temporal cues of the video surveillance scene. They are used to weight the target samples.

The different experiments carried out have shown that the proposed specialization approach is efficient and generic. We have been able to integrate multiple observation strategies. It can also be applied to any classifier / detector. In addition, we have implemented into the Logiroad OD SOFT software the loading and utilizing possibilities of a detector provided by our approach. We have also shown the advantages of the specialized detectors by comparing their results to the result of Logiroad's Vu-meter method.

Keywords : Semi-Supervised Learning, Transfer learning, Specialization, Sequential Monte Carlo filter, Urban traffic objects detection/classification, video-surveillance.

Table des matières

Liste des figures

Liste des tableaux

Introduction

La vision par ordinateur est un domaine de recherche très actif. Son objectif principal est de permettre à un "système artificiel intelligent" d'analyser et d'interpréter automatiquement le contenu des images (ou d'une séquence vidéo) capturées à l'aide d'une ou de plusieurs caméras. L'analyse consiste à déterminer la catégorie, la position et le mouvement de chaque objet présent dans la scène en exploitant souvent des connaissances a priori sur la scène. Parmi les applications de vision par ordinateur qui ont gagné l'intérêt des travaux de recherches, nous mentionnons : la biométrie pour la vérification et l'identification des individus, la robotique, la réalité augmentée, la vidéo-surveillance, l'assistance aux conducteurs et l'analyse de trafic routier.

Les travaux de cette thèse portent sur la détection multi-objets à base d'apprentissage pour l'analyse automatique des scènes de vidéo-surveillance du trafic routier.

Cette thèse s'inscrit dans un cadre d'une cotutelle entre l'université Clermont Auvergne (Clermont Ferrand, France) et l'université de Sfax (Sfax, Tunisie). Elle est préparée au sein du laboratoire Institut Pascal (Clermont Ferrand, France), du laboratoire LATIS (Laboratory of Advanced Technology and Intelligent Systems) de l'Ecole Nationale d'Ingénieurs de Sousse (Sousse, Tunisie) et l'entreprise Logiroad (Nantes, France) qui a financé ces travaux dans le cadre d'une convention CIFRE.

Dans ce chapitre d'introduction, nous présentons le contexte scientifique et le contexte applicatif des travaux de notre thèse.

L'utilisation de techniques de type machine learning dans la détection et le suivi d'objets est devenue de plus en plus populaire. Les travaux de cette thèse sont issus principalement de l'apprentissage artificiel et traitent la détection d'objets dans des séquences vidéos de surveillance de trafic routier.

1 Contexte scientifique

Cette section introduit les notions d'apprentissage artificiel et ses principaux types, les notions de détection de catégories d'objets ainsi que les challenges associés à la détection d'objets.

1.1 L'apprentissage artificiel

L'apprentissage artificiel ou automatique dans le domaine de la vision artificielle cherche à constituer une représentation de l'objet d'intérêt "modèle" en se basant sur une base de données. Celle-ci est composée d'un ensemble d'images dites positives contenant l'objet cherché et un ensemble d'exemples négatifs où les images ne contiennent pas cet objet. Le modèle fourni en sortie permet d'associer à une image donnée une étiquette représentant la catégorie de l'objet.

Dans le cas de cette thèse, le terme "objet" fera référence à tout objet de trafic routier tel que piéton, voiture, moto, vélo, bus.

L'apprentissage artificiel se divise principalement en trois catégories : apprentissage supervisé, apprentissage non supervisé et apprentissage semi-supervisé. La Figure 1 présente un exemple d'un modèle de classification sorti d'un apprentissage artificiel supervisé. Ce dernier prend en entrée une base d'apprentissage étiquetée. La classification s'effectue sur un nouvel ensemble d'échantillons dit base de test.

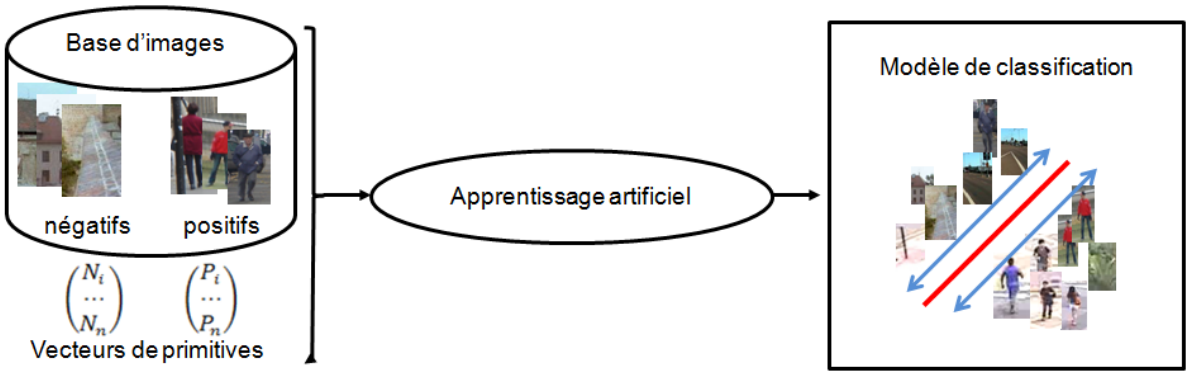


FIGURE 1 – Exemple de sortie d'un apprentissage artificiel : Un modèle de classification de type *Support Vector Machines* (SVM) après entrainement sur une base d'apprentissage étiquetée prise en entrée. La classification est réalisée sur une nouvelle base de test

— *Apprentissage supervisé :* Le classifieur s'entraîne sur des données étiquetées de toutes les classes recherchées (ou à étudier) qui sont connues dès le départ. Dans ce cas, l'algorithme utilise les données d'apprentissage et leurs étiquettes associées comme des instances du problème à résoudre pour ajuster les paramètres de la fonction de décision. Parmi les méthodes d'apprentissage supervisé nous pouvons citer les méthodes de régression linéaire, l'algorithme Adaboost, les machines à vecteur de support, les k plus proches voisins.
— *Apprentissage non-supervisé :* L'algorithme utilise des données non étiquetées qui présentent un ensemble de données sans aucune information à propos des classes de leur appartenance. Il doit donc découvrir par lui-même les corrélations existantes entre les exemples d'apprentissage pour trouver les classes les plus pertinentes. Parmi les méthodes d'apprentissage non-supervisé on peut citer les méthodes de "clustering" où l'objectif est de regrouper des données proches en différents ensembles appelés clusters. Ainsi, nous citons les méthodes d'extraction de règles d'association qui calculent des probabilités conditionnelles pour chaque variable aléatoire en tenant compte des autres variables.
— *Apprentissage semi-supervisé :* L'algorithme combine simultanément des données étiquetées et non-étiquetées au cours de la phase d'apprentissage et les différentes classes sont connues à l'avance. Dans ces conditions, la combinaison de deux catégories de données permet d'améliorer significativement la qualité de l'apprentissage notamment lorsque les jeux de données sont très grands où l'étiquetage manuel peut s'avérer fastidieux. Parmi les méthodes de ce mode d'apprentissage, on peut distinguer le "clustering" semi-supervisé et la classification semi-supervisée. Le but de "clustering " est de regrouper les instances les plus similaires entre elles en exploitant l'information apportée par les données étiquetées. Par contre, la classification utilise les données étiquetées pour séparer les instances des différentes classes. Elle affine la fonction de classification à l'aide des données non étiquetées.

Dans cette thèse nous nous intéressons tout particulièrement à la catégorie d'apprentissage semi-supervisé.

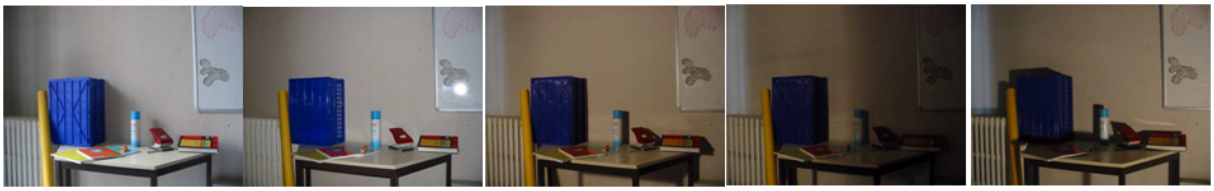
1.2 La détection d'objets

La détection d'objets dans une image ou dans une vidéo est la première information importante à extraire dans plusieurs applications de vision par ordinateur telles que la vidéo surveillance, le suivi de personnes, la surveillance de trafic, l'annotation sémantique des vidéos. La détection d'une catégorie d'objet consiste à déterminer si une ou plusieurs instances de cette catégorie d'objet sont présentes

dans une image. Dans le cas de présence, l'algorithme renvoie la segmentation de la zone contenant l'objet ou les coordonnées du rectangle englobant l'instance détectée.

Malgré la présence de méthodes de détection robustes de certaines catégories d'objets tels que la détection de visages et de piétons, les performances se dégradent généralement si on est exposé à un nombre élevé de catégories d'objets (e.g. chaises, chiens, chats, bus, camions, motos,...) et/ou à une large variation intra-classe qui introduit d'autres variations dans la forme, texture, couleur, etc. En effet, un algorithme de détection d'objet doit trouver l'objet d'intérêt, en dépit des changements de son échelle et de sa position dans l'image, des variations d'illumination et du fond, des occultations, des différents points de vues et de la variabilité intra-classe. Néanmoins, concevoir un détecteur d'objet qui résout tous ces problèmes est une tâche non triviale. La FIGURE 2 illustre certains défis qu'un détecteur d'objet doit surmonter.

(a) Différents points de vues [Hou *et al.*, 2007]



(b) Différentes conditions d'illumination [Angulo et Marcotegui, 2005]

(c) Variations intra-classe

FIGURE 2 – Exemples de challenges à surmonter en détection d'objets.

La section 1 du chapitre 1 donne un aperçu sur les différentes catégories des méthodes de détection d'objet dans une vidéo.

2 Contexte applicatif

Cette thèse rentre dans le cadre d'une convention CIFRE avec l'entreprise Logiroad [1]. Logiroad est une société d'édition de logiciels d'aide à la décision dans le domaine de l'entretien et de l'exploitation des réseaux routiers, aux services des gestionnaires routiers. Pour le domaine d'exploitation, elle pro-

1. http ://www.logiroad.fr/

pose d'analyser automatiquement le trafic routier à partir de séquences vidéos. Étant donné une ou plusieurs caméras statiques qui surveillent une scène routière, l'analyse automatique de trafic prend en entrée des séquences d'images/vidéos pour donner en sortie un ensemble de statistiques exploitables sous forme d'un comptage par catégorie, comptage directionnel, suivi de trajectoire de chaque objet et calcul de matrice origine/destination.

Afin d'atteindre ses objectifs, Logiroad propose une solution qui se base sur une méthode composée de trois étapes : soustraction fond-forme, détection des blobs et mise en correspondance des blobs détectés. La FIGURE 3 illustre une représentation simplifiée de la solution utilisée initialement par Logiroad.

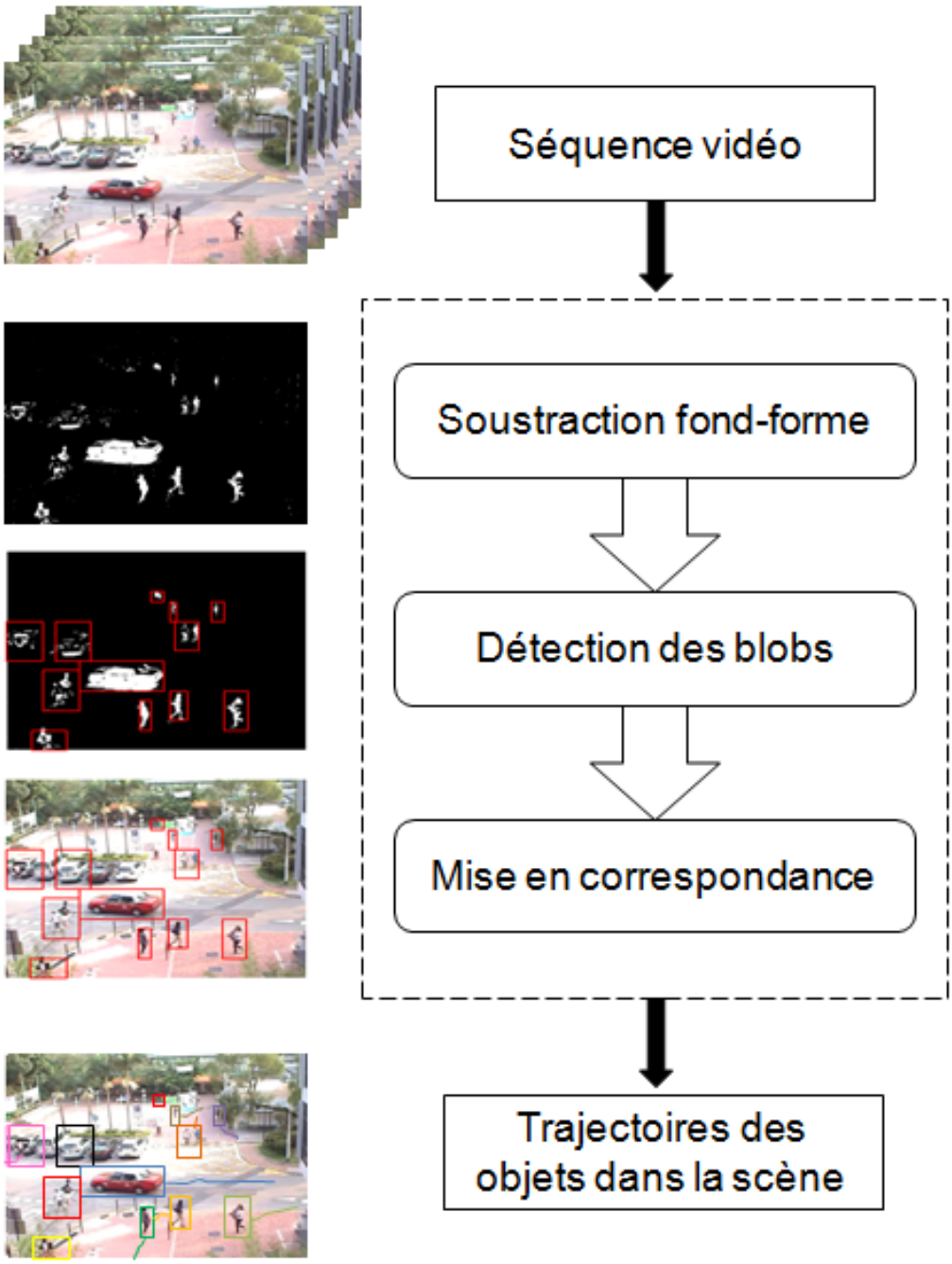
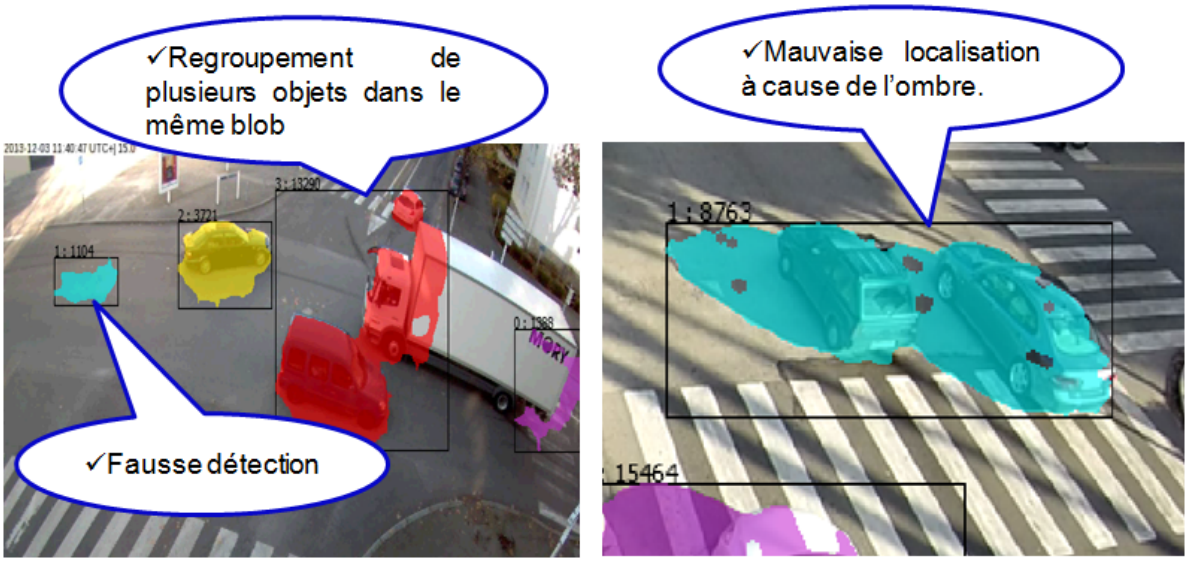


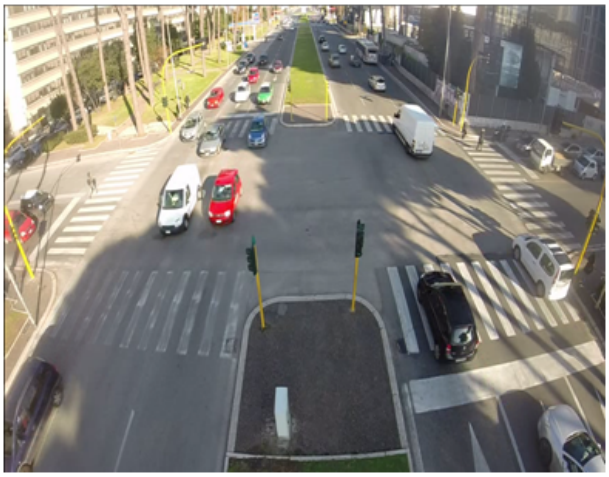
FIGURE 3 – Organigramme de la méthode initiale de Logiroad

Cependant, les scènes routières présentent certaines contraintes telles que différentes conditions d'illumination, congestion, plusieurs catégories d'objets(véhicules lourds/légers, 2 roues ou plus, piétons, bus,...), effet de l'horaire d'enregistrement sur la séquence vidéo (heure de pointe ou heure ordinaire, jour ou nuit), type de trafic (simple ou dense) et/ou multiples types d'infrastructures routières (intersection, bifurcation, rond-point,...). Ces contraintes rendent l'analyse de la scène une tâche complexe et limitent les performances de la solution utilisée à traiter des scènes de trafic dense et/ou produisant de l'ombre. La FIGURE 4 visualise certaines difficultés rencontrées par la méthode développée par Logiroad.



(a) Exemples de situations d'échecs de la méthode.

(b) Exemple de scène de trafic dense



(c) Exemple de scène présentant l'ombre

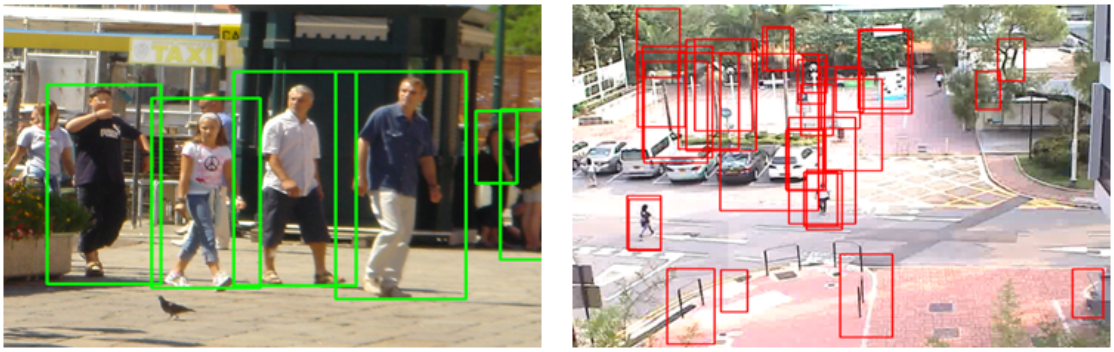
FIGURE 4 – Difficultés de l'analyse de trafic par la méthode de Logiroad

Dans cette thèse, nous proposons un algorithme de détection multi-objets basé sur un apprentissage capable de traiter deux grandes difficultés : des situations de trafic dense et d'illumination complexe.

1. Une séquence de trafic dense est une scène où les véhicules sont très proches, ce qui rend leur segmentation difficile en se basant sur l'extraction fond-forme. D'un autre côté, lorsque la vidéo présente une telle situation, un pixel peut être classé "objet" (classé "forme") sur une longue durée et par conséquence la mise à jour du fond devient complexe.

2. Une scène d'illumination complexe comme par exemple, lorsque le soleil provoque des ombres avec portées importantes et très perturbatrices lors de l'extraction des objets. Ces ombres sont considérées comme des formes dans la plupart des méthodes d'extraction de fond.

Un point clé de l'apprentissage des détecteurs basés sur les apparences d'objets est la création d'une base de données, où des milliers d'échantillons manuellement étiquetés sont nécessaires. Cependant la majorité des travaux montre que la performance d'un détecteur générique chute de manière significative lorsqu'il est appliqué à une séquence particulière (FIGURE 5 présente un exemple de cette situation). Les échantillons de la base d'apprentissage sont dans leur majorité différents de ceux de la base cible.



(a) Résultat de détection sur une image Test INRIA. (b) Résultat de détection sur une image de la scène CUHK

FIGURE 5 – Chute des performances d'un détecteur générique (entrainé sur la base INRIA Person dataset) lors d'application à la scène CUHK

En plus de la dépendance du classifieur de la base utilisée lors de la phase d'apprentissage, entraîner un seul détecteur pour gérer tous les scénarios urbains s'avère une tâche fastidieuse à cause de la particularité des scènes routières. Une scène de circulation urbaine peut présenter des changements d'apparence dû à l'évolution de temps au cours de la journée et des saisons. La FIGURE 6 montre un exemple de variation d'apparence d'une scène capturée de même point de vue.

Une solution intuitive pour éviter la chute de performance est d'utiliser des échantillons étiquetés de la scène cible pour générer un détecteur spécialisé à la scène étudiée. Ce dernier donne une meilleure performance par rapport à un détecteur générique. Néanmoins, l'étiquetage manuel des données pour chaque scène et la répétition d'apprentissage autant de fois que le nombre de classes d'objets dans la scène cible sont deux tâches ardues qui demandent énormément de temps. La solution fonctionnelle pour contourner ceci est de collecter et d'étiqueter automatiquement des échantillons de la scène cible et de transférer uniquement les échantillons utiles à partir de la base de données source vers la base cible spécialisée. Notre travail se dirige vers une solution qui spécialise un détecteur générique vers une scène cible. Cette solution cherche à maximiser les performances d'un détecteur d'objet à toute nouvelle scène tout en minimisant l'intervention humaine et en accélérant le temps d'adaptation vers la nouvelle scène.

Le schéma bloc de la spécialisation que nous proposons est représenté sur la FIGURE 7. C'est une formalisation originale du transfert d'apprentissage transductif basé sur un filtre séquentiel de Monte Carlo [Doucet *et al.*, 2001](en anglais : *Transductive Transfer Learning (TTL) based on a Sequential Monte Carlo (SMC)*). Elle permet de spécialiser un classifieur générique à une scène cible. Dans le filtre proposé, nous commençons par une étape de prédiction qui cherche des échantillons de la scène cible. Ensuite, nous déterminons la pertinence de ces échantillons durant l'étape de mise à jour. L'étape d'échantillonnage sélectionne des échantillons à la fois à partir de la base source et de la base cible pour créer une nouvelle base spécialisée. Cette dernière sera utilisée pour entrainer un détecteur spécialisé à

FIGURE 6 – Exemple de variation d'apparence d'une même scène capturée de même point de vue à différentes heure de jour.

l'itération suivante. Une fois que le critère d'arrêt est atteint, nous aurons en sortie le dernier classifieur spécialisé et la base de données spécialisée associée.

FIGURE 7 – Schéma général de notre approche proposée de spécialisation.

3 Contributions

Nos principales contributions sont les suivantes :

(1) *Une formalisation originale pour la spécialisation d'un classifieur basée sur le filtre SMC (TTL-SMC) :* Cette formalisation approxime la distribution cible inconnue comme étant un ensemble d'échantillons composant la base de données spécialisée. Les objectifs de cette formalisation sont la sélection des échantillons pertinents tout en considérant les poids attribués lors de l'étape mise à jour et l'apprentissage d'un classifieur spécialisé à la scène avec le même algorithme d'apprentissage que le classifieur générique. Cette formalisation est générique puisqu'elle peut être appliquée pour spécialiser tout type de classifieur.

(2) *Stratégies d'observation :* Notre deuxième contribution consiste à proposer deux stratégies d'observation à intégrer dans l'étape de mise à jour de filtre SMC. Une stratégie d'observation sert à sélectionner les vraies propositions et à éviter la distorsion de la base spécialisée avec des échantillons mal étiquetés. Ces stratégies utilisent des informations *a priori* sur la scène cible et des indices contextuels visuels extraits de la vidéo pour attribuer un poids à chaque échantillon retourné par l'étape de prédiction. Les indices visuels proposés n'intègrent pas le

score retourné par le classifieur utilisé pour ne pas faire dériver l'apprentissage du classifieur spécialisé.

(3) *Application de l'approche proposée pour la détection multi-objets :* Nous appliquons l'approche présentée pour spécialiser des détecteurs de piétons et de voitures. Nous avons montré, à travers plusieurs expérimentations sur des scènes de trafic routier difficiles, la généricité de l'approche. Nous avons intégré à cette approche deux stratégies d'observation. Et nous l'avons appliqué à une spécialisation d'un détecteur de type HOG-SVM et également d'un détecteur à base d'apprentissage profond.

4 Publications réalisées dans le cadre de cette thèse

[Maâmatou *et al.*, 2016a] Houda Maâmatou, Thierry Chateau, Sami Gazzah, Yann Goyat, and Najoua Essoukri Ben Amara. Sequential Monte Carlo Filter Based on Multiple Strategies for a Scene Specialization Classifier. In *EURASIP Journal on Image and Video Processing*, EURASIP JIVP 2016, Springer International Publishing.

[Maâmatou *et al.*, 2016b] Houda Maâmatou, Thierry Chateau, Sami Gazzah, Yann Goyat, and Najoua Essoukri Ben Amara. Transductive transfer learning to specialize a generic classifier towards a specific scene. In *International Conference on Computer Vision Theory and Applications*, VISAPP 2016.

[Maamatou *et al.*, 2015] Houda Maamatou, Thierry Chateau, Sami Gazzah, Yann Goyat, Najoua Essoukri Ben Amara. Transfert d'apprentissage par un filtre séquentiel de Monte Carlo : application à la spécialisation d'un détecteur de piétons. In *Journées francophones des jeunes chercheurs en vision par ordinateur*, ORASIS 2015.

[Mhalla *et al.*, 2016] Ala Mhalla, Houda Maâmatou, Thierry Chateau, Sami Gazzah, and Najoua Essoukri Ben Amara. Faster R-CNN Scene Specialization with a Sequential Monte-Carlo Framework. In *Digital Image Computing : Techniques and Applications*, DICTA 2016.

[Mhalla *et al.*, 2017] Ala Mhalla, Thierry Chateau, Houda Maâmatou, Sami Gazzah, and Najoua Essoukri Ben Amara. SMC Faster R-CNN : Toward a Scene Specialized Multi-Object Detector. In *Computer Vision and Image Understanding*, CVIU (en cours de révision), Elsevier.

5 Structure du document

Le manuscrit de la thèse est organisé en trois parties dont chacune comporte deux chapitres. La première présente un état de l'art sur la détection d'objets, sur le transfert d'apprentissage et sur l'analyse des vidéos du trafic routier en vue de justifier les choix fixés dans nos travaux. La seconde partie est dédiée à la description des principales contributions réalisées au cours de la thèse. La troisième partie concerne les expérimentations effectuées pour la validation de l'approche proposée et l'implémentation pour l'intégration dans OD SOFT.

Plus précisément, nous donnons dans le **chapitre 1** un aperçu détaillé sur les méthodes de détection d'objets, sur les types d'apprentissage semi-supervisé et sur le transfert d'apprentissage naturel et artificiel. Ensuite, les différentes motivations et les différents types de transfert automatique d'apprentissage sont exposés. Nous présentons par la suite, les différentes catégories des techniques du transfert d'apprentissage et la différence entre l'apprentissage multi-tâches et le transfert d'apprentissage. A la fin du chapitre, nous décrivons des applications du transfert d'apprentissage dédiées à la détection d'objets.

Le **chapitre 2** définit l'analyse automatique d'une scène du trafic routier et présente les primitives les plus utilisés pour la détection d'objets. Le détecteur choisi pour nos travaux est présenté par la suite. Puis, nous présentons les limites des détecteurs génériques et l'apport de spécialisation via la présentation de certains travaux existants.

Le **chapitre 3** traite la description de l'approche de spécialisation proposée. Nous commençons par donner un bref rappel sur le filtre Séquentiel de Monte Carlo. Ensuite, nous présentons le schéma bloc descriptif de notre solution proposée relative à une itération donnée et nous décrivons le rôle de chaque étape du filtre.

Dans le **chapitre 4**, nous exposons principalement deux stratégies d'observation. Une stratégie d'observation intervient une fois que l'étape de prédiction du filtre SMC a collecté les échantillons de la scène cible, pour juger la pertinence de ces échantillons. Pour cet objectif, nous proposons d'extraire un ensemble d'indices spatio-temporels pour pondérer les échantillons cibles et automatiser la sélection des échantillons associés aux bonnes étiquettes. La première stratégie calcule deux scores et ensuite elle les utilise pour l'affectation d'un poids à un échantillon. La deuxième stratégie classe les points d'intérêt en point mobile (ou point de forme) et statique (ou point de fond). Après, elle calcule un poids à chaque proposition en fonction de la nature et de nombre des points d'intérêt qui se trouvent dans sa *ROI* associée.

Le **chapitre 5** décrit les différents tests effectués et les résultats obtenus. Il commence par un rappel sur le déroulement du processus itératif de la spécialisation. Nous présentons dans la deuxième section les différents détecteurs HOG-SVM génériques. Ensuite, dans la troisième section, nous évaluons notre approche à travers plusieurs expérimentations pour mettre en évidence la performance de notre approche et l'apport de la spécialisation. Nous terminons le chapitre par une démonstration de la généricité de la méthode proposée par une spécialisation d'un détecteur à base d'apprentissage profond.

Dans le **chapitre 6**, nous présentons le logiciel OD SOFT de la société Logiroad tout en précisant sa configuration et son mode de fonctionnement. Nous décrivons également comment nous avons intégré dans OD SOFT un détecteur spécialisé par notre approche. Dans la dernière section du chapitre, nous montrons à travers une comparaison sur quatre base de données l'apport des détecteurs spécialisés par rapport à la méthode de détection Vu-mètre de Logiroad.

Nous terminons le manuscrit par une conclusion générale résumant le bilan des travaux réalisés et quelques perspectives futures à cette thèse.

Première partie

État de l'art

Cette première partie, composée de deux chapitres, expose les techniques les plus présentes dans la littérature relatives à la détection d'objets, à l'apprentissage semi-supervisé, au transfert d'apprentissage et à l'analyse automatique des scènes de vidéo-surveillance du trafic routier.

Le premier chapitre détaille principalement le transfert d'apprentissage artificiel, ses motivations et ses différents types. Il décrit une catégorisation des méthodes de transfert d'apprentissage selon la nature des connaissances transférées et illustre ses différences avec l'apprentissage multi-tâches. Le chapitre se termine par une présentation des applications du transfert d'apprentissage pour la détection d'objets.

Le deuxième chapitre de cette partie donne une définition de l'analyse automatique des scènes de trafic routier. Ensuite, il révèle les descripteurs les plus populaires dans l'application visée par cette thèse, à savoir la vidéo-surveillance de la circulation routière. Le choix d'un détecteur pour nos travaux et sa description sont par la suite présentés. Ce chapitre répond également à la question " Est-il nécessaire de faire une spécialisation du détecteur ? " via la démonstration des limites d'un détecteur générique lorsqu'il est appliqué à une scène particulière et via la description de certains travaux existants de spécialisation. A la fin, le chapitre décrit la courbe ROC comme outil d'évaluation d'un détecteur d'objet.

Chapitre 1

L'apprentissage semi-supervisé et le transfert d'apprentissage

Introduction

Les méthodes d'apprentissage semi-supervisées constituent une catégorie des méthodes d'apprentissage artificiel. Elles utilisent à la fois des données étiquetées et non-étiquetées pour ajuster le fonctionnement de classificateur avec des nouvelles données. Le transfert d'apprentissage est défini comme étant la capacité d'utiliser des connaissances et des compétences apprises dans des activités précédentes pour réaliser une nouvelle activité [Pan et Yang, 2010]. Le transfert d'apprentissage est étudié dans plusieurs domaines comme la psychologie cognitive, la linguistique, l'interaction Homme-machine et l'apprentissage automatique. Nous nous intéressons aux cas du transfert d'apprentissage pour la classification, la régression, le regroupement et la détection d'une catégorie d'objet.

Nous commençons dans la première section par une revue de littérature sur les approches de détection d'objets qui représente l'application de cette thèse. Dans la deuxième section, nous décrivons l'apprentissage semi-supervisé et ses différentes méthodes. Dans la troisième section, nous présentons le transfert d'apprentissage naturel chez un enfant tout en décrivant le passage d'un apprentissage supervisé vers le transfert et l'utilisation des connaissances déjà acquises. Dans la quatrième section nous exposons le transfert d'apprentissage artificiel, ses différentes motivations et ses différents types. Au niveau de la cinquième section, nous présentons les techniques de transfert d'apprentissage en fonction de la nature des connaissances sources transférées. Dans la sixième section, nous expliquons la différence entre l'apprentissage multi-tâches et le transfert d'apprentissage. Au cours de la dernière section, nous présentons les applications de transfert d'apprentissage tout en détaillant celles qui font la détection d'objets.

1 Les différentes approches de détection d'objets

Nous avons décrit, dans l'introduction, la détection d'objets et les challenges associés. Dans cette section, inspirée principalement des travaux de Shantaiya *et al.*[Shantaiya *et al.*, 2013], nous présentons les trois grandes familles d'approches de détection d'objets dans une vidéo. Ces trois familles sont la détection basée sur le mouvement, la détection basée sur les attributs et la détection basée sur l'apprentissage.

1.1 Détection basée sur le mouvement

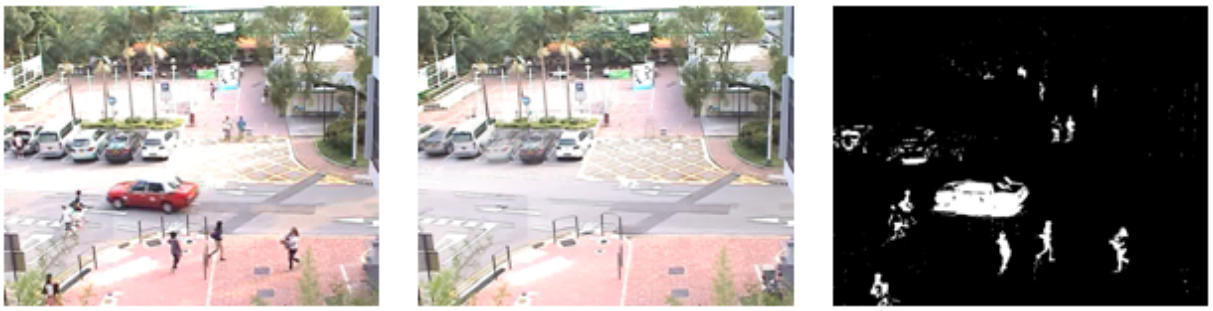
Deux classes représentent principalement les méthodes de détection basées sur le mouvement d'objet à savoir : la soustraction fond-forme et le flot optique.

Soustraction fond-forme

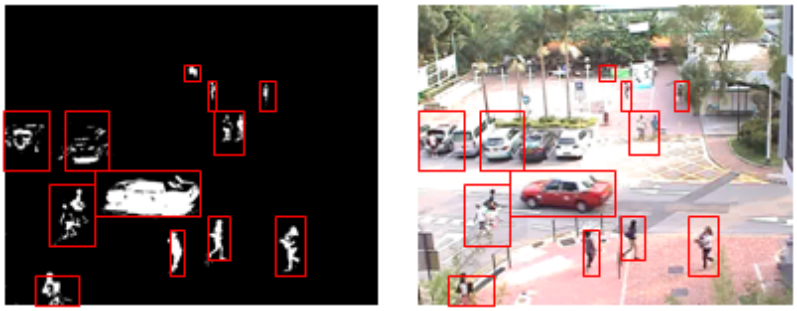
Cette classe de méthodes repose sur la détection du changement temporel au niveau d'un pixel ou un bloc de pixels tout en se basant sur l'hypothèse que le fond est statique et la forme (représentant l'objet) est en mouvement continu. La forme est détectée par la soustraction d'un modèle de fond de toutes les images de la séquence vidéo suivi d'une ou de plusieurs règles permettant la classification d'un pixel en fond ou forme. Une image "modèle de fond" peut être la première image de la vidéo, une image moyenne ou médiane de vidéo calculée globalement ou localement. Le résultat est une segmentation de l'image représentant d'un côté le fond et de l'autre côté les objets mobiles sans aucune sémantique dans la réponse et sans information sur la nature de l'objet détecté. Une étape de détermination de blob et post-traitement est souvent recommandée pour différencier les objets et ajouter du sens à la segmentation brute.

La littérature montre plusieurs algorithmes de soustraction fond-forme qui partagent la même architecture. Cette dernière peut être composée de trois étapes :

1. Calcul d'un modèle de fond
2. Classification des pixels en fond ou en forme
3. Mise à jour du modèle de fond



(a) Image CUHK_Square (b) Modèle de fond (médiane) (c) Résultat de soustraction de fond



(d) filtrage et extraction des boites englobantes (e) Détection des objets

FIGURE 8 – Détection d'objets à base d'une soustraction fond-forme

Lan Wu [Wu, 2008] a présenté une méthode basée sur la différence entre images successives avec seuillage pour la détection et le suivi des joueurs de hockey en estimant les positions et les tailles de manière automatique tout en conservant l'identité des objets dans deux séquences vidéos synchronisées. Néanmoins, sa méthode est incapable de détecter le joueur s'il reste statique pendant un moment. Les joueurs perdent alors leurs identités en cas d'occlutation dans deux points de vues différents. Lo et Velastin [Lo et Velastin, 2001] ont proposé d'utiliser la valeur médiane des n dernières images comme

un modèle de fond. L'inconvénient majeur de ce type de méthode est la nécessité d'un espace mémoire pour le stockage de n valeurs pour chaque pixel de l'image. Ainsi, le filtre médian ne permet pas une description statique rigoureuse. La modélisation de chaque pixel (fond et forme) par une mixture des gaussiennes est introduite par Stauffer et Grimson [Stauffer et Grimson, 1999] en 1999. Border *et al.*[Bodor *et al.*, 2003] ont utilisé la mixture des gaussiennes pour la segmentation fond-forme. Les valeurs des pixels de fond sont modélisées comme une mixture des gaussiennes. A chaque itération, il y aura une évaluation simple pour déterminer les pixels de fond. Les pixels qui ne correspondent pas au fond sont classés forme. Ensuite, les pixels formes sont regroupés par une analyse 2D de composantes connectées. La limite de cette technique est la perturbation rapide par les changements de lumière dans des scènes extérieures à cause du soleil et de la présence des nuages.

Elgammal *et al.*[Elgammal *et al.*, 2000] ont proposé d'exprimer la distribution de fond par un modèle non-paramétrique basé sur un estimateur KDE, Olivier *et al.*[Oliver *et al.*, 2000] ont utilisé une décomposition des valeurs propres pour modéler le fond de la scène, etc. Massimo Piccardi a élaboré dans [Piccardi, 2004] un état de l'art que nous recommandons pour tout lecteur intéressé. Néanmoins, il est à noter que ces méthodes présentent diverses difficultés techniques telles que des fausses détections à cause d'objets mobiles de fond, de la sensibilité à la variation de luminosité (évolution de temps au cours de la journée ou changement brusque de lumière), de l'agitation des branches d'arbres, de la vibration de caméra à cause de vents, et des ombres portées des objets [Toyama *et al.*, 1999].

Le flot optique

Le flot optique est une des méthodes les plus utilisées. C'est l'estimation d'un champ de vecteurs de déplacement qui définit la translation de chaque pixel dans une région entre une image source et une image cible. La vitesse et la direction de chaque pixel de l'image doivent être calculées. Ce type de méthode est très intéressant dans la détection et le suivi d'objets en vidéo avec déplacement de fond ou caméra mobile. La majorité des méthodes d'aujourd'hui sont simialaires à celle de Horn et Schunck [Horn et Schunck, 1981]. Elle est développée en 1981 et elle repose sur deux hypothèses [Sun *et al.*, 2010] :

1. Conservation d'intensité lumineuse : Un pixel conserve son intensité lumineuse au cours de son déplacement. Cette hypothèse s'écrit selon l'équation (1.1) :

$$I(x, y, t) = I(x + \delta x, y + \delta y, t + \delta t), \quad (x, y) \in \Omega, t \in \Re \tag{1.1}$$

   Il est à noter que x et y dépendent du temps et devraient être écrites $x(t)$ et $y(t)$, mais pour des raisons de clarté sont écrites x et y. De plus, δx et δy sont les coordonnées cartésiennes du vecteur associé au pixel et décrivant son déplacement au cours de l'intervalle de temps δt.

2. Cohérence spatiale : Les pixels voisins appartiennent au même objet et se déplacent de la même manière.

Horn et Schunck [Horn et Schunck, 1981] ainsi que Lucas et Kanade [Lucas *et al.*, 1981] ont adopté une version approchée de la condition de conservation d'intensité et ceci en dérivant l'intensité $I(x(t), y(t), t)$ par rapport au temps :

$$\frac{\partial_I}{\partial_x}.\frac{dx}{dt} + \frac{\partial_I}{\partial_y}.\frac{dy}{dt} + \frac{\partial_I}{\partial_t} = 0 \tag{1.2}$$

Ainsi à partir de l'équation(1.2), la contrainte fondamentale du flot optique est définit par l'équation (1.3) :

$$\partial_x I.u + \partial_y I.v + \partial_t I = 0 \tag{1.3}$$

La méthode Lucas et Kanade [Lucas *et al.*, 1981] est caractérisée par un temps de calcul avantageux mais souffre du manque de précision. Par contre, la méthode de Horn et Schunck [Horn et Schunck, 1981] a l'avantage de fournir un système linéaire lors de passage aux equations d'Euler-Lagrange. Mais, elle a un désavantage de lisser la solution de manière uniforme alors que le calcul de flot optique passe par restitution des discontinuités de mouvements dans plusieurs applications. Ces deux méthodes sont valides uniquement dans le cas de petits mouvements et elles sont sensibles au bruit. Pour corriger le problème de robustesse aux données aberrantes dans les méthodes précedentes, Michael Julian Black [Black, 1992] a remplacé le critère quadratique par une fonction qui pénalise moins les fortes discontinuités. Black et Anandan [Black et Anandan, 1996] ont introduit un modèle robuste pour gérer les données aberrantes, engendrées par les réflexions, les occultations et les extrémités de mouvement. Mais, ils n'ont pas modélisé les statistiques de l'erreur de conservation d'intensité. Nombreux sont les travaux qui ont étendu l'hypothèse de conservation de luminosité soit en rendant plus admissible physiquement [Haussecker et Fleet, 2001] soit en appliquant un pré-filtrage linéaire ou non-linéaire sur des images [Toth *et al.*, 2000],[Sun *et al.*, 2008]. Comme le cas de soustraction fond-forme, la notion d'objet n'existe pas dans les techniques qui calculent le flot optique et leurs contraintes restent non-valides dans beaucoup d'applications. En effet, chaque pixel peut bouger indépendamment de ses voisins et la luminance n'est pas forcément conservée entre deux images successives. La luminosité d'une scène peut changer grâce à un déplacement de nuage et un objet peut être occulté [Chesnais, 2013].

Les approches basées sur la soustraction fond-forme ou sur le flot optique détectent les objets en mouvement sans aucune information sur leur nature. Elles sont caractérisées par un temps de traitement plus ou moins acceptable dans beaucoup d'applications mais elles sont sensibles au bruit et aux variations des conditions d'éclairement.

(a) Image 1 MIT Trafic dataset (b) Image 5 (c) Image 10

FIGURE 9 – Détection de mouvements par flot optique avec la méthode Lukas-Kanade sur des images de MIT Traffic dataset [Lucas *et al.*, 1981].

1.2 Détection basée sur les attributs

La détection d'objets basée sur les attributs repose sur la caractérisation de l'image en un ensemble de caractéristique (ou primitives) importants tels que la couleur, le modèle, la forme, la taille, etc. L'objet d'intérêt qu'on cherche à détecter est modélisé en fonction de ce jeu de primitives. Par exemple, la couleur d'un objet est une information facile à extraire et elle est relativement constante vis-à-vis du changement des points de vue. Zhenjun *et al.*[Han *et al.*, 2008] ont utilisé un ensemble de primitives formé de la couleur, l'histogramme d'orientations de contours et le descripteur SIFT. La limite de leur méthode est qu'elle est valide uniquement si le fond est unique. Saravanakumar *et al.*[Saravanakumar *et al.*, 2011] ont représenté l'objet en fonction des propriétés de l'espace couleur HSV pour la détection

et le suivi des personnes. Une version adaptative de l'algorithme K-means est appliquée pour regrouper les valeurs des couleurs des centroïdes d'objets et du transfert de leurs coordonnées à l'image suivante pour le suivi.

Dans certains cas la détection d'objets peut être une sorte de mise en correspondance des caractéristiques entre un modèle et les images de la séquence. Liu *et al.*[Liu *et al.*, 2011] ont proposé des modèles hybrides pour la détection d'objets. Les modèles sont composés de plusieurs types de caractéristiques tels que le squelette/contours, les régions de texture, les régions plates homogènes. La limitation de cette dernière est que le modèle hybride doit être mis à jour de manière adaptative en ajustant les coefficients des caractéristiques ou en substituant les caractéristiques anciennes avec celles récemment trouvées. D'autres travaux utilisent la technique de mise en correspondance de modèle fixe par corrélation. Cette technique utilise la position du pic normalisé de corrélation croisée entre le modèle d'objet et une image pour bien localiser l'objet. Elle est non-sensible au bruit et aux effets de luminosité, mais souffre d'une complexité de calcul élevée. Pan *et al.*[Pan *et al.*, 2008a] ont proposé un algorithme dit CAPOA (Content-Adaptive Progressive Occlusion Analysis) qui analyse la situation d'occultation dans une région d'intérêt donnée et génère un masque de modèle correspondant. Le suivi d'une même instance d'objet est assez difficile avec cet algorithme.

Les approches basées sur les connaissances peuvent ne pas fournir une détection efficace pour le multi-objets. Elles donnent également des performances dégradées si le fond et l'objet ont une couleur similaire ou dans un cas d'occultation. L'utilisation d'un modèle fixe est une limite à la détection multi-échelles et la mise en correspondance d'un modèle déformable est difficile.

1.3 Détection basée sur l'apprentissage

La détection basée sur l'apprentissage consiste à apprendre l'apparence de l'objet à l'aide des algorithmes d'apprentissage et de chercher par la suite l'objet directement dans l'image. Pour ce faire, il y a une génération d'un détecteur d'objet. C'est une fabrication d'un modèle discriminant étant donné un ensemble d'images d'apprentissage. Ce détecteur est capable de connaitre l'apparence ou l'allure générale de l'objet cherché tout en utilisant un classifieur [Chesnais, 2013]. La FIGURE 10 inspirée de [Chesnais, 2013] présente les différents blocs d'un système de détection basé sur l'apprentissage supervisé que nous décrivons brièvement par la suite.

Base d'apprentissage : Une base de données d'apprentissage (ou ensemble d'apprentissage) est un ensemble d'images étiquetées appartenant au moins aux deux classes : objet d'intérêt et fond ou "non-objet" d'intérêt. On peut avoir un ou plusieurs objets d'intérêts mais dans nos travaux nous nous intéressons aux bases de données qui permettent de distinguer un objet contre tout le reste. Cette base de données doit présenter une grande variété d'échelles, de points de vue, d'illumination et de résolutions d'images. Nous appelons parfois base d'apprentissage l'ensemble des descripteurs calculés à partir des images étiquetées [Chesnais, 2013].

Descripteur : Un descripteur est un vecteur de primitives qui décrit l'information brute d'une image sous une forme exploitable. Par abus de langage, le terme descripteur sert à désigner suivant les cas le type de descripteur, le vecteur descripteur ou la primitive de descripteur [Chesnais, 2013].

Apprentissage : L'apprentissage est un algorithme qui génère un modèle à partir d'une base de données. Ce modèle fait la correspondance des entrées aux sorties souhaitées. Nous avons présenté l'apprentissage dans le paragraphe 1.1 de l'introduction (page 1).

Classifieur : Le Classifieur est une fonction qui attribue à un descripteur une étiquette. Il est constitué par le modèle d'objet appris par l'algorithme d'apprentissage et le module de comparaison. Au cours de la comparaison (ou classification), il y a recours aux techniques d'évaluation de relation entre le modèle et le descripteur de l'image étudiée pour donner une mesure de similarité (ressemblance). Si la mesure est suffisamment élevée, le classifieur attribue l'étiquette d'objet recherché, sinon le descripteur ne représente pas l'objet d'intérêt. Souvent en détection d'objet, la classe d'objet recherché est représenté par l'étiquette "1" et le fond est représenté par "-1".

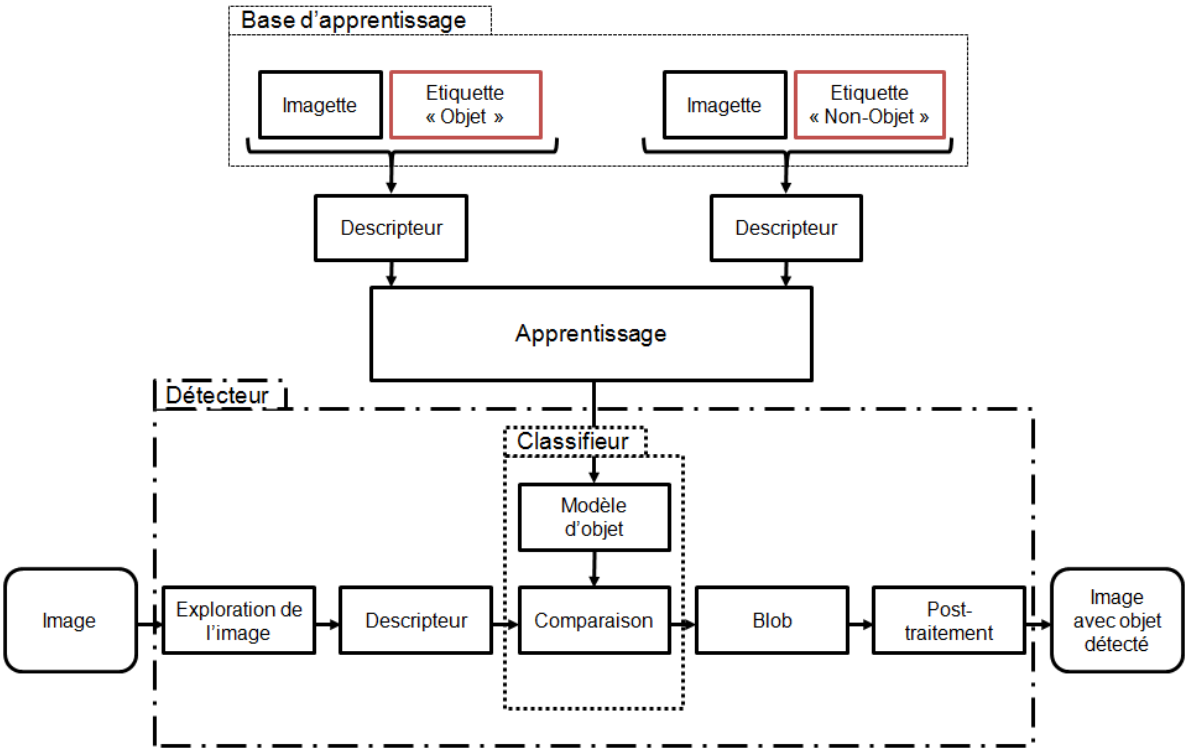


FIGURE 10 – Synoptique d'un système de détection d'objet basé sur l'apprentissage supervisé

Détecteur : Le détecteur est un algorithme constitué par plusieurs modules : exploration d'image, descripteur, classifieur, blob et post-traitements :

- L'exploration d'image est une fonction qui fait le parcours de l'image et cherche les zones qui peuvent contenir l'objet cherché. Le balayage par fenêtre fixe est une des méthodes d'exploration d'image les plus connues. Récemment, le module d'exploration d'image est défini comme une entité externe de détecteur, connu par le nom module de propositions d'objets telle que la méthode de recherche sélective [Uijlings *et al.*, 2013], la MCG [Arbeláez *et al.*, 2014],...
- Le blob est la transformation du résultat de module de comparaison en rectangle englobant la zone qui est classée comme objet recherché.
- Les post-traitements : Souvent il faut une étape de regroupement ou de filtrage des réponses retournées par le classifieur pour représenter une réponse plus lisible et acceptable. Par exemple, on applique un regroupement sur les rectangles englobants autour d'un objet pour donner un seul rectangle englobant l'objet.

Plusieurs articles présentent des méthodes de détection d'objet basées sur l'apprentissage [Han *et al.*, 2006, Viola et Jones, 2001a, Dalal et Triggs, 2005, Alvarez *et al.*, 2009, Danescu *et al.*, 2011, Sivaraman et Trivedi, 2013, Lin *et al.*, 2012, Yuan *et al.*, 2011, Sun et Watada, 2015]. Le principal avantage de cette catégorie par rapport aux précédentes est que le résultat de détection est plus sémantique étant donné que le système est entrainé pour détecter uniquement l'objet d'intérêt. Dans cette thèse nous nous intéresserons plus particulièrement à cette classe de méthodes étant donné qu'elle permet d'éviter ou au moins de réduire certains problèmes liés aux deux autres classes basées sur le mouvement et les attributs.

2 L'apprentissage semi-supervisé et ses différentes méthodes

Selon Chapelle [Chapelle *et al.*, 2006] et sachant $\chi = \{x_1, ..., x_n\}$, un ensemble d'échantillons de taille n, avec x_i représente le vecteur caractéristique de l'échantillon i, l'apprentissage semi-supervisé est un cas où l'ensemble de données $\chi = \{x_1, ..., x_n\}$ est divisé en deux parties :

— Les données $\chi_l = (x_1, \ldots, x_l)$ pour lesquelles les étiquettes $Y_l = (y_1, \ldots, y_l)$ sont fournies.

— Les données $\chi_u = (x_{(l+1)}, \ldots, x_{(l+u)})$ dont les étiquettes ne sont pas connues.

En plus des données non étiquetées, l'algorithme prend en considération certaines informations de supervision mais pas nécessairement pour tous les exemples.

De cette définition, l'apprentissage semi-supervisé (*en anglais Semi-Supervised Learning (SSL)*) se situe à mi-chemin entre l'apprentissage supervisé et non supervisé. La première idée d'utiliser des données non-étiquetées dans la classification date depuis l'année 1965 [Scudder, 1965]. L'objectif de cette utilisation est de générer un classifieur meilleur que celui entrainé uniquement par des données étiquetées. Puisque ces données sont très limitées par rapport aux données non-étiquetées qui peuvent être disponibles.

La FIGURE 11 présente une taxonomie des différents types d'apprentissage. Nous décrirons dans la suite, les méthodes d'apprentissage semi-supervisé les plus connues.

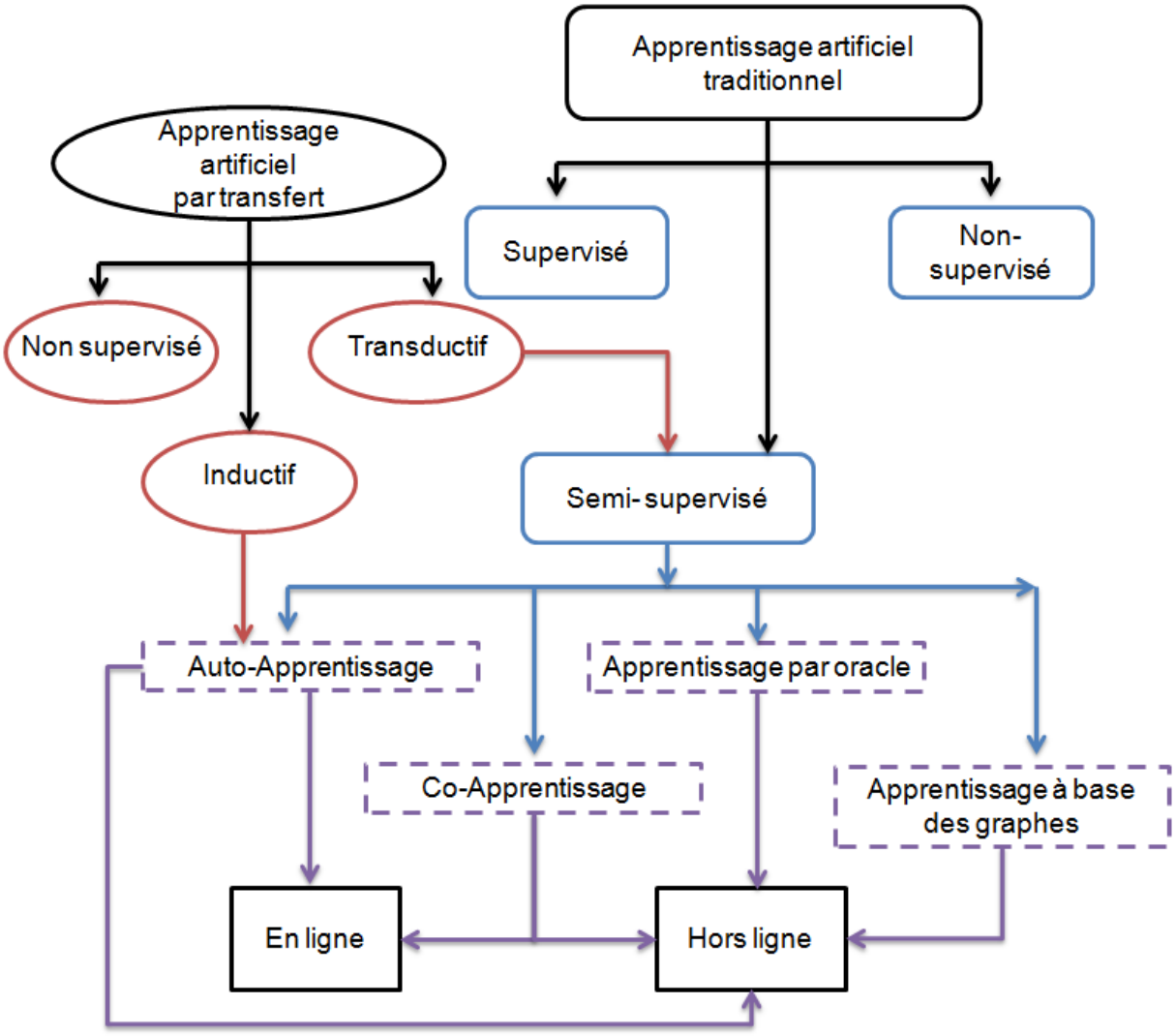


FIGURE 11 – Taxonomie des différents types d'apprentissage

2.1 L'auto-apprentissage

L'auto-apprentissage (*connu en anglais par self-learning ou self-training ou self-labeling*), est une méthode qui utilise sa propre fonction de décision pour se mettre à jour et améliorer ses performances de manière itérative en cours de fonctionnement. Initialement, cette catégorie de méthodes se base sur un classifieur entraîné avec des données labellisées de manière supervisée. Ensuite, un ensemble de données non labellisées est récupéré au fur et à mesure des calculs durant un intervalle de temps régulier. Cet ensemble sera étiqueté par le classifieur initial et sera utilisé pour le ré-entraînement ou la mise à jour de ce dernier.

Le problème de ce type d'apprentissage est de définir l'intervalle de temps suffisant pour la collecte de données et de configurer les paramètres de la fonction de décision. Si cette dernière est assez sélective alors seulement les données qui sont très semblables à celles étiquetées seront utilisées pour l'apprentissage de l'itération suivante, alors qu'elles ne présentent pas d'informations importantes de variabilité. Dans le cas contraire, il y a un risque d'introduire des données erronées qui dégradent la performance du système dans le temps. La littérature cite plusieurs travaux qui utilisent ce type de méthode d'apprentissage [Scudder, 1965, Agrawala, 1970, Rosenberg *et al.*, 2005, Rattani, 2010].

2.2 Le co-apprentissage

Le co-apprentissage ou co-training est une extension de l'auto-apprentissage. Ce type d'apprentissage utilise deux sources différentes d'informations pour apprendre les règles initiales et corréler leurs résultats afin d'améliorer l'apprentissage par des données non-labellisées. Nous aurons deux classifieurs entraînés sur les données étiquetées. Un ensemble de données non étiquetées est collecté tout au long de l'utilisation du système. Les données étiquetées avec forte confiance par l'un ou l'autre des classifieurs sont ajoutées aux données d'apprentissage pour le ré-entrainement des deux classifieurs. La FIGURE 12 montre que le co-apprentissage entre deux détecteurs de voitures proposé par Levin *et al.*[Levin *et al.*, 2003] donne deux détecteurs plus performants que les deux initiaux.

Ce type d'apprentissage corrige la limite liée à l'ajout des exemples relativement ressemblant à ceux qui sont connus au départ. L'utilisation d'un deuxième classifieur introduit des données à forte variabilité par rapport à ceux retenues par le premier classifieur et vice-versa [Blum et Mitchell, 1998, Levin *et al.*, 2003, Rattani, 2010]. Plus les jeux de caractéristiques sur lesquels les classifieurs sont entrainés sont indépendants, meilleures sont les performances de co-apprentissage et plus les classifieurs encodent de variabilité intra-classe. Néanmoins, la difficulté est comment prouver l'indépendance des deux sources d'informations sachant que Dalal *et al.*[Dalal *et al.*, 2006] ont montré que le détecteur de piétons à base d'apparence et celui à base de mouvement sont fortement corrélés entre eux.

2.3 L'apprentissage par oracle

Les méthodes d'apprentissage par oracle utilisent une entité externe au système dite « Oracle » pour étiqueter les données non-étiquetées qui seront utilisées pour le ré-entraînement du classifieur final. Il est essentiel que l'oracle et le classifieur final soient indépendants pour échapper au phénomène de dérivation. Un oracle est caractérisé par sa robustesse et sa capacité de précision qui détermine la performance du système final. Il peut être construit à l'aide d'un seul algorithme comme le cas de Nair et Clark [Nair et Clark, 2004] qui ont présenté un oracle à base d'un algorithme de soustraction fond-forme. Il peut egalement combiner et/ou fusionner deux où plusieurs algorithmes. Nous citons les travaux de Chesnais *et al.*[Chesnais, 2013, Chesnais *et al.*, 2012] qui ont construit un oracle par trois classifieurs indépendants entrainé chacun sur un signal : apparence, extraction fond/forme et flot optique. Cet oracle est utilisé par la suite pour créer une base contextualisée (base contenant des échantillons issus suite à des observations de la même scène et enregistrées par la même caméra). Ensuite, ils créent le détecteur final de piétons et l'entraînent sur la base fournie par l'oracle.

(a) (b)

Résultats de détection avant co-apprentissage

(c) (d)

Résultats de détection après co-apprentissage

FIGURE 12 – Effet de co-apprentissage sur la détection. Comparaison des résultats de détection de deux détecteurs (a)&(c) détecteur entrainé sur le signal d'apparence des véhicules et (b)&(d) détecteur entrainé sur la soustraction de fond [Levin *et al.*, 2003]

La sélection des échantillons en se basant uniquement sur le résultat de soustraction fond-forme, rend le classifieur de Nair et Clark [Nair et Clark, 2004] très sensible au risque de dérivation. Certains objets statiques ou possédant une apparence similaire au fond peuvent être classés comme échantillons négatifs et d'autres objets de fond mobiles sont étiquetés comme objet d'intérêt. De plus, Chesnais *et al.*[Chesnais, 2013, Chesnais *et al.*, 2012] ont émis l'hypothèse que les trois signaux sont indépendants ce qui ne peut pas être toujours vrai dans les applications du monde réel.

2.4 L'apprentissage à base des graphes

L'apprentissage à base des graphes se reposent sur l'hypothèse que les exemples non étiquetés doivent recevoir les mêmes étiquettes que leurs proches voisins. Les méthodes d'apprentissage à base des graphes permettent d'introduire une variabilité intra-classe satisfaisante. Dans un graphe, un nœud représente un échantillon (étiqueté on non-étiqueté) et chaque arête reliant deux nœuds présente le poids d'une similarité entre eux [Chapelle *et al.*, 2006]. Le principe de fonctionnement est de propager les étiquettes depuis les nœuds étiquetés vers ceux non-étiquetés : les étiquettes se propagent par les arêtes possédant un poids élevé et s'accumulent lorsque les arêtes ont un poids faible. Blum et Chawla [Blum et Chawla, 2001] ont proposé une méthode à base des graphes qui combine les données étiquetées et non-étiquetées pour entraîner un classifieur. Rattani [Rattani, 2010] a utilisé ce type de méthodes pour adapter des systèmes biométriques aux nouvelles données disponibles. Rattani [Rattani, 2010] a montré que ces nouvelles données permettent d'améliorer la mise à jour de modèles en encodant plus de variabilité intra-classe.

2.5 L'apprentissage semi-supervisé par transfert transductif

Toutes les méthodes d'apprentissage semi-supervisé sont des méthodes d'apprentissage par transfert transductif ou dites aussi méthodes de transfert d'apprentissage transductif lorsqu'elles ne se basent pas sur l'hypothèse que les données d'apprentissage et les données futures (données de test)

sont dans le même espace de caractéristiques et ont la même distribution. Cette dernière est une hypothèse admise par les algorithmes d'apprentissage traditionnels, mais elle ne peut pas être retenue dans les applications du monde réel [Pan et Yang, 2010].

L'apprentissage par transfert est une catégorie d'apprentissage qui vise à exploiter et à appliquer des connaissances déjà acquises dans des activités antérieures pour améliorer le système d'apprentissage d'une nouvelle activité, ou de la même activité mais dans un nouveau domaine même si l'hypothèse faite par les algorithmes d'apprentissage traditionnel n'est pas valide [Pan et Yang, 2010].

Enfin, il faut noter que les algorithmes d'apprentissage traditionnel ou par transfert peuvent être également classés en deux classes :

— Apprentissage en ligne : Un algorithme d'apprentissage en ligne prend en considération les échantillons un par un (ou partie par partie) tout au long du processus de classification et met à jour le classifieur de manière séquentielle. Cette classe d'algorithme ne peut pas créer un bon classifieur parce que l'algorithme traite à chaque fois uniquement une partie de problème. Néanmoins dans certaines applications, cette classe est couramment utilisée parce qu'elle est capable de traiter une grande quantité de données et de s'adapter à des nouvelles conditions [Rattani, 2010, Chesnais, 2013].
— Apprentissage hors ligne : Pour un algorithme d'apprentissage hors ligne, toutes les données sont collectées avant la phase d'apprentissage ce qui permet d'avoir une vision globale du problème à traiter sous réserve d'avoir l'espace mémoire nécessaire pour stocker toutes les données. Néanmoins, la mise à jour de classifieur pour s'adapter au changement des données n'est possible que par une nouvelle exécution de toute la procédure [Rattani, 2010, Chesnais, 2013].

Nous nous sommes principalement intéressés dans ces travaux de thèse aux méthodes de transfert d'apprentissage transductif (connu en anglais : *Transductive Transfer Learning*) hors ligne. La suite de ce chapitre donne une description détaillée sur le transfert d'apprentissage, son utilité et ses différents types.

3 Le transfert d'apprentissage naturel

L'enfant commence à apprendre petit à petit à marcher et parler, puis à lire et écrire. Cet apprentissage au départ est supervisé par les parents et les professeurs qui récompensent le petit enfant en fonction du résultat fourni ou de l'action prise. Une récompense est positive si le résultat est correct et une récompense est négative dans le cas contraire. Avec le temps, l'être humain devient capable de transférer ses connaissances acquises antérieurement afin de résoudre une nouvelle tâche dans les nouvelles conditions qu'il rencontre. Par exemple, apprendre à reconnaître des pommes peut aider à reconnaitre des poires ; ainsi apprendre à jouer avec des jeux électroniques permet de faciliter l'apprentissage du piano : c'est le transfert d'apprentissage naturel.

Selon Perkins *et al.*[Perkins *et al.*, 1992], le transfert d'apprentissage se produit lorsqu'un apprentissage dans un contexte peut améliorer (transfert d'apprentissage positif) ou affaiblir (transfert d'apprentissage négatif) des performances reliées dans un autre contexte. Le transfert inclut le transfert proche (contextes et performances étroitement liées) et le transfert différent (contextes et performances assez différent). Le transfert des connaissances repose sur une évaluation du degré de similarité entre les contextes source et cible et leurs données associées. La similarité a une définition subjective tout en dépendant à la fois des capacités de l'apprenant et des autres notions telles que l'abstraction et la généralisation. L'identification des liens de similarité entre les deux domaines (source et cible) est une étape essentielle dans le processus de transfert.

4 Le transfert d'apprentissage artificiel

Plusieurs méthodes d'apprentissage automatique traditionnel fonctionnent sous une hypothèse commune : les données d'apprentissage et de test sont issues du même espace de caractéristiques et partagent la même distribution de répartition d'échantillons. Si la distribution est différente, il faudra reprendre l'apprentissage à partir de zéro tout en collectant des nouvelles données du domaine cible et en traitant les tâches de manière isolée. Cependant, dans de nombreuses applications du monde réel, la collecte des nouvelles données est côuteuse et parfois elle est difficile voire impossible [Pan et Yang, 2010].

Le transfert d'apprentissage tente de changer cette situation en développant des méthodes pour transférer des connaissances et compétences apprises dans une ou plusieurs tâches sources afin de l'utiliser pour améliorer une tâche cible avec prise en considération des liens de similarité entre ces tâches.

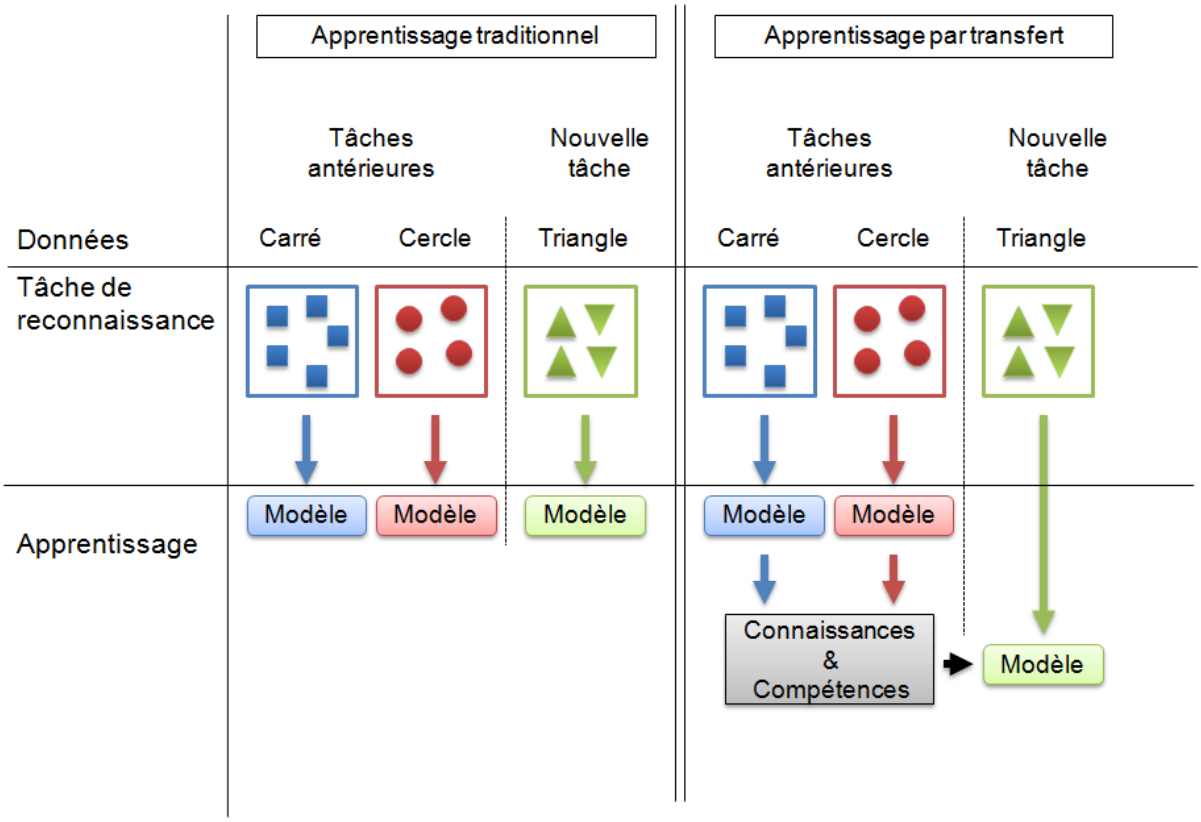


FIGURE 13 – Différences entre l'apprentissage traditionnel et l'apprentissage par transfert [Pan et Yang, 2010]

La FIGURE 13 montre la différence entre l'apprentissage traditionnel et le transfert d'apprentissage. L'apprentissage traditionnel commence à partir de zéro le processus d'apprentissage à toute nouvelle tâche indépendamment des autres tâches alors que les techniques de transfert d'apprentissage utilisent les connaissances acquises dans des tâches précédentes lors de l'apprentissage d'une nouvelle tâche.

En particulier, donner la capacité de reconnaissance visuelle par transfert d'apprentissage à des machines peut améliorer infiniment le domaine de la vision par ordinateur puisque le besoin d'une détection robuste de nombreuses catégories d'objet devient de plus en plus indispensable avec la génération des voitures et des robots automatiques destinés à être une partie de notre vie quotidienne.

Le processus de transfert commence par un besoin d'apprendre une tâche cible dans un contexte ou domaine cible tout en sachant qu'un ensemble de tâches sources dans un domaine source est disponible et que l'on a un ensemble de relations qui sont crées à base de similarité ou ressemblance entre les problèmes source et cible. Généralement les relations de similarité sont établies entre les données sources et les données cibles ou solutions [Aytar, 2014]. Définir la tâche cible, les tâches sources disponibles qui peuvent être bénéfiques à l'amélioration de l'apprentissage de la tâche cible et les relations existantes entre les tâches sources et la tâche cible est la réponse à la question "**D'où transférer ?**"

Une fois que les problèmes sources, le problème cible et les liens de similarité sont bien précisés, l'étape suivante est de décider quoi transférer ? comment ? et quand ? La question "**Quoi transférer ?**" détermine le type de connaissances à transférer du domaine source vers le domaine cible. Ces connaissances peuvent être la solution (ou le modèle d'apprentissage et ses paramètres) ou les données (les échantillons d'apprentissages). La question "**Comment transférer ?**" détermine la nature de transfert c'est à dire si les connaissances transférées seront utilisées telles quelles sont ou bien si elles doivent subir des transformations pour s'adapter aux nouvelles conditions. Elle définit également la manière d'utilisation de ces connaissances lors de la phase d'apprentissage de la nouvelle tâche. La question "**Quand transférer ?**" doit évaluer dans quelle situation le transfert peut être avantageux et dans quel cas il est préférable de ne pas transférer. Cette question cherche à éviter tout cas de transfert négatif en déterminant la quantité de transfert à partir des sources définies. Si les tâches sources ne sont pas similaires à la cible et/ou s'il y a une quantité de données suffisante pour l'apprentissage de tâche cible, le transfert peut engendrer des effets négatifs au lieu d'améliorer le processus d'apprentissage [Aytar, 2014].

4.1 Motivations du transfert d'apprentissage

Le but du transfert d'apprentissage est l'amélioration de l'apprentissage d'une tâche cible en ramenant des connaissances apprises au sujet d'autres tâches sources. Le transfert possède plusieurs avantages, notamment la réduction ou la suppression de l'effort indispensable à l'annotation d'une grande quantité de données. La possibilité de diminuer le nombre d'exemples nécessaires à l'apprentissage d'une nouvelle catégorie, ainsi que la possibilité d'éviter la reprise à partir de zéro du processus d'apprentissage d'une même activité mais dans des nouvelles conditions. La FIGURE 14 décrit trois niveaux d'amélioration de performance d'un apprentissage par transfert (démarrage supérieur, pente supérieure, asymptote supérieure) en comparant avec une méthode d'apprentissage de la tâche cible sans transfert.

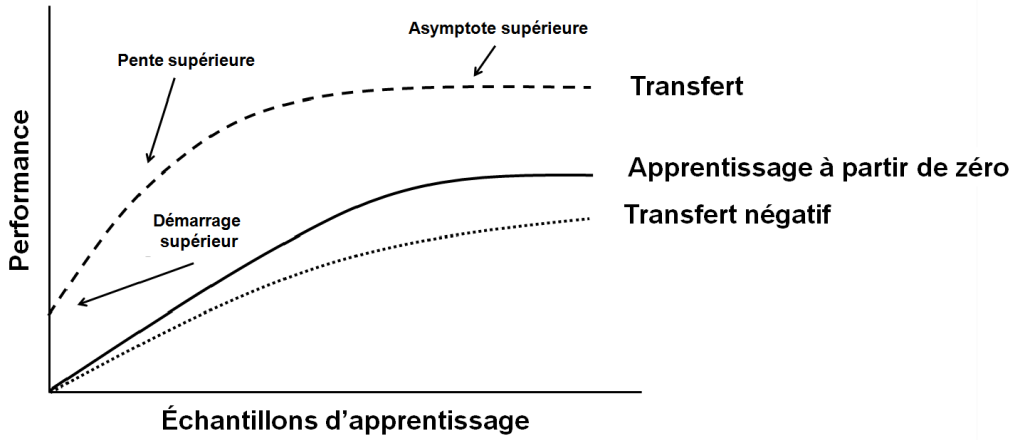


FIGURE 14 – Les avantages de transfert d'apprentissage[Tommasi, 2013, Aytar, 2014]

Démarrage supérieur (Higher start) : Même avec des données cibles limitées, Les méthodes de transfert d'apprentissages démarrent avec des performances meilleures que celles de l'apprentissage à partir de zéro [Tommasi, 2013, Aytar, 2014].

Pente supérieure (Higher slope) : Les performances s'améliorent plus rapidement par l'ajout des instances sources supplémentaires dans le processus d'apprentissage. Cela indique que le temps nécessaire pour apprendre pleinement la tâche cible en exploitant les connaissances transférées, est faible par rapport à ce qui est nécessaire dans le cas de l'apprentissage à partir de zéro [Tommasi, 2013, Aytar, 2014].

Asymptote supérieure (Higher asymptote) : Les performances finales de l'approche de transfert sont nettement supérieures aux performances de l'apprentissage effectué avec les données cibles seulement [Tommasi, 2013, Aytar, 2014].

Si le transfert d'apprentissage dégrade les performances alors **un transfert négatif (negative transfer)** s'est produit. Pour une tâche donnée, l'efficacité de toute méthode de transfert dépend de la tâche source et de la manière dont elle est liée à la tâche cible. Si la relation est forte et le transfert est possible, les performances dans la tâche cible peuvent être améliorées de manière significative par le transfert. Cependant, si les tâches ne sont pas suffisamment liées ou si la relation n'est pas bien exploitée par la méthode de transfert, les performances ne s'améliorent pas et elles peuvent même se dégrader [Torrey et Shavlik, 2009, Tommasi, 2013].

4.2 Différents types de transfert d'apprentissage

Selon Pan et Yang [Pan et Yang, 2010], Il existe trois types de transfert d'apprentissage en se basant sur les différentes relations entre les tâches et les domaines sources et cibles. Le Tableau 1 page 26 résume les différents liens des types de transfert d'apprentissage et les compare par rapport l'apprentissage traditionnel.

1. **Le transfert d'apprentissage inductif :** Pour ce type de transfert, les domaines source et cible peuvent être similaires ou non, mais les tâches sont différentes. Quelques données cibles annotées sont nécessaires pour produire un modèle prédictif à utiliser dans le domaine cible. Deux cas se présentent :
   - Un transfert d'apprentissage inductif avec données sources annotées : L'intérêt ici de transferer des connaissances sources est d'atteindre une performance élevée dans la réalisation de la tâche cible. Ce type de transfert se rapproche de la configuration d'apprentissage multi-tâches à la différence que l'apprentissage multi-tâches apprend les tâches source et cible simultanément [Pan et Yang, 2010].
   - Un transfert d'apprentissage inductif sans données sources annotées : C'est une configuration similaire à un cas d'auto-apprentissage comme le présente Rania *et al.* [Raina *et al.*, 2007]. C'est une situation où les espaces des étiquettes source et cible sont différents et les connaissances du domaine source ne peuvent pas être utilisées directement [Pan et Yang, 2010].

   Nous citons à titre d'exemple, l'apprentissage d'un détecteur de cheval par le transfert de connaissances à partir d'un détecteur de vache, l'apprentissage d'un détecteur de vélo à partir d'un détecteur de moto [Aytar, 2014],...
2. **Le transfert d'apprentissage transductif :** C'est un cas de transfert où uniquement des données non-annotées sont disponibles dans le domaine cible. La distribution conjointe des échantillons et des classes sources est reliée mais différente de la distribution cible où les classes cibles sont inconnues [Farajidavar *et al.*, 2014]. Les tâches source et cible sont les mêmes mais les domaines sont différents. Deux cas se présentent selon les situations des domaines source et cible :
   - Les espaces de caractéristiques entre les domaines source et cible sont différents.

— Les espaces de caractéristiques entre les domaines source et cible sont similaires, mais les distributions de probabilités marginales sont différentes.

Dans ce type de transfert, nous trouvons l'adaptation d'un détecteur de visage à des photos spécifiques pour une bonne manipulation des nouvelles conditions de domaine [Jain *et al.*, 2011], le transfert pour la classification de texte [Daume III et Marcu, 2006], l'adaptation d'un détecteur générique de piétons à une nouvelle scène de trafic routier [Wang et Wang, 2011],...

3. **Le transfert d'apprentissage non-supervisé :** Comme le cas de transfert d'apprentissage inductif, les tâches source et cible sont différentes et les domaines peuvent être similaires ou non. Mais, dans ce type d'apprentissage il y a aucune donnée étiquetée ni dans le domaine source ni dans le domaine cible. La tâche cible est souvent un problème non supervisé comme le regroupement, la réduction de dimension ou l'estimation de densité.

   A titre d'exemple, nous citons le travail de Dai *et al.*[Dai *et al.*, 2008] qui présente une approche de transfert non-supervisé pour le regroupement d'un ensemble de données dans le domaine cible en exploitant une grande quantité de données non-étiquetées disponible dans le domaine source mais en apprenant un espace de caractéristique commun à travers les domaines.

Tableau 1 – Les différents types de transfert d'apprentissage

<table>
<tr><td colspan="2" rowspan="2"></td><td colspan="2">Domaines source et cible
sont identiques ?</td></tr>
<tr><td>Oui</td><td>Non</td></tr>
<tr><td rowspan="2">Tâches source et cible
sont identiques ?</td><td>Oui</td><td>App. traditionnel</td><td>Trans. app. transductif</td></tr>
<tr><td>Non</td><td colspan="2">Trans. app. inductif / Trans. app. non-supervisé</td></tr>
</table>

5 La catégorisation des méthodes du transfert d'apprentissage

Une méthode de transfert d'apprentissage est la réponse à la question "Comment faire le transfert d'apprentissage ?". Elle prend en considération les connaissances sources transférées pour résoudre le problème ou la tâche cible.

Ces méthodes sont catégorisées en fonction des connaissances transférées.

5.1 Méthodes de transfert d'exemples

Ces méthodes se concentrent sur le transfert d'exemples sources qui peuvent être réutilisées en résolvant la tâche cible. Cependant, les données sources peuvent ne pas être utilisables dans leurs formes brutes et ne pas être toutes utiles mais certains exemples peuvent renforcer le processus d'apprentissage cible suite à une fonction de pondération. Malgré l'utilisation d'exemples sources et cibles, le transfert d'exemples apprend uniquement la tâche cible.

Il existe plusieurs méthodes de transfert d'exemples qui sont décrites pour des applications d'intelligence artificielle et de vision par ordinateur.

Dai *et al.*[Dai *et al.*, 2007] présentent une extension "TrAdaBoost" de l'algorithme de dopage basique "Adaboost". Elle réduit les poids des instances qui sont mal prédites afin de diminuer leur effet indésirable le plus possible sur le processus d'apprentissage. TrAdaBoost permet la construction d'un modèle de classification de bonne qualité tout en utilisant des données de distributions et de quantités différentes : une quantité réduite de données étiquetées d'une distribution cible qui est généralement insuffisante pour l'apprentissage d'un bon classifieur et une grande quantité de données d'une autre distribution source. A chaque itération, si une instance est mal prédite alors l'algorithme réduit son poids d'apprentissage pour atténuer son effet aux itérations suivantes. De cette façon, les exemples

qui sont non similaires avec les nouvelles données affectent moins le processus d'apprentissage d'une itération à une autre. Néanmoins, les anciennes instances qui sont en accord avec les données récentes aident l'algorithme à mieux entrainer le classifieur.

Jiang et Zhai [Jiang et Zhai, 2007] ont proposé une méthode heuristique pour minimiser la différence des probabilités conditionnelles des domaines source et cible en supprimant les échantillons qui perturbent l'apprentissage cible. Duan *et al.*[Duan *et al.*, 2009] ont proposé une approche "Domain Transfer SVM (DTSVM)" pour une tâche de classification de vidéos. DTSVM minimise la fonction de risque structurel de SVM et la divergence moyenne maximale. C'est le critère qui identifie la différence entre la distribution des exemples sources et cibles en apprenant une fonction de noyau optimisée.

Wang *et al.*[Wang *et al.*, 2010] ont introduit un système de classification visuelle des objets à marge maximale qui se base sur l'hypothèse qu'un modèle d'objet doit répondre avec précision forte aux exemples de catégories sources similaires et doit répondre négativement aux catégories sources non-similaires. Lim *et al.*[Lim *et al.*, 2011] ont illustré l'amélioration de performance par emprunt et pondération d'un ensemble d'échantillons à partir de plusieurs catégories d'objet visuellement proche lors d'apprentissage d'un détecteur d'objet cible (canapé). La FIGURE 15 illustre le principe du modèle proposé. Il cherche les bons échantillons à emprunter en associant un poids à chaque échantillon et cherche les bonnes transformations à appliquer pour chaque échantillon pour augmenter la flexibilité d'emprunt. Les exemples transformées dans le rectangles bleu (ou rouge) sont similaires en vue de face (ou de profil) du canapé. Les images barrées ne seront pas empruntés pour l'apprentissage parce qu'elles ont des poids faibles.

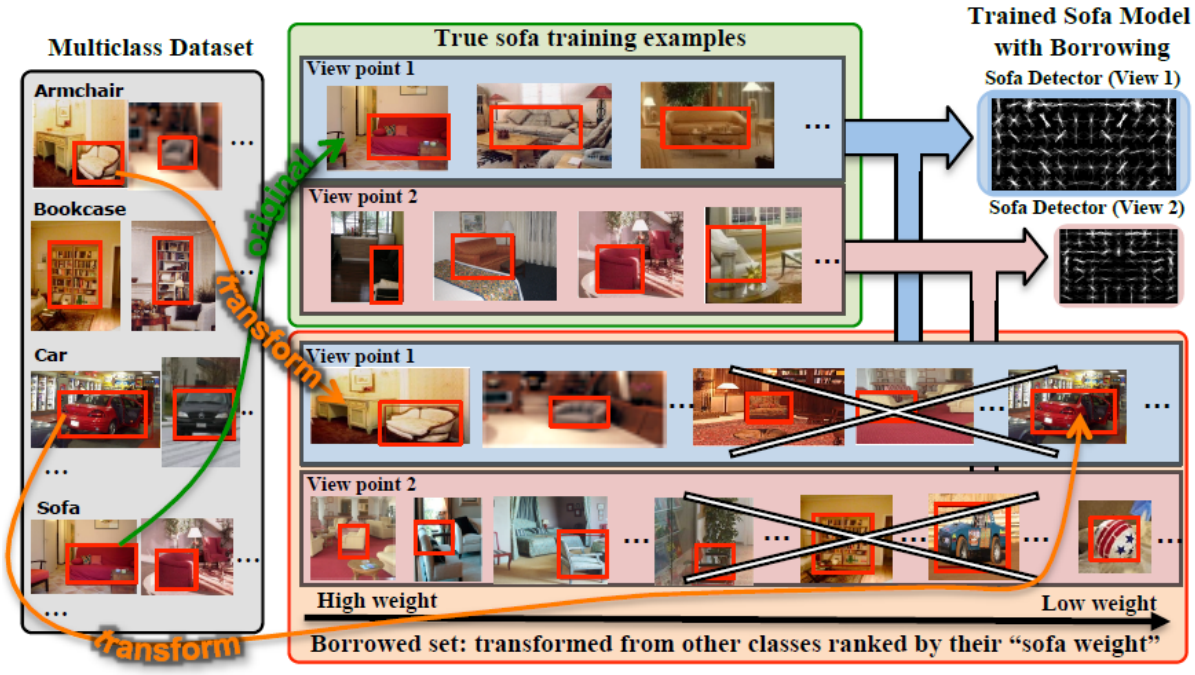


FIGURE 15 – Illutration de transfert d'exemples : Apprentissage d'un détecteur de canapé par emprunt d'échantillons à partir d'autres classes reliées visuellement [Lim *et al.*, 2011].

Haung *et al.*[Huang *et al.*, 2006] ont proposé un algorithme de ressemblance KMM (a kernel-mean matching) pour l'apprentissage direct du ratio de la distribution source par rapport à la distribution cible, par correspondance des deux moyennes des données sources et cibles en produisant un espace de noyau Hilbert. L'avantage principal de l'utilisation de KMM est la possibilité d'éviter l'estimation de densité pour les deux domaines qui peut être difficile si l'ensemble de données est réduit.

Sugiyama *et al.*[Sugiyama *et al.*, 2008] ont proposé un algorithme connu sous le nom de *Kullback-Leibler Importance Estimation Procedure (KLIEP)* pour estimer directement le ratio de densité source par rapport à la densité cible en se basant sur la minimisation de la divergence Kullback-Leibler. KLIEP peut être intégré à la validation croisée pour effectuer la sélection du modèle automatiquement en deux étapes : (1) l'estimation des poids des données de domaine source ; (2) apprentissage de modèles avec des données pondérées.

5.2 Méthodes de transfert de modèle

Ce type de transfert consiste à transférer les paramètres du modèle entraîné sur le domaine source. Il se base sur l'hypothèse que les modèles de tâches liées doivent partager un certain nombre de paramètres ainsi que les distributions des hyperparametres.

Fei-Fei *et al.*[Fei-Fei *et al.*, 2006] ont introduit une approche bayésienne de transfert qui utilise les paramètres du classifieur source et apprend le modèle cible en mettant à jour les paramètres du modèle à la l'aide d'un ou de plusieurs exemples de la catégorie cible. Zweig et Weinshall [Zweig et Weinshall, 2007] ont proposé une approche de transfert d'apprentissage qui combine plusieurs classifieurs d'objet de différents niveaux hiérarchiques en un seul classifieur en utilisant une configuration basée sur des modèles de catégorie d'objet. L'objectif de cette méthode est de capturer différents aspects de l'objet.

Yang *et al.*[Yang *et al.*, 2007] ont présenté une approche dite A-SVM *(Adaptative Support Vector Machines)* pour l'adaptation de classifieurs à des nouveaux domaines. C'est une variante du transfert d'apprentissage à marge maximale qui tire profit de la connaissance visuelle des données sources ou d'autres formes des connaissances antérieures.

Gao *et al.*[Gao *et al.*, 2008a] ont choisi de combiner un ensemble de modèles au lieu d'utiliser un seul modèle pour transférer les connaissances utiles vers le domaine cible. Ils ont proposé de : (i) approximer les poids de modèles en se basant sur des structures variées dans le domaine cible, (ii) fournir une estimation basée sur le voisinage des graphes, (iii) fournir une étape de prédiction pour propager des étiquettes à partir des échantillons proches. La prédiction est utile si les modèles d'apprentissage sont incapables de fournir une réponse précise pour certains échantillons.

Stark *et al.*[Stark *et al.*, 2009] ont proposé un modèle basé sur une forme probabiliste qui permet le transfert des connaissances sur trois niveaux différents : la forme et l'apparence des parties, la symétrie locale entre les parties, et une partie de la topologie. Le modèle permet d'effectuer un transfert partiel ou complet des connaissances en transférant les paramètres du modèle. La FIGURE 16 montre un exemple de transfert de paramètres.

Aytar et Zisserman [Aytar et Zisserman, 2011] ont proposé de transférer un modèle d'une première catégorie vers une catégorie cible ; adapter un modèle de moto vers un modèle de vélo et un modèle de cheval vers un modèle d'âne. Afin d'éviter l'apprentissage du modèle cible à partir de zéro, ils ont introduit le modèle de la catégorie source pre-entrainé comme un régulateur de la fonction de coût lors d'apprentissage de la catégorie cible. Gao *et al.*[Gao *et al.*, 2012] se sont basés sur l'hypothèse que les bons détecteurs doivent partager certaines propriétés statistiques. Ils ont pris des statistiques de bas-niveau des distributions de probabilité d'un ensemble de variables aléatoires à travers les paramètres d'un modèle source. Ils ont proposé de renforcer ces statistiques en apprenant le modèle de la tâche cible.

5.3 Méthodes de transfert de la représentation des caractéristiques

Ce type de transfert se positionne entre le transfert d'exemples et le transfert des paramètres du modèle. L'objectif est de trouver une bonne représentation des caractéristiques qui simplifie l'apprentissage de la solution du problème cible. Une représentation de caractéristiques qui minimise la divergence des domaines et l'erreur de modèle de classification ou de régression. En particulier, lorsque le problème de la cible a très peu d'exemples et la généralisation est difficile, ce type de transfert permet

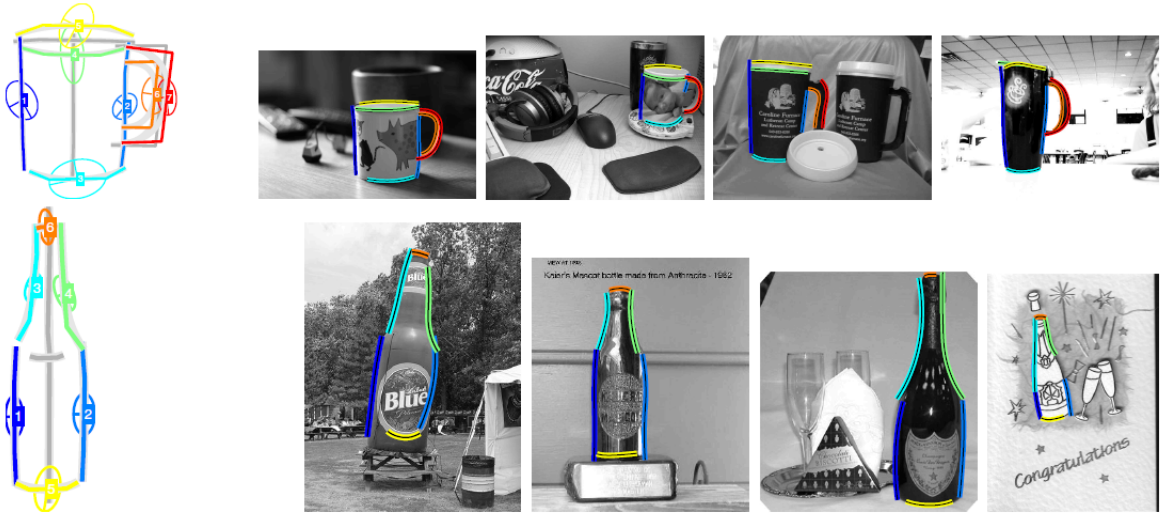


FIGURE 16 – Transfert partiel et complet de connaissances en transférant les paramètres du modèle : la forme de modèle à gauche et des exemples de détections à droite [Stark *et al.*, 2009]

de mieux guider l'apprentissage en limitant l'espace de recherche des caractéristiques en des caractéristiques les plus significatives. Le transfert de représentation des caractéristiques peut être considéré comme une étape de passage d'un bas niveau de caractéristique à un niveau moyen. Rania *et al.*[Raina *et al.*, 2007] ont proposé d'appliquer une méthode de codage épars. C'est une méthode de construction de caractéristiques non-supervisé pour l'apprentissage de caractéristiques haut-niveau. L'inconvénient de cette dernière est qu'elle se base sur des vecteurs haut niveau de biais appris et optimisés pour le domaine source et qui peuvent ne pas être adaptés pour le domaine cible.

Wang et Mahadevan [Wang et Mahadevan, 2008] ont proposé une approche basée sur l'analyse procustéenne [1] pour multiplier les alignements sans correspondance afin de transférer des connaissances à travers les domaines. Fink [Fink, 2005] a présenté une méthode d'apprentissage d'un classifieur d'objet à l'aide d'un seul échantillon. Pour ce faire, il a donné un poids élevé aux dimensions pertinentes des caractéristiques pour la classification en utilisant les exemples disponibles de classes connexes. Quattoni *et al.*[Quattoni *et al.*, 2008] ont proposé d'apprendre une représentation d'image prototype éparse à partir de données non-étiquetées et des données de catégories visuelles liées. Cette approche peut exploiter toute fonction du noyau arbitraire. La méthode est basée sur l'approximation conjointe sur l'espace des prototypes pour trouver un sous-ensemble de prototypes discriminants.

Saenko *et al.*[Saenko *et al.*, 2010] ont introduit une méthode qui adapte des domaines visuels particuliers à des nouvelles conditions d'images en appliquant des transformations qui minimisent les changements induits par le domaine dans la distribution des caractéristiques. Yao *et al.*[Yao *et al.*, 2011] ont proposé de représenter l'image par les réponses attribuées d'un ensemble de classifieurs qui sont entrainés dans un contexte supervisé. Le contenu de chaque image est décrit en utilisant un ensemble d'actions de base. La FIGURE 17 présente le principe de l'approche de Yao *et al.*[Yao *et al.*, 2011] pour la description de l'action humaine.

Blitzer *et al.*[Blitzer *et al.*, 2006] ont proposé un algorithme de correspondance structurelle (*connu par structural correspondence learning (SCL)*) qui utilise des données non étiquetées de domaine cible pour extraire certaines caractéristiques pertinentes qui réduisent la différence entre les domaines. L'algorithme SCL définit un ensemble de caractéristiques pivot sur des données non étiquetées des deux domaines. Ensuite, le SCL supprime ces caractéristiques à partir des données et traite chaque carac-

1. L'analyse procustéenne est une technique de comparaison d'objets. Elle se base sur l'idée de déformer un objet afin de le rendre semblable à une référence sans déformer sa forme intrinsèque.

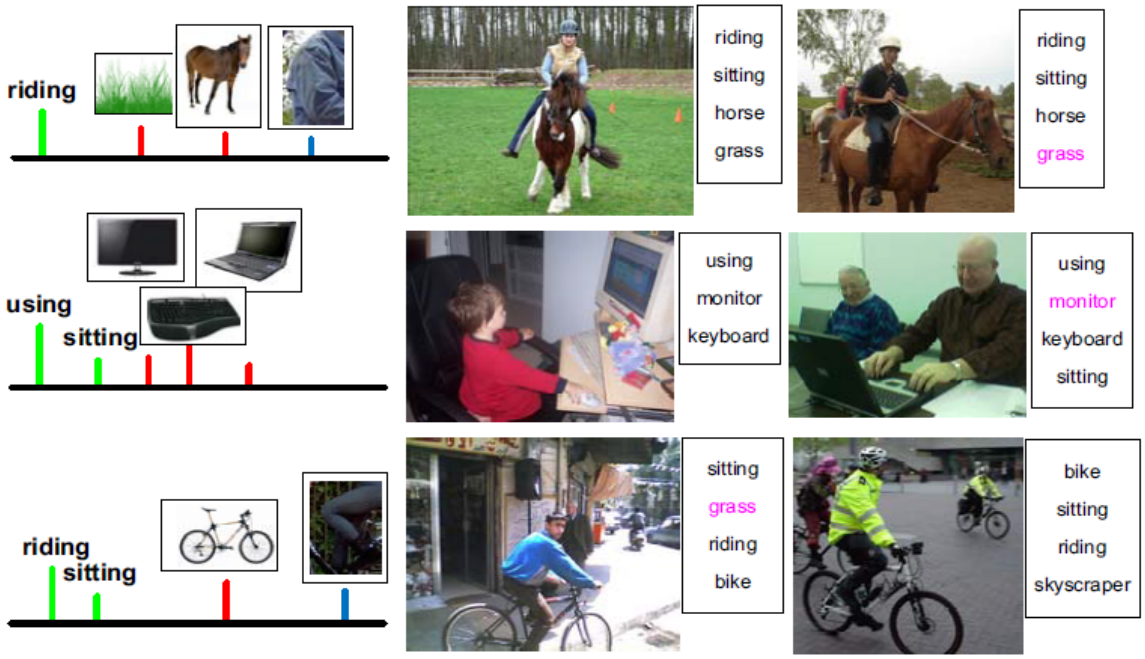


FIGURE 17 – Description de l'action humaine : Une action est représenté sous la forme d'une sommation pondéréé d'un sous-ensemble d'attributs et de partie d'action de base [Yao *et al.*, 2011]

téristique comme un nouveau vecteur d'étiquette. Les caractéristiques pivot sont des caractéristiques spécifiques aux domaines et des connaissances à priori. Si les caractéristiques pivot sont bien définies, alors le modèle d'apprentissage code bien la correspondance entre les caractéristiques des différents domaines. Néanmoins, l'inconvénient de cette méthode est comment sélectionner ces caractéristiques ?

Dans [Xue *et al.*, 2008], Xue *et al.* ont proposé un algorithme appelé Topic-bridged PLSA (TPLSA) pour gérer le problème de classification de texte inter-domaines en permettant le transfert des connaissances acquises à partir des documents d'un domaine à un autre domaine. L'algorithme a étendu l'algorithme d'analyse sémantique latente probabiliste traditionnelle (PLSA) pour intégrer des données étiquetées et non étiquetées de différents domaines, mais liés, dans un modèle de probabilité conjointe.

Pan *et al.*[Pan *et al.*, 2008b] ont exploité la méthode d'incorporation de divergence de moyenne maximale (*connue par Maximum Mean Discrepancy Embedding (MMDE)*, désignéé pour la réduction de dimension des caractéristiques, pour apprendre un espace de dimension faible permettant de réduire la différence des distributions entre les domaines source et cible. Cependant, cette méthode souffre d'un taux de complexité élevé. Une amélioration de MMDE est proposée par Pan *et al.*[Pan *et al.*, 2011] qui se résume en une extraction efficace des caractéristiques connues sous le nom d'analyse de composantes de transfert (*en anglais Transfer Component Analysis (TCA)*). Une idée similaire est proposée par Wang *et al.*[Wang *et al.*, 2008] qui se résume en une analyse de transfert discriminatoire connue sous le nom *transferred discriminative analysis (TDA)*. TDA est un algorithme qui s'éxécute de manière itérative pour trouver le meilleur sous-espace pour les données cibles. Il applique des méthodes de *clustering* pour générer des étiquettes pseudo-classe pour les données cibles non-étiquetées. Ensuite, il applique des méthodes de réduction de dimension aux données cibles et aux données sources étiquetées.

Douze *et al.*[Douze *et al.*, 2011] ont montré que la classification d'image à base d'une représentation haut-niveau (à base des attributs) améliore la tâche de recherche d'image. Ainsi, ils ont montré que la combinaison des attributs avec le Fisher permet d'augmenter les performances. Song *et al.*[Song *et al.*, 2011] ont proposé d'exploiter, d'une manière itérative, les sorties d'une tâche en tant que caractéristiques haut-niveau d'une autre tâche afin d'améliorer la classification et la détection d'objets dans des nouvelles conditions.

5.4 Méthodes de transfert des connaissances relationnelles

D'après Pan et Yang [Pan et Yang, 2010], les méthodes de transfert des connaissances relationnelles traitent les problèmes de transfert dans des domaines relationnels, où les données ne sont pas indépendantes et identiquement distribuées (non-i.i.d) et peuvent être représentées par des relations multiples. Par exemple, des données de réseau Internet et des données des réseaux sociaux. Ce type de méthodes cherche à transférer la relation entre les données d'un domaine source à un domaine cible et il se base généralement sur des techniques statistiques d'apprentissage relationnel pour résoudre les problèmes de la tâche cible. Les réseaux de Markov logique [Richardson et Domingos, 2006] (*connus par Markov Logic Networks (MLN)*) est une manière simple qui combine la probabilité et la logique du premier ordre. Un réseau MLN est obtenu en associant des poids à des formules (ou clauses) suivant une base de connaissance du premier ordre et peut être vue comme un modèle pour la construction ordinaire des réseaux de Markov. Dans un réseau MLN, les entités d'un domaine relationnel sont présentées par des prédicats et leurs relations sont présentées par une logique du premier ordre. Mihalkova *et al.*[Mihalkova *et al.*, 2007] ont présenté un système complet qui utilise des MLNs. Le système est proposé sous le nom *Transfer via Automatic Mapping And Revision TAMAR*, il transfère automatiquement l'architecture des relations d'un domaine source à un domaine cible et après transforme la structure transférée pour améliorer sa performance dans un domaine cible. A titre d'exemple, un gérant d'une entreprise dans un cadre industriel peut jouer le même rôle qu'un enseignent dans le cadre académique. De plus, la relation entre le gérant et les ouvriers est similaire à la relation d'un enseignent avec ses étudiants. Ainsi, il y a une possibilité de transfert à partir du gérant vers un enseignant et de la relation gérant-ouvriers vers la relation enseignant-étudiants.

6 L'apprentissage Multi-tâches

Généralement, le transfert d'apprentissage peut être vue comme étant un cas particulier de l'apprentissage multi-tâches (connu en anglais *Multi-task Learning (MLT)*). La plupart des travaux d'apprentissage multi-taches peuvent être facilement modifiés pour l'apprentissage par transfert. Néanmoins, le transfert d'apprentissage a pour objectif de transmettre des connaissances sources vers un domaine cible pour l'amélioration de la performance d'une tâche cible alors que l'apprentissage multi-tâches vise à apprendre toutes les tâches simultanément pour renforcer la relation entre ces tâches. De plus, les poids des données sources et cibles dans les fonctions objectifs sont les mêmes dans l'apprentissage multi-tâches. En revanche, dans l'apprentissage par transfert, les poids des différents domaines peuvent être différents dans les fonctions objectifs [Pan et Yang, 2010]. Particulièrement, l'apprentissage multi-tâches est recommendé lorsque chacune des tâches possède un nombre limité d'échantillons d'apprentissage parce qu'il donne des performances nettement supérieures aux performances de l'apprentissage de chaque tâche séparément [Aytar, 2014].

Les approches d'apprentissage multi-tâches sont divisés en deux catégories selon la relation des tâches [Aytar, 2014] :

- La première catégorie recommande que les vecteurs de classifieurs soient proches les uns des autres. La relation entre les tâches peut être satisfaite par la minimisation des normes de Frobenius des différences de vecteurs de classifieurs ou bien par le partage de certains points en communs. Daumé III [Daumé III, 2009] a proposé une fonction de mappage de noyaux pour traiter les problèmes NLP. Cette fonction représente les données des deux domaines source et cible dans un espace de primitives de dimension élevée. Ensuite, Daumé III utilise des méthodes discriminantes pour l'apprentissage de classifieur. La limite de cette fonction est que le noyaux construit dépend de la connaissance du domaine ce qui rend difficile la généralisation de la fonction à d'autres domaines et d'autres applications.

- La deuxième catégorie nécessite que les paramètres de modèle sont générés à partir d'une représentation de caractéristiques communes. Cette représentation est induite principalement par la régularisation de la matrice des paramètres du modèle où chaque colonne correspond aux paramètres d'une tâche séparée. Nous citons à titre d'exemple les travaux de Kumar et Daumé III [Kumar et Daume III, 2012] qui ont proposé une approche de regroupement et de chevauchement des tâches d'apprentissage dans un apprentissage multi-tâches. Les paramètres de chaque groupe de tâches sont supposés situées dans un sous-espace de faible dimension. L'approche ne suppose pas de structure de groupement disjointe, et les tâches de différents groupes sont autorisés à se chevaucher les uns avec les autres en partageant une ou plusieurs tâches de base.

Les travaux traditionnels d'apprentissage multi-tâches cherchent à entrainer conjointement plusieurs tâches de prédiction qui sont appliquées pareillement et à partager de l'information à travers ces tâches. Cependant, les nouveaux travaux visent à déterminer les relations entre les différentes tâches. Ceci permet de préciser les différents sous-groupes de tâches qui peuvent être bénéfiques à être proches avec des poids différents et les tâches non liées n'auront pas d'effet les unes sur les autres [Aytar, 2014]. Le travail de Kumar et Daumé III [Kumar et Daume III, 2012] permet la sélection de partage d'information à travers les tâches.

7 Les applications pour la détection d'objets

La littérature présente beaucoup d'applications de transfert pour la classification d'images dans un domaine cible [Raina *et al.*, 2007, Dai *et al.*, 2007, Tommasi *et al.*, 2010, Tommasi, 2013, Tommasi et Caputo, 2009], la reconnaissance des écritures manuscrites, la reconnaissance des caractères alphabétiques, la classification des pages Web [Raina *et al.*, 2007], la résolution des problèmes psychologiques de la programmation neuro-linguistique [Blitzer *et al.*, 2006, Jiang et Zhai, 2007], la localisation wifi et la classification de texte [Pan *et al.*, 2011], etc. Néanmoins, nous nous intéressons dans cette section aux travaux qui traitent du transfert d'apprentissage pour la détection d'objets. Particulièrement, la détection d'objets doit prendre en considération un nombre important des défis. Nous citons par exemple, l'alignement des images d'apprentissage pour l'entrainement du détecteur, la prise en considération d'un ensemble des a priori pour établir la correspondance entre les modèles source et cible. Ainsi, la prise en compte des différentes tailles de l'objet d'intérêt et des points de vue différents entre les domaines source et cible.

Zhang *et al.*[Zhang *et al.*, 2008] ont proposé une méthode d'adaptation du classifieur par combinaison de la fonction objectif du domaine source avec celle du domaine cible. Ils ont approximé le terme d'apprentissage par une expansion de Taylor de second ordre pour réduire la quantité d'information nécessaire à l'adaptation pour la détection de piétons.

Aytar et Zisserman [Aytar et Zisserman, 2011] ont proposé d'appliquer le transfert entre catégories d'objets similaires (la FIGURE 18 présente le principe de la méthode et illustre des résultats de détection avec le détecteur de vélo obtenu). Ils ont présenté une modification de la fonction objectif de l'apprentissage qui conserve la convexité et l'optimisation des méthodes SVM et ont apporté le bénéfice d'apprendre un modèle cible avec un peu d'échantillons. Ils ont présenté deux applications de la méthode proposée : une application pour transfert entre catégories et une deuxième application de transfert d'une classe supérieure vers une classe subordonnée (transfert d'un détecteur générique d'animal vers une catégorie spécifique tels que cheval, mouton, vache)

Kuzborskij *et al.*[Kuzborskij *et al.*, 2013] ont étendu la méthode PMT-SVM developpé dans [Aytar et Zisserman, 2011] pour appliquer le transfert d'un problème multi-classes à un autre problème multi-classes dans un cadre de classification d'images. Donahue *et al.*[Donahue *et al.*, 2013] ont élaboré une extension de la PMT-SVM [Aytar et Zisserman, 2011] pour la classification multi-classes avec des données de plusieurs points de vue et pour la détection d'objets dans les vidéos. L'extension élaborée

intègre une régularisation laplacienne qui combine des échantillons traditionnellement étiquetés avec des contraintes codées sur les échantillons non-étiquetés du domaine cible.

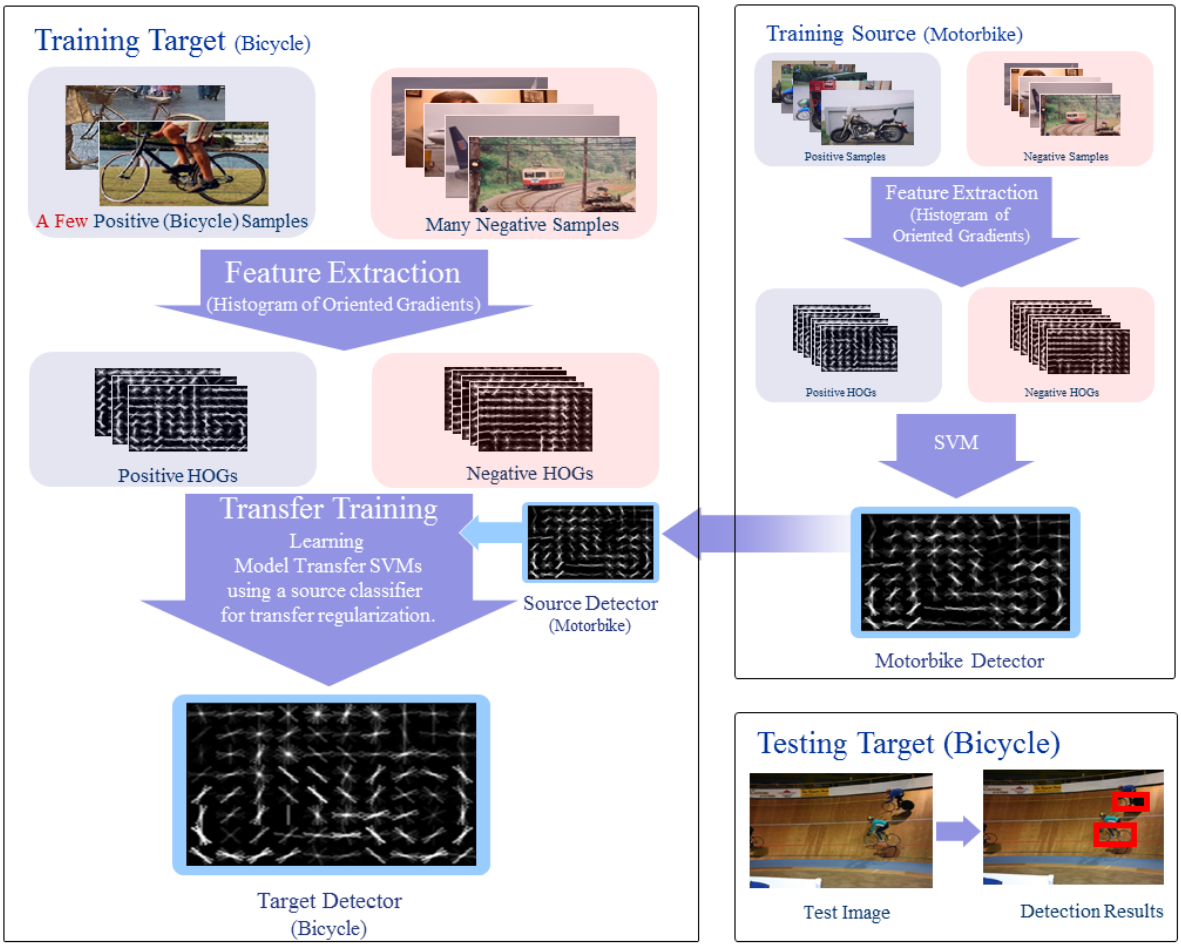


FIGURE 18 – Méthode de transfert de Aytar et Zisserman [Aytar et Zisserman, 2011]. Apprentissage d'un détecteur de vélo à base de peu d'échantillons de vélo et un détecteur source de moto.

Lim *et al.*[Lim *et al.*, 2011], et Gao *et al.*[Gao *et al.*, 2012] ont présenté des applications de transfert entre différentes catégories d'objets pour améliorer l'apprentissage d'une catégorie cible. Wang et Wang [Wang et Wang, 2011] ont introduit une méthode d'adaptation d'un détecteur générique de piétons vers un détecteur spécifique à une scène de trafic routier (plus de détails sont disponibles dans la section 5.3 du chapitre 2, page 45). Tang *et al.*[Tang *et al.*, 2012] ont utilisé des variables binaires pour pondérer des exemples qui seraient ajoutés ou exclus de l'ensemble d'apprentissage pour adapter un détecteur de vélo, chien ou voiture entrainé sur des images statiques en un détecteur du même objet dans une vidéo. Wang *et al.*[Wang *et al.*, 2012b] ont proposé une méthode non-paramétrique qui utilise un arbre de vocabulaire comme un vecteur binaire pour coder un exemple visuel pour l'adaptation d'un détecteur de piétons à une vidéo.

Conclusion

Dans ce chapitre nous avons commencé par une vue générale sur les trois familles des méthodes de la détection d'objets et sur les différents types d'apprentissage semi-supervisé. Ensuite, le transfert d'apprentissage naturel et celui artificiel sont présentés. Dans la cinquième section, nous avons dressé un état de l'art sur la catégorisation des techniques de transfert d'apprentissage en fonction de type des

connaissances sources transférées. Nous avons exposé l'apprentissage Multi-tâches tout en précisant ses différences avec le transfert d'apprentissage dans la sixième section. Et nous avons donné un aperçu sur certaines applications du transfert pour la détection d'une catégorie d'objet dans la dernière section. Dans le chapitre suivant, nous allons présenter l'analyse des scènes de trafic routier et nous allons détailler des travaux qui font l'usage du transfert d'apprentissage pour la détection des objets de trafic routier.

Chapitre 2

L'analyse automatique des scènes de trafic routier

Introduction

Les progrès de la vision par ordinateur, des technologies des caméras et de l'analyse automatique de vidéo ont contribué au développement des applications dans le domaine de vidéo-surveillance du trafic routier. Cela consiste à installer des Systèmes Intelligents de Transport (ITS : *Intelligent Transport Systems*) aux bords des routes et à l'intérieur des véhicules afin de traiter automatiquement et le plus rapidement possible les informations du trafic. Les objectifs de ces systèmes sont l'amélioration des performances d'infrastructures, de la sécurité et du confort des usagers. Ces systèmes ITS se basent principalement sur l'apparence visuelle pour la détection, la reconnaissance et le suivi des objets mobiles du trafic afin de détecter les incidents (les accidents et les encombrements), analyser le comportement des objets et fournir une description de fluidité du trafic (fluide, ralenti, bouchon, impossible (axe coupé), inconnu).

Dans la première section, nous définissons l'analyse automatique des scènes de trafic routier. Ensuite, dans la deuxième section, nous exposons les jeux de primitives les plus utilisées pour la détection d'objets de trafic à travers une vue sur certains travaux existants. Dans la troisième section, nous présentons le détecteur choisi pour nos travaux et nous donnons un aperçu sur ses caractéristiques dans la quatrième section. Dans la cinquième section, nous illustrons les limites des détecteurs génériques et les contraintes liées à une solution intuitive. Ensuite, nous présentons des travaux existants de la spécialisation de détecteurs et leurs inconvénients. Enfin, dans la dernière section, nous présentons l'outil d'évaluation d'un détecteur d'objet et les bases de données sources et cibles utilisées dans nos expérimentations.

1 Définition de l'analyse automatique des scènes de trafic routier

Dans un contexte de vidéo-surveillance, l'analyse automatique des scènes de trafic routier est un traitement qui considère en entrée une séquence vidéo avec ou sans autres données *a priori* sur la scène et / ou sur la caméra et qui fournit en sortie un ensemble de statistiques de haut niveau. La Figure 19 illustre la chaine d'analyse automatique d'une scène de trafic routier.

La vidéo d'entrée est généralement enregistrée par une caméra montée en face d'une branche de route. Les données a priori peuvent être la distance entre la caméra et la scène, un masque pour isoler la zone de traitement, les paramètres intrinsèques de la caméra, etc. Les informations de sortie peuvent être de nature différentes telles qu'un tableau de comptage de véhicules en fonction de leur catégorie, une base de données des trajectoires de chaque objet, des matrices origines / destination, etc.

Afin d'établir les statistiques de sortie, le traitement peut être composé d'une combinaison et/ou une substitution des étapes suivantes : détection, reconnaissance ou classification et suivi de mouvement des différents objets de la scène du trafic.

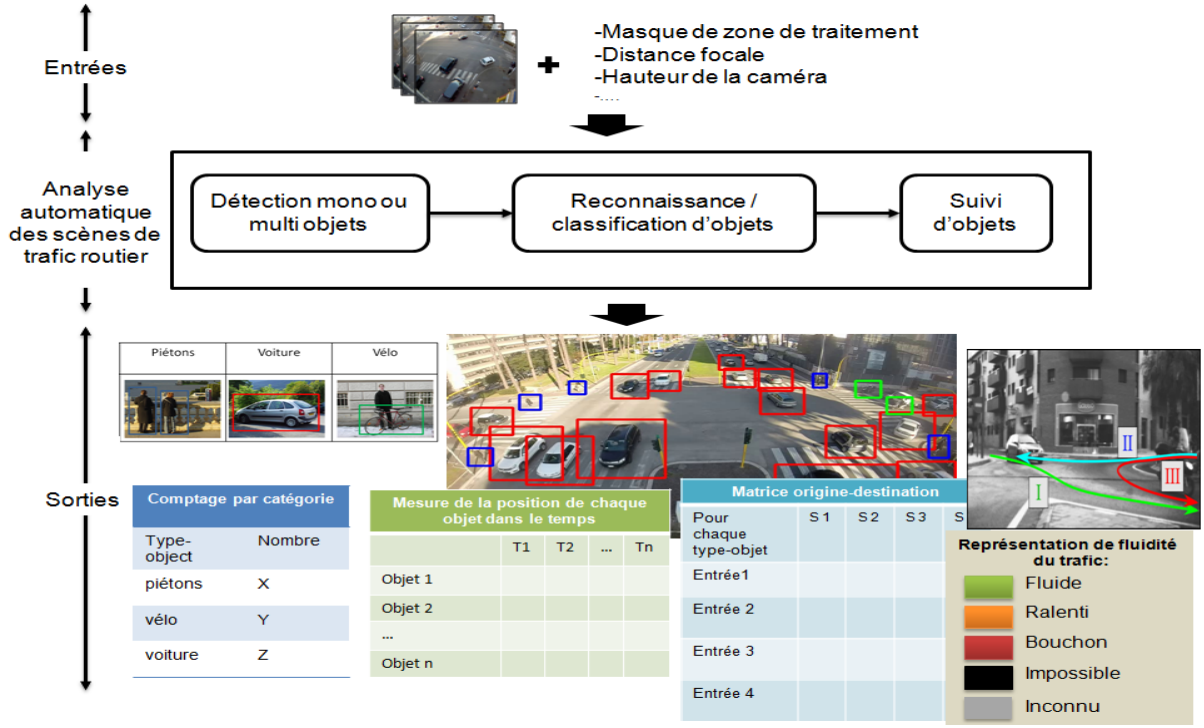


FIGURE 19 – Chaine d'analyse automatique des scènes de trafic routier

Il est à noter qu'un objet de trafic est toute entité mobile dans une scène de circulation routière. Cet objet peut être un piéton, une voiture, une moto, un vélo, un bus, un tram, un camion, une camionnette et un utilitaire. Une portion de route observée par une caméra peut être une intersection, un carrefour giratoire, un embranchement, une bifurcation, etc. La FIGURE 20 illustre certains exemples de scènes.

2 Descripteurs pour la vidéo-surveillance du trafic

Dans ce travail, nous nous intéressons principalement à la détection d'objets qui constitue la première étape de l'analyse automatique de vidéo (FIGURE 19 dans plusieurs applications de ITS. Ainsi, la précision et la robustesse des techniques de détection d'objets utilisées dans cette étape ont une grande importance par la suite sur les étapes de reconnaissance et de suivi.

La littérature en vidéo-surveillance du trafic présente deux catégories de techniques de détection d'objets : des techniques de segmentation de mouvement qui utilisent l'information de mouvement pour distinguer les piétons et les véhicules du fond statique de la scène. Cependant, cette catégorie de techniques ne donne aucune information sur la nature de l'objet. La deuxième catégorie de techniques qui nous intéressent le plus, utilise les caractéristiques d'apparence d'objets pour isoler l'objet du fond de la scène qui l'entoure, tout en donnant une description sémantique sur la classe de l'objet.

En particulier, les caratéristiques SIFT [Lowe, 1999], SURF [Bay *et al.*, 2006], pseudo-Haar (Haar-like)[Viola et Jones, 2001a, Viola et Jones, 2001b] et les HOG [Dalal et Triggs, 2005] sont les caractéristiques les plus populaires qui permettent un codage de l'objet sous la forme d'un vecteur. Le vecteur est ensuite utilisée dans une méthode basée apprentissage pour caractériser la classe d'objet.

FIGURE 20 – Exemples de scènes de circulation routière

SIFT :

Scale Invariant Feature Transformation sont des caractéristiques détectées par une approche de filtrage par étage, qui identifie l'orientation locale du contour autour des points clés stables dans l'espace d'échelles. Ces caractéristiques sont invariantes aux changements d'échelle de l'image, à la translation et à la rotation, elles sont partiellement invariantes aux changements d'illumination et à la projection affine. Ces caractéristiques décrivent l'apparence des points saillants uniquement et peuvent être utilisées pour chercher la correspondance des points d'objets d'une images à une autre dans une vidéo. En plus du vecteur de primitives, l'échelle des caractéristiques et l'orientation de chaque point clé sont aussi calculées [Lowe, 1999].

Une représentation plus riche des classes de véhicules sous forme d'un descripteur SIFT modifié est utilisée dans [Ma et Grimson, 2005] pour la détection de différentes catégories de véhicules. Gea *et al.*[Gao *et al.*, 2008b] ont proposé une approche qui se base sur la ré-identification des points d'intérêts SIFT comme particules initiales pour améliorer les performances de suivi de véhicule dans une scène de vidéo-surveillance.

Qian *et al.*[Qian *et al.*, 2013] ont combiné le descripteur SIFT avec le classifieur SVM pour la détection et le suivi multi-objets. La combinaison proposée permet une bonne performance de suivi dans des situations complexes. Par contre, la dimension du descripteur SIFT, son coût algorithmique ainsi que la faible adaptation aux variations d'illumination sont les limites d'utilisation de ces caractéristiques dans des applications qui exigent un traitement en temps réel tel que la vidéo-surveillance du trafic.

SURF :

Speeded up Robust Features sont des caractéristiques proposées par [Bay *et al.*, 2006], sous forme d'un descripteur de points d'intérêt invariant à l'échelle et à la rotation. Le filtre gaussien de SIFT a été remplacé par une boîte de filtres ce qui permet de calculer ces caractéristiques avec une complexité plus réduite que celle de SIFT mais avec une baisse légère au niveau de la performance. Afin de localiser les points d'intérêt, l'algorithme SURF adopte une approximation de matrice Hessienne sur une image intégrale.

Cependant, l'inconvénient du descripteur SURF est qu'il est incapable de détecter des objets symétriques qui sont souvent présents dans les scènes de circulation routière, en particulier tous les véhicules ont une propriété de symétrie horizontale. Pour corriger cet inconvénient, Hsieh *et al.*[Hsieh *et al.*, 2014] ont proposé un nouveau descripteur SURF symétrique pour faire correspondre toutes les paires symétriques à l'aide d'une transformation miroir. Ensuite, ils l'ont appliqué pour la détection et la reconnaissance de marque et de modèle des véhicules. Shujuan *et al.*[Shujuan *et al.*, 2015] ont utilisé une mixture des caractéristiques pseudo-Haar et SURF avec un classifieur en cascade et le classifieur Gentle AdaBoost pour une détection en temps réel de véhicules. Malgré la robustesse et la capacité de détection en temps réel, le descripteur SURF est peu avantageux pour la surveillance de trafic routier à cause de son instabilité vis-à-vis des variations d'illumination.

Pseudo-Haar (Haar-like features) :

Les caractéristiques pseudo-Haar représentent les différences d'intensités ; horizontales, verticales, diagonales, entre la région médiane et les zones de bordure, et entre le centre et les zones alentours [Viola et Jones, 2001a, Viola et Jones, 2001b]. Le recours au traitement d'images par ces caractéristiques est motivé par leur rapidité. Lors de la détection d'objets, la méthode d'image intégrale est utilisée pour déterminer la présence ou l'absence des caractéristiques dans chaque position de l'image quelle que soit la taille. Le principal avantage de cette méthode est la réduction du temps de calcul du despcriteur.

Elkerdawi *et al.*[Elkerdawi *et al.*, 2014] ont utilisé les caractéristiques pseudo-Haar pour entrainer un classifier en cascade en utilisant l'algorithme AdaBoost afin de détecter et suivre des véhicules dans une scène de surveillance d'autoroute. Miller *et al.*[Miller *et al.*, 2015] ont traité l'application de comptage de véhicules en utilisant les caractéristiques pseudo-Haar et en modélisant le mouvement des véhicules par un modèle de Markov caché. Ces caractéristiques sont utilisées également avec la notion de mouvement des véhicules par Bai *et al.*[Bai *et al.*, 2006] et Momin et Mujawar [Momin et Mujawar, 2015] pour la détection des véhicules dans une autoroute et dans une scène urbaine. La rapidité de calcul et la sensibilité aux structures verticales, horizontales et symétriques, rendent les caractéristiques pseudo-Haar bien adaptées pour les applications de temps réel. Mais, l'apprentissage du classifieur en cascade généralement utilisé avec ces caractéristiques nécessite une base de données importante. Elle prend beaucoup de temps ce qui les rend moins avantageuses pour une utilisation dans le cas du transfert d'apprentissage où il faut entrainer plusieurs fois le système.

HOG :

Dalal et Triggs [Dalal et Triggs, 2005] ont proposé de calculer (Histogram of Oriented Gradient) les histogrammes directionnel du gradient de l'image. Le descripteur HOG est une représentation de l'information de gradient des contours. Les HOG avec le classifieur SVM ont été proposés initialement sous forme d'un détecteur HOG-SVM de piétons. Une approche en deux étapes a été présentée par Han *et al.*[Han *et al.*, 2006] pour détecter des personnes et des voitures dans des images statiques de manière robuste en étendant les descripteurs HOG pour la classification.

Buch *et al.*[Erich Buch *et al.*, 2009] ont étendu le concept de HOG à 3D-HOG pour la détection de piétons et des véhicules dans des scènes urbaines. Dans [Erich Buch *et al.*, 2009], des surfaces de modèles

3D ont été utilisés à la place des grilles 2D des cellules. Les principaux avantages des descripteurs HOG sont : un temps de calcul réduit, une invariance d'illumination et de géométrie et une haute puissance discriminatoire. Il est à noter que la version originale des caractéristiques HOG est non-adaptée aux applications temps réel. Prisacariu et Reid [Prisacariu et Reid, 2009] ont utilisé le GPU et la plateforme de calcul parallèle NVIDIA CUDA pour une implémentation parallèle des HOGs qui a permis une détection d'objets avec des performances temps réel. Negri *et al.*[Negri *et al.*, 2008] ont publié un travail qui utilise un classifier à base de l'algorithme Discrete Adaboost associé avec les caractéristiques HOG. Il ont construit trois détecteurs ; un détecteur entrainé sur des caractéristiques pseudo-Haar, un sur les HOG et un détecteur sur la combinaison des deux types de caractéristiques. Leur étude a montré que le détecteur fusionnant les caractéristique combine les avantages de Haar et de HOG, et atteint un taux de bonnes détections élevé avec un nombre réduit de fausses détections. Sun et Watada [Sun et Watada, 2015] ont proposé des nouvelles caractéristiques boosted-HOG pour la détection de véhicules et de personnes dans des images fixes de circulation routière. Les caractéristiques boosted-HOG sont calculées après l'entraînement du classfieur AdaBoost sur les échantillons de la base d'apprentissage. Un SVM linéaire est ensuite entrainé sur ses caractéristiques boosted-HOG. Ces dernières combinent à la fois les avantages de HOG et ceux de classifieur AdaBoost.

Caractéristiques CNN ou DCNN :

Deep Convolutional Neural Networks (*CNN, ou DCNN, ou ConvNet*) sont des réseaux de neurones convolutifs profonds. L'avantage principal de ConvNets est qu'ils peuvent être entraînés de bout en bout [Sermanet *et al.*, 2013]. Depuis l'année 2012, ils ont atteint la meilleure performanace de classification par rapport aux autres méthodes d'état de l'art sur la base ImageNet Large Scale Visual Recognition Challenge (ILSVRC) contenant 1.2 million d'images avec 1000 classes [Krizhevsky *et al.*, 2012]. Sermant *et al.*[Sermanet *et al.*, 2013] ont proposé une méthode (dite OverFeat) d'extraction de primitives "puissanttes" pour la communauté de recherche dans le domaine de la vision par ordinateur. Ces primitives sont entièrement et directement entrainées à partir des pixels de l'image. Sermant et ses collègues ont présenté également une approche basée sur le balayage multi-échelles par une fenêtre fixe qui peut être utilisée pour la classification, la localisation et la détection d'objets dans des images. Ils ont montré également que l'apprentissage d'un réseau selon leur approche améliore la précision de toutes les tâches.

Girshick *et al.*[Girshick *et al.*, 2013] ont proposé de calculer un vecteur de primitives CNN fixe pour chaque région de proposition d'échantillon pour localiser et segmenter des objets. Ils ont nommé cette proposition par R-CNN et l'ont comparé avec Overfeat de Sermant *et al.*[Sermanet *et al.*, 2013]. La comparison que la détection de R-CNN dépasse OverFeat. Les auteurs ont trouvé que les CNN donnent des performances de détection d'objets très élevées sur la base Pascal VOC par rapport aux caractéristique HOG. Girshick *et al.*[Girshick *et al.*, 2013] décrivent deux raisons de cette performance de détection. La première est que les paramètres de CNN sont partagées à travers toutes les catégories. La deuxième est que le vecteur de primitives CNN possède une dimension réduite par rapport à d'autres approches telles que le codage de pyramides spatiales de sac de mots (*bag of words*).

Plusieurs travaux [Wang *et al.*, 2014a, Xie *et al.*, 2015, Guan *et al.*, 2016, Bautista *et al.*, 2016] traitent la détection des véhicules en utilisant les CNN. Wang *et al.*[Wang *et al.*, 2014a] ont proposé une architecture deep learning de 2D-DBN qui maintient l'information descrimitive pour la détection de véhicules. Une architecture qui utilise des plans de second ordre pour la détermination de taille de l'architecture profonde et pour augmenter le taux de détection. Bautista *et al.*[Bautista *et al.*, 2016] ont présenté un système pour la détection et la classification de véhicules dans des vidéos. Ils ont montré que les CNN donnent des résultats de détection intéressants dans des vidéos de trafic routier à faible résolution.

3 Choix d'un détecteur pour nos travaux

Le but de notre travail est de proposer une approche générique qui permet de transférer des connaissances acquises précédemment et de sélectionner des échantillons cibles pour la spécialisation d'un détecteur à une scène cible tout en tirant profit de certains indices visuels issus du contexte de la scène étudiée. En ce sens, nous nous sommes intéressés à travailler sur une structure simple afin d'évaluer l'apport de la combinaison du transfert d'apprentissage avec des informations provenant du contexte de la scène pour la spécialisation d'un détecteur à une nouvelle scène.

Nos travaux doivent être modulaires et indépendants du détecteur ou classifieur utilisé. Notre approche doit fournir un détecteur spécialisé avec un temps de spécialisation acceptable et sans exigence d'échantillons étiquetés de la scène cible. De plus, le détecteur doit pouvoir fonctionner en temps réel.

En se basant sur la description des caractéristiques les plus utilisées pour la détection des objets du trafic routier dans la section 2, nous constatons que les histogrammes de gradients orientés sont bien adaptés pour notre cas d'application. Les HOGs ont une bonne puissance discriminatoire et dans [Prisacariu et Reid, 2009] a été montré qu'elles peuvent être utilisées pour la détection temps réel. D'autre part, la plupart des approches de transfert d'apprentissages mentionnées dans la section 5 du chapitre 1 (page 26) ont utilisé ou proposé une variante des SVM.

Le choix de détecteur s'est donc porté sur le HOG-SVM, un détecteur qui présente un bon compromis entre simplicité et rapidité. Le modèle du détecteur HOG-SVM est un modèle monolithique, sans cascade et son parcours dans l'image est basé sur le formalisme de fenêtre glissante [Chesnais, 2013]. En particulier, ce détecteur a été utilisé dans plusieurs travaux [Aytar, 2014, Lim *et al.*, 2011, Aytar et Zisserman, 2011, Dalal et Triggs, 2005, Sun et Watada, 2015, Wang *et al.*, 2014b, Wang et Wang, 2011, Duan *et al.*, 2009, Yang *et al.*, 2007] pour la détection de multiples catégories d'objets tels que les piétons, les vélos, les motos, les voitures,...

Nous insistons sur la nature générique de la méthode de spécialisation que nous proposons. Même si elle est illustrée avec un détecteur HOG-SVM, elle doit être indépendante du type de détecteur.

4 Description du détecteur HOG-SVM

Dans cette section, nous reprenons principalement les étapes de génération d'un détecteur HOG-SVM comme elles ont été décrites dans les travaux de Dalal et Triggs [Dalal et Triggs, 2005] qui proposent un modèle pour la détection de piétons dans une image numérique ou séquence vidéo. La génération du détecteur se résume en deux grandes étapes :(i) Le calcul des caractéristiques HOG des exemples positifs et négatifs de la base d'apprentissage et (ii) L'apprentissage SVM et là récupération du détecteur. Le détecteur obtenu sera utilisé pour le balayage de l'image à la recherche de l'objet d'intérêt.

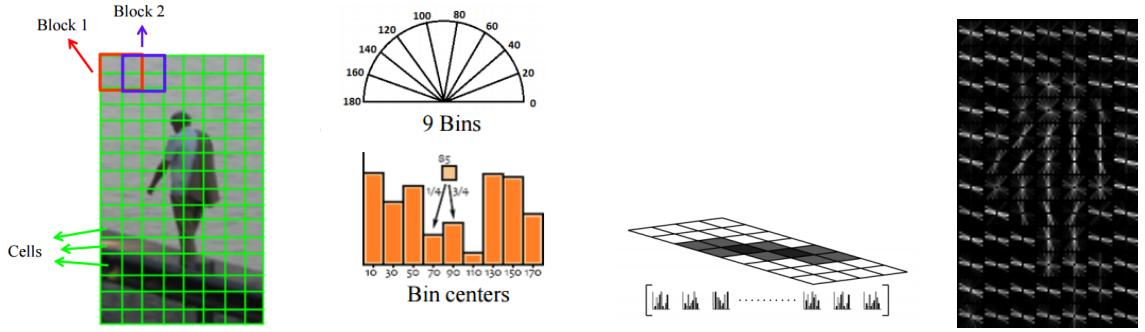
Dans le cas de l'application proposée par Dalal et Triggs [Dalal et Triggs, 2005], la base d'apprentissage est composée d'un ensemble d'imagettes de taille 64×128, les imagettes positives présentent un piéton au centre et un peu de son contour et le contenu des imagettes négatives est variable à condition qu'il soit dépourvu d'une instance de piétons.

4.1 Calcul des caractéristiques HOG

Le Tableau 2 décrit les détails du processus de calcul des caractéristiques HOGd selon les travaux de Dalal et Triggs [Dalal et Triggs, 2005] et la FIGURE 21 illustre certaines instructions du calcul et visualise le vecteur final.

Tableau 2 – Les instructions de calcul des caractéristiques HOG

1- Calcul des gradients horizontaux et verticaux sans lissage 2- Calcul d'orientation et de norme des gradients. (Pour les images couleurs, on prend le canal avec l'amplitude maximale de chaque pixel) 3- Choix de fenêtre à utiliser. 4- Division de la fenêtre en blocks de (16 x 16) avec 50 % de chevauchement. 5- Division de chaque block en cellule de (8 x 8) : donc (2 x 2) cellule par block. 6- Quantification de l'orientation du gradient en 9 angles vote pondéré par le norme du gradient Interpolation des votes de manière bilinéaire entre les angles voisins Le vote peut être pondéré par une gaussienne pour ne pas augmenter le poids des pixels proches des contours des blocks. 7- Concaténation des histogrammes



(a) Division de fenêtre en blocks et cellules (b) Quantification du gradient d'orientation en neuf angles de 20° (0-180) (c) Vecteur final de caractéristiques (d) Visualisation du vecteur

FIGURE 21 – Certaines instructions de calcul des HOG et visualisation du vecteur final [Shah, 2013].

4.2 Apprentissage SVM et récupération du détecteur

Ce paragraphe est fortement inspiré des travaux de Dalal et Triggs [Dalal et Triggs, 2005] et de Jonathan Milgram [MILGRAM, 2007].

La technique de machines à vecteurs de support est une technique très répandue dans les travaux d'état de l'art. C'est une approche discriminante d'apprentissage statistique supervisé qui a été proposée par Vladimir Vapnik en 1995. Dans cette section, nous nous limitons au cas de SVM linéaire. Elle a un avantage majeur qui est la classification linéaire des problèmes non linéairement séparables. Son principe consiste à transformer la représentation des données pour les décrire dans un espace de plus grande dimension et à déterminer l'hyperplan optimal qui permet de séparer les données d'apprentissage de manière linéaire [MILGRAM, 2007].

Dans le cadre de la classification binaire d'objets, un hyperplan optimal est le séparateur qui donne un nombre minimal d'erreurs de classification parmi les données d'apprentissage.

De point de vue purement mathématique, un hyperplan est constitué de l'ensemble de points $\mathbf{x} \in \Re^d$ vérifiant l'équation (2.1) :

$$\mathbf{w}.\mathbf{x} + b = 0 \tag{2.1}$$

La règle de décision consiste à classer la donnée $\mathbf{x}$ dans la première classe, caractérisée par l'étiquette $y = 1$, si $f(\mathbf{x}) > 0$ (avec $f(\mathbf{x}) = \mathbf{w}.\mathbf{x} + b$), ou dans la seconde classe caractérisée par l'étiquette $y = -1$, si $f(\mathbf{x}) \leq 0$.

Soit un ensemble de données d'apprentissage $\{\mathbf{x}_i \in \Re^d, i = 1, \ldots, n\}$ et d'étiquettes $\{y_i \in \{1, -1\}, i = 1, .., n\}$. Si les données des deux classes sont linéairement séparables, il y aura au moins un hyperplan qui vérifie l'équation (2.2) :

$$f(\mathbf{x}_i)y_i \geq 0, (i = 1, \ldots, n) \tag{2.2}$$

Déterminer l'hyperplan optimal pour un ensemble d'apprentissage, c'est calculer les valeurs de $\mathbf{w}$ et b qui vérifient les contraintes (2.2) et maximisent la marge de séparation. La marge est la distance minimale entre les données d'apprentissage et leurs projections sur l'hyperplan. Elle se calcule par l'équation (2.3) :

$$\min_{i=1,..,n} \{\frac{|\mathbf{w}.\mathbf{x}_i + b|}{\|\mathbf{w}\|}\} \tag{2.3}$$

Pour des raisons de simplification ; l'hyperplan est définit par l'équation canonique suivante (2.4) :

$$\min_{i=1,..,n} \{|\mathbf{w}.\mathbf{x}_i + b|\} = 1 \tag{2.4}$$

Autrement dit, la fonction de décision $f(\mathbf{x})$ est égale à 1 pour l'ensemble de vecteurs de support de la première classe et -1 pour les vecteurs de support de la deuxième classe. Et la marge de séparation est égale à $1/(\|\mathbf{w}\|)$. De tout ce qui précède, la détermination de l'hyperplan ramène à un problème d'optimisation qui minimise $\frac{1}{2}\|\mathbf{w}^2\|$ sous les contraintes $y_i(\mathbf{w}.\mathbf{x}_i + b) \geqslant 1, (i = 1, \ldots, n)$ ce qui garantit une solution unique.

5 Est-il nécessaire de faire une spécialisation du détecteur ?

Dans cette section, nous cherchons à répondre à la question "Est-il vraiment nécessaire de faire une spécialisation du détecteur ?" Pour y arriver, nous illustrons les différentes limites des détecteurs génériques lorsqu'ils sont appliqués sur des scènes de surveillance de trafic routier. Puis, nous présentons les inconvénients liés à l'utilisation d'une solution intuitive. Ensuite, nous décrivons certains travaux d'état de l'art qui justifient le choix de faire une spécialisation.

5.1 Limites des détecteurs génériques

Lorsqu'un détecteur générique est appliquée sur les images d'une scène spécifique ses performances sont souvent moins bonnes que l'évaluation de ce détecteur sur une base de test (dont les conditions sont souvent proches de celle de la base d'apprentissage). La cause principale de cette chute est les différences d'apparence entre les échantillons utilisés pour l'apprentissage et ceux de la scène observée. Parmi les différences d'apparence, nous citons :

- Les échantillons positifs de la base générique présentent l'objet d'intérêt avec une taille grande et il se trouve avec une taille beaucoup plus petite dans la nouvelle scène. La situation inverse peut se présenter également.
- La base générique contient généralement un ensemble d'échantillons positifs pris de multiples points de vues alors que la scène spécifique présente un seul point de vue (exemple un point de vue plongeante pour une scène de surveillance de trafic dans une autoroute)
- La base générique présente un ensemble d'échantillons négatifs assez variés mais une scène de vidéo surveillance présente un fond presque stable

(a) Échantillons d'apprentissage de la base INRIA Person dataset (les positifs en haut et les négatifs en bas)

(b) Image de la scène CUHK_square dataset surveillée par une caméra fixe

FIGURE 22 – Différences d'apparence des échantillons de la base générique et de la scène cible

La FIGURE 22 illustre les différences entre les échantillons d'apprentissage et les échantillons de la scène surveillée. Le Tableau 3 montre certaines situations d'échec d'un détecteur entrainé sur la base publique INRIA Person dataset et appliqué sur la scène de CUHK_square dataset (le détail de cette base est donnée dans la paragraphe 6.2 page 50). Les rectangles verts représentent les détections rendues par le détecteur et les flèches rouges signalent ses échecs.

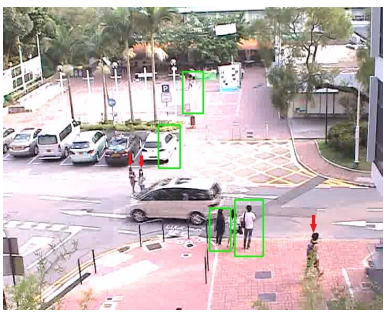
5.2 Inconvénients de la solution intuitive

Pour échapper à la plupart des limites citées dans le Tableau 3, une solution intuitive peut être adoptée. Elle consiste à construire un détecteur spécialisé à la scène en utilisant uniquement des échantillons étiquetés de la scène elle même. Ce détecteur doit avoir une performance supérieure à celle du détecteur générique. Par contre, l'étiquetage manuel des échantillons à chaque scène et la répétition du processus d'apprentissage pour toutes les classes d'objet que l'on souhaite détecter dans la scène, sont des tâches ardues et qui prennent beaucoup de temps.

D'autre part, il est nécessaire de disposer de vidéos de la scène cible couvrant : (i) une grande variabilité en termes de conditions météorologiques et (ii) une variabilité suffisante des objets du trafic routier à annoter.

Ces inconvénients peuvent être évités grâce à une spécialisation d'un détecteur générique vers la scène cible. Elle repose sur un étiquetage automatique des données de la scène et / ou un transfert de données génériques étiquetées, déjà disponibles, et utiles pour apprendre un détecteur spécialisé à la nouvelle scène étudiée. C'est le type de solution visée dans cette thèse. Dans la section suivante nous décrivons deux travaux d'état de l'art qui ont proposé des approches de spécialisation.

Tableau 3 – Exemples des limites des détecteurs génériques

Illustration de situations d'échec d'un détecteur générique de piétons	
Beaucoup de détections ratées : Une bonne détection, dite aussi un vrai positif, est une détection rendue par le détecteur et contient l'objet d'intérêt. Par contre, une détection ratée c'est un cas où une instance de l'objet recherché est bien présente dans l'image, mais le détecteur n'arrive pas à la détecter. L'image ci-contre présente beaucoup d'exemples.	
Nombre de faux positif élevé : Un faux positif est une détection rendue par le détecteur mais elle ne contient pas l'objet recherché. Elle peut contenir une partie du fond de la scène ou un autre type d'objet que l'objet d'intérêt recherché.	
Mauvaise localisation : Une mauvaise localisation se produit lorsque beaucoup d'instances de l'objet cherché sont très proches les unes des autres. Parmi ces situations de mauvaise localisation, l'image ci-contre montre deux situation de mauvaise localisation. La première situation de deux instances de l'objet sont détectés comme un seul objet (marquée par une flèche rouge). La deuxième situation indique deux objets détectés au lieu de trois(l'objet non détecté est marquée par la flèche bleue).	
Objet partiellement occulté : Un détecteur générique n'arrive pas à détecter une instance de l'objet recherché lorsque cette instance est partiellement occultée par un autre objet ou par un élément de fond. L'image à droite présente deux exemples de situations ; deux piétons occultés partiellement par une voiture et un piéton occulté par une mince branche d'arbre.	
Détections parasites : Une détection parasite est une détection qui englobe une instance de l'objet recherché ou une partie de l'instance et qui se chevauche beaucoup avec une autre détection (parfois se localise totalement à l'intérieur d'une autre détection).	

5.3 Travaux existants de spécialisation

L'objectif principal de la spécialisation est de fournir un détecteur spécialisé à une scène particulière avec le minimum d'intervention manuelle. La spécialisation est désignée par plusieurs termes dans la littérature tels que adaptation, contextualisation, transfert d'apprentissage,...

Thierry Chesnais a présenté une approche de spécialisation dans sa thèse [Chesnais, 2013] sous le nom "contextualisation d'un détecteur par un oracle". Son approche se repose sur deux points l'extraction de l'information provenant de la scène et sa prise en compte dans le détecteur pour améliorer ses performances. L'application de ses travaux est présentée sous forme d'un système contextualisé complet de détection de piétons. Ce système crée automatiquement une base d'apprentissage contextualisée à l'aide d'un oracle. Puis il entraîne un détecteur spécialisé à la scène sur la base créée. La FIGURE 23 présente le schéma bloc de son système contextualisé à base d'un oracle. L'oracle est constitué de trois classifieurs indépendants basés sur l'apparence, la segmentation fond/forme et le flot optique. Chacun fournit un ensemble de détections qui doivent être fusionnées pour former une base d'apprentissage contextualisée. Le détecteur contextualisé est entraîné sur la base formée.

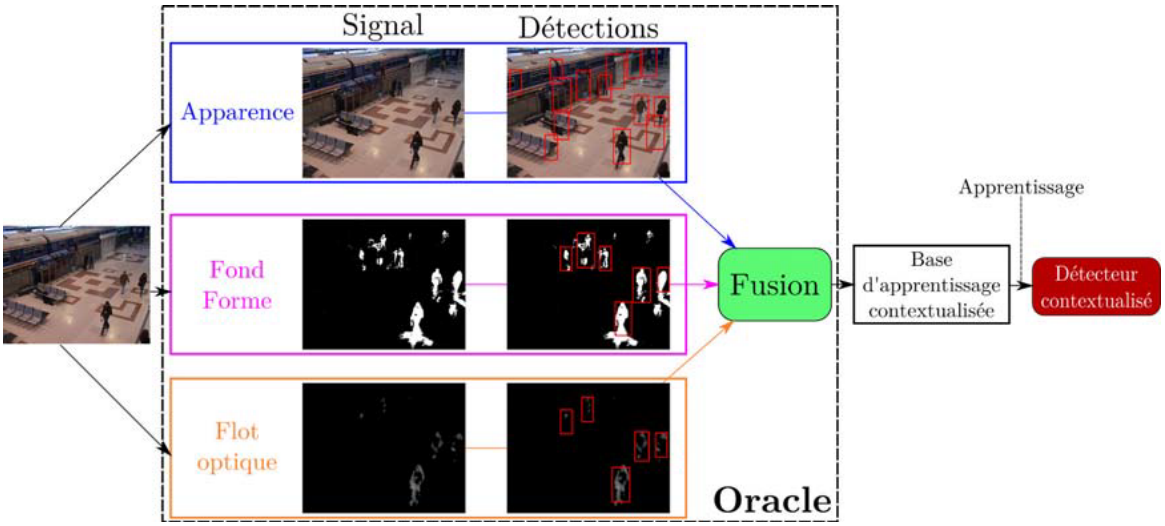


FIGURE 23 – Schéma de fonctionnement du système contextualisé à base d'un oracle [Chesnais, 2013].

Deux types d'oracles : un 2D et un autre 3D ont été proposés pour extraire les échantillons de la base contextualisée. Un oracle a pour rôle d'étiqueter les données provenant de la scène.

Pour le premier oracle, la base des exemples positifs est le résultat d'une fusion de trois classifieurs génériques qui sont le classifieur d'apparence, d'extraction fond / forme et de flot optique tout en privilégiant le signal d'apparence. Par contre, la base négative est créée par une sélection aléatoire qui exclut les imagettes prises dans la base positive. Le deuxième oracle est constitué du classifieur d'apparence et de celui d'extraction fond / forme. L'oracle 3D est totalement automatique mais ce dernier nécessite la connaissance des paramètres de la caméra. La calibration réalisée a permis de limiter l'espace de recherche et la réduction du temps de calcul.

La contextualisation proposée spécialise un détecteur à une scène ce qui améliore significativement les performances de ce dernier par rapport à celles d'un détecteur générique. Cependant, il est important de noter que ce détecteur n'est pas parfait car la base créée automatiquement peut contenir des erreurs d'étiquetage. Ainsi, l'approche est basée sur l'hypothèse que les classifieurs composant l'oracle sont indépendants, ce qui n'est pas facile à valider. Une autre lacune de cette solution est qu'elle est limitée à l'utilisation des échantillons extraits de la scène cible uniquement, qui peuvent être très rares et insuffisants pour créer un détecteur spécialisé. Autrement dit, elle ne profite pas d'échantillons de bases sources disponibles et qui peuvent être bénéfiques pour l'apprentissage du détecteur spécialisé.

Une autre solution de spécialisation, dite "Transfert d'un détecteur générique de piétons vers des scènes spécifiques", a été proposée par Meng Wang et ses collaborateurs dans [Wang et Wang, 2011, Wang *et al.*, 2012a, Wang *et al.*, 2014b]. Elle consiste à adapter un détecteur de piétons entrainé sur une base publique à une scène de trafic routier. La méthode commence par un détecteur HOG+SVM créé selon la méthode de Dalal et Triggs et entraîné sur la base INRIA Person dataset (dite aussi base source) [Dalal et Triggs, 2005] qui va extraire des échantillons de la scène étudiée. Ensuite, il y a une étape d'étiquetage des exemples extraits. La dernière étape fait le ré-apprentissage du détecteur spécialisé à la scène à la fois sur des exemples de la scène étudiée et d'autres échantillons de la base source qui présentent une similarité visuelle forte avec les exemples de la scène cible. Une fois le détecteur spécifique ré-entrainé, il est capable de détecter les piétons dans la nouvelle séquence en se basant uniquement sur l'apparence de l'objet d'intêret. Le schéma général de la méthode proposée par Wang *et al.*[Wang *et al.*, 2014b] est illustré par la FIGURE 24.

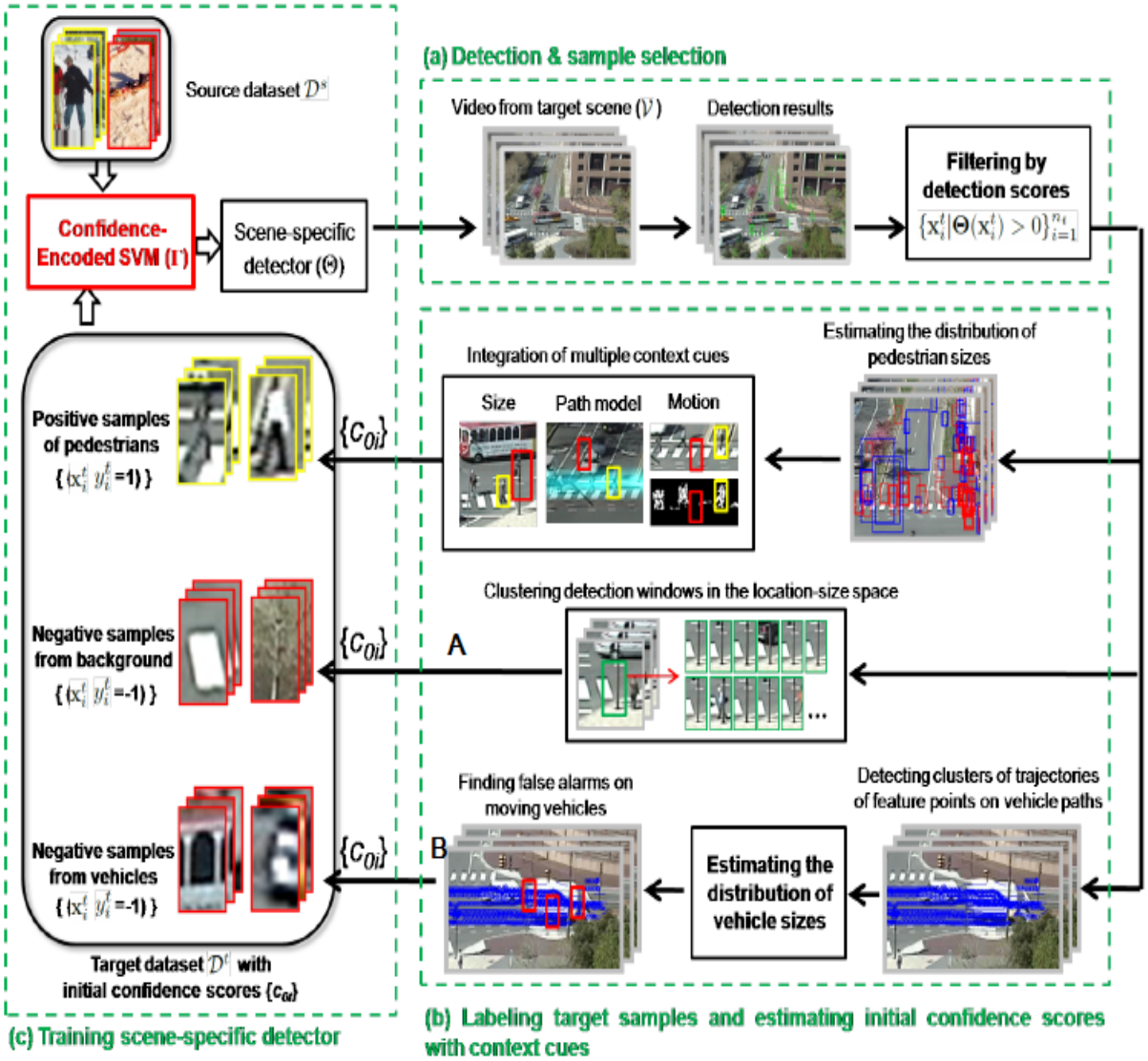


FIGURE 24 – Diagramme de la méthode de transfert d'apprentissage proposée [Wang *et al.*, 2014b]. La méthode a trois modules :(a) détecter et sélectionner des échantillons de la scène cible, (b) l'étiquetage des échantillons cibles et l'estimation de leurs scores de confiance initiaux en utilisant des indices contextuels, et (c) l'apprentissage du détecteur spécifique à la scène.

La méthode se déroule sur plusieurs itérations et s'arrête si le nombre d'itérations fixé au départ est atteint ou bien le nombre d'exemples ajoutés à la base cible au niveau de l'étape 2 se stabilise. Aux premières itérations, la première étape donne une base cible contenant beaucoup de fausses détections (due à la différence entre la base source et la séquence vidéo de test). La deuxième étape cherche à corriger les étiquettes des exemples de la base fournie par l'étape 1 en utilisant certains indices contextuels. Cette étape donne en résultat, trois sous-ensembles : 1) des piétons (exemples positifs) 2) des non piétons extraits du fond (exemples négatifs partie A) et 3) des non piétons extraits de véhicules (exemples négatifs partie B) et calcule le score de confiance pour chaque exemple des trois sous-ensembles. La troisième étape se compose de deux tâches ; la sélection des exemples sources qui ressemblent aux exemples issus de la scène étudiée et l'apprentissage d'un nouveau détecteur avec la base source et la base cible, en tenant compte des scores de confiance attribués aux étiquettes. Ces deux dernières tâches sont formulées dans une seule fonction objectif dite « Confidence –Encoded SVM ». Cette étape permet de mettre à jour les paramètres du SVM $(\mathbf{W}, b)$.

L'étiquetage effectué au cours de l'étape 2 se base principalement sur la division spatiale de la scène en "chemins piétons" et un autre "chemins voiture". Pour obtenir des exemples positifs avec une grande confiance, il y a une sélection des détections qui se localisent dans le "chemin piéton". Cette sélection est par la suite filtrée par d'autres indices contextuels telle que l'estimation de la taille des blobs de piétons, la répétition des détections dans un même emplacement, et le taux de pixels en mouvement à l'intérieur de la détection. Une partie A des exemples négatifs est extraite à partir des détections de fond qui ont un score de détection positif et inférieur à 0.5. Ce sont des fausses détections avec la même apparence qui se répètent au cours du temps dans la même position. Une partie B des exemples négatifs est formée des détections localisées dans le chemin de véhicule. Une détection est classée en un véhicule ou une partie de véhicule en se basant sur le suivi des points d'intérêt et le clustering des détections en fonction de la taille de détection et du mouvement des pixels à l'intérieur. Le dernier clustering permet d'éliminer les piétons qui se détectent accidentellement dans le chemin véhicule. La possibilité d'attribuer des étiquettes erronées est encore possible, mais l'effet des fausses étiquettes est réduit grâce à l'étape 3 qui transfère des échantillons sources correctement étiquetés et présentant une forte similarité visuelle avec la plupart des échantillons cibles sélectionnés. Les échantillons sources sont pondérés en fonction de similarité aux échantillons cibles et les poids attribués sont dits des coefficients de confiance. Ensuite, un nouveau détecteur est fourni suite à l'apprentissage d'un SVM sur toutes les données sources et cibles. La version confidence-encoded SVM traite les échantillons en fonction de leurs coefficients de confiance associés et pénalise gravement une erreur de classification effectuée sur un échantillon possédant un coefficient élevé.

La comparaison des performances du détecteur spécialisé selon la méthode proposée par Wang *et al.*[Wang *et al.*, 2012a] a montré un gain de 48% et de 36%, respectivement sur la base MIT Traffic dataset et CUHK_square dataset, par rapport aux performances du détecteur générique sur les deux scènes de trafic routier. Mais, la méthode se limite au classifieur de type SVM puisqu'elle agit sur la fonction de coût. Si nous l'appliquons à un autre classifieur, il faudra modifier la fonction d'apprentissage du classifieur pour traiter les échantillons d'apprentissage en fonction de leur poids.

L'analyse et l'interprétation des résultats de ces deux travaux, permettent de répondre par oui à la question "Est-il vraiment nécessaire de faire une spécialisation ?". Ainsi, elles mettent en évidence les avantages d'une spécialisation par transfert d'apprentissage d'un détecteur à une scène particulière de vidéo-surveillance de trafic routier. Nous pouvons citer l'utilisation des connaissances antérieures, l'étiquetage automatique des données de la scène, et l'amélioration de la performance du détecteur spécialisé. Malgré cela, les inconvénients de ces travaux compliquent et troublent l'utilisation industrielle de ces solutions. Nous présenterons, dans le chapitre 3, une solution alternative destinée à une adoption industrielle. C'est une approche à base de filtre séquentiel de Monte Carlo pour la spécialisation d'un détecteur générique vers une scène particulière de circulation routière ; 1) Elle utilise des stratégies d'observation pour l'étiquetage automatique des données extraites de la scène cible. 2) Elle

profite des données sources disponibles qui présentent une similarité visuelle aux échantillons cibles sélectionnés. 3) Elle est générique permettant l'intégration de différentes stratégies d'observation et l'utilisation de tout type de classifieur sans besoin de modification de la fonction d'apprentissage du classifieur tout en tenant en compte du poids d'importance de chaque échantillon.

6 Évaluation d'un détecteur d'objet

Nous présentons ici l'outil d'évaluation le plus utilisé en détection d'objets (la courbe ROC), ainsi que les bases de données utilisées par la suite.

6.1 Courbe ROC

La courbe ROC (Receiver Operator Characteristic) est une courbe qui représente le taux de bonnes détections en ordonnée en fonction de taux de fausses détections en abscisse. Une bonne détection (dite aussi vrai positif ou *True positive* en anglais), est une détection qui contient et englobe correctement l'objet d'intérêt. Le taux de bonnes détections (ou taux de vrais positifs (TVP)) représente le nombre d'instances de l'objet recherché qui sont détectées par rapport au nombre total d'instances contenues dans la base de test. Une fausse détection (dite aussi faux positif ou *False positive* en anglais), est une détection qui ne contient pas l'objet d'intérêt ou qui englobe seulement une petite portion de ce dernier. Le taux de faux positifs correspond au nombre de mauvaises détections par rapport au nombre de fenêtres testées. La FIGURE 25 présente un exemple de la courbe ROC et des différentes situations possibles. Un détecteur capable de détecter toutes les instances de l'objet recherché, sans ou avec un faible taux de faux positifs c'est une situation de "Discrimination parfaite". La situation "Pas de discrimination" c'est quand le détecteur répond de manière aléatoire. Une fois, il donne une réponse correcte et une autre il donne une réponse fausse. Détecteur 1 et Détecteur 2 présentent des exemples d'allures possibles de courbe ROC dans des situations réelles.

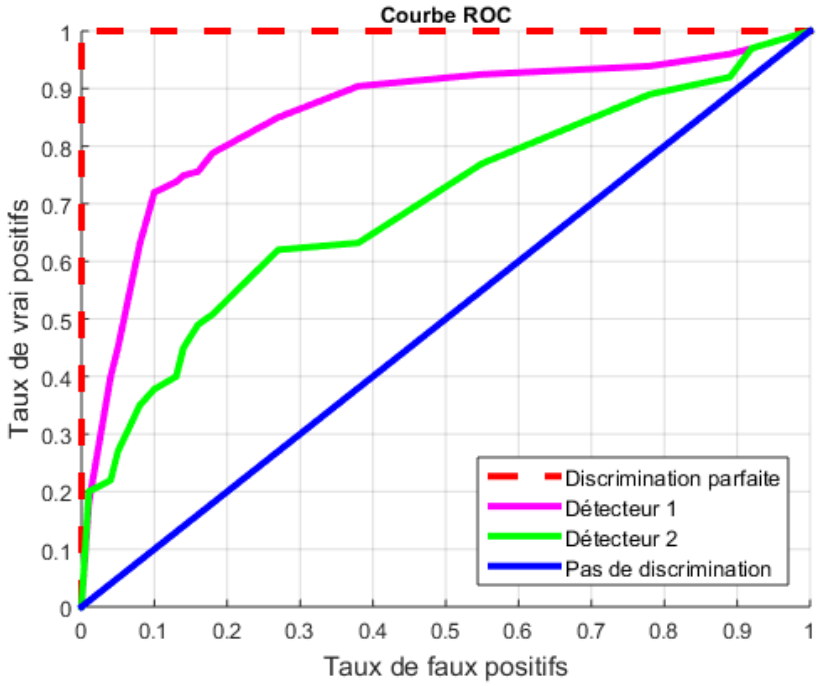


FIGURE 25 – La courbe ROC et les différents cas possibles

Nous comparons les performances des détecteurs par rapport au même nombre d'images de la base de test. C'est pourquoi nous pouvons remplacer le taux de fausses détections par le nombre de faux positifs par image (connu en anglais par *False Positive Per Image (FPPI)*). Ainsi, le couple (FPPI, TVP) représente le point de fonctionnement du détecteur. Ces deux critères sont liés. En effet, augmenter (resp. diminuer) le taux de vrais positifs implique généralement d'augmenter (resp. de diminuer) le nombre de faux positifs par image. Deux détecteurs sont comparés par rapport à un ensemble de points de fonctionnement.

L'idée alors est de faire varier le seuil d'admission de bonne détection et pour chaque valeur de ce seuil, le TVP et le FPPI sont calculés et reportés dans un graphique sous forme d'une courbe reliant l'un ensemble des points de fonctionnement. Ainsi, l'ordonnée d'une courbe ROC sera compris entre 0 et 1 tandis que l'abscisse sera compris entre 0 et N. Un détecteur est classé plus performant qu'un autre si sa courbe ROC est au-dessus de l'autre courbe ROC alors qu'un détecteur est dit parfait s'il a comme point de fonctionnement (0, 1). Par exemple, dans la FIGURE 25 nous pouvons déduire que le "Détecteur 1" est plus performant que "Détecteur 2".

Critère de validation d'une bonne détection : Une détection sera acceptée comme une bonne détection, si le chevauchement entre la fenêtre de détection et le rectangle de vérité terrain dépasse 0.5 de leur réunion. Sinon, la détection est classée fausse détection. C'est la règle PASCAL [Everingham *et al.*, 2010] que nous adoptons pour calculer le taux de vrais positifs et établir les courbes ROC pour comparer les performances des détecteurs.

Suite à une détection sur une base de test, nous aurons les sorties suivantes [Goutte et Gaussier, 2005] :

— Vrai positif (VP) : C'est une détection rendue par le détecteur et englobe une instance de l'objet recherché. C'est à dire le critère de validation de bonne détection est vrai. Il est à noter que dans le cas où deux détections englobent la même instance d'objet, alors on considère comme vrai positif la détection qui présente un chevauchement maximal avec la vérité terrain. La deuxième détection ne sera pas compté si le critère de validation est valide. Sinon, elle sera considérée comme Faux positif.

— Faux positif (FP) : C'est une détection rendue par le détecteur mais ne contient pas l'objet recherché. Autrement dit, c'est le cas ou le critère de validation de bonne détection est faux.

— Vrai négatif (VN) : C'est une fenêtre ou région qui ne présente pas l'objet et le détecteur a répondu correctement à cette région. Il est important de signaler qu'il existe plusieurs fenêtres négatives.

— Faux négatif (FN) : C'est une fenêtre ou région qui présente une instance de l'objet recherché mais le détecteur n'a pas rendu cette instance.

A partir de ces sorties, nous calculons la précision **P**, le rappel **R** et la **F-Score**.

— La Précision **P** : C'est la capacité de détecteur à détecter l'objet recherché et à ne pas générer de faux positifs. C'est la proportion des bonnes détections par rapport toutes les détections rendues (2.5).

$$\mathbf{P} = \frac{VP}{VP + FP} \tag{2.5}$$

— Le Rappel **R** : C'est la capacité de détecteur à détecter toutes les instances de l'objet recherché disponibles dans la base de test. C'est la proportion des bonnes détections par rapport à toutes les instances de l'objet (2.6).

$$\mathbf{R} = \frac{VP}{VP + FN} \tag{2.6}$$

— **F-Score** ou **F-mesure** c'est une mesure qui représente la moyenne harmonique de la précision et du rappel (2.5). C'est la capacité de détecteur à détecter toutes les instances de l'objet et de ne pas générer des fausses détections.

$$\textbf{F-Score} = \frac{2\mathbf{PR}}{\mathbf{P} + \mathbf{R}} \tag{2.7}$$

6.2 Bases de données

Nous décrivons les bases de données utilisées dans nos travaux :

— **INRIA Person Dataset [Dalal et Triggs, 2005] :** La base de données est divisée en deux ensembles : des images positives originales avec les annotations correspondantes et des images négatives originales. De plus, 2416 imagettes positives, au format 64×128 pixels (tels qu'elles sont utilisées dans [Dalal et Triggs, 2005]), normalisées sont disponibles. Pour générer des imagettes négatives pour l'apprentissage, un ensemble fixe de 12180 fenêtres (10 fenêtres par image négative) est extrait aléatoirement à partir de 1218 images négatives originales. La base INRIA est la base source utilisée dans cette thèse. La FIGURE 22a présente un exemple de ses échantillons.

— **CUHK_Square dataset [Wang *et al.*, 2012a] :** Il s'agit d'une séquence de vidéosurveillance de 60 minutes, enregistrant une scène de trafic routier par une caméra stationnaire. Nous avons extrait uniformément (comme il a été proposé dans [Wang *et al.*, 2012a]) 452 images de cette vidéo, dont les 352 premières images ont été utilisées pour la spécialisation et les 100 dernières images ont été utilisées pour le test. La FIGURE 26a montre un exemple de cette base.

— **MIT Traffic dataset [Wang *et al.*, 2009] :** Une caméra statique a été utilisée pour enregistrer un ensemble de 20 courtes séquences de vidéo, la durée de chacune est de 4 minutes et 36 secondes. A partir des 10 premières vidéos, nous avons extrait 420 images pour la spécialisation. Et à partir des 10 dernières vidéos, 100 images ont été extraites pour le test. La FIGURE 26b montre une image de la scène MIT.

— **Logiroad Traffic dataset :** Il s'agit d'un enregistrement d'une scène de circulation routière, qui a été réalisée par une caméra stationnaire de 20 minutes. Le même raisonnement a été appliqué ; nous avons extrait de façon uniforme 700 images de cette vidéo, dont les 600 premières images ont été utilisées pour la spécialisation et les 100 dernières images ont été utilisées pour le test. La FIGURE 26c illustre une image de la scène Logiroad.

Les bases CUHK_Square, MIT Traffic, et Logiroad Traffic dataset sont les trois scènes cibles de circulation routière qui sont utilisées pour évaluer notre approche de spécialisation.

Conclusion

Dans ce chapitre, nous avons défini l'analyse automatique d'une scène de trafic routier dans la première section. Ensuite, nous avons décrit les avantages et les inconvénients des primitives les plus utilisées pour la détection d'objets dans une scène de circulation routière.

Dans un deuxième temps, nous avons présenté le détecteur retenu pour les expérimentations et la validation de nos travaux.

Ensuite, nous avons montré la nécessité de faire une spécialisation via la présentation des limites des détecteurs génériques, et les inconvénients de la solution intuitive et de certains travaux existants. Dans la dernière section du chapitre, nous avons présenté la courbe ROC qui sera utilisée comme outil d'évaluation des performances d'un détecteur d'objet et les différentes bases de données utilisées dans nos expérimentations.

Dans le chapitre suivant, nous détaillons la contribution principale de cette thèse : la spécialisation par filtrage de Monte Carlo.

(a) Image de la scène CUHK_square dataset

(b) Image de la scène MIT Traffic dataset

(c) Image de la scène Logiroad Traffic dataset

FIGURE 26 – Base de données cibles

Deuxième partie

Contributions

En vue de bien positionner les travaux de cette thèse, nous avons dressé un état de l'art dans la partie précédente sur les méthodes de transfert d'apprentissage et sur les techniques de détection d'objets tout en considérant l'application d'analyse des scènes du trafic routier ciblée par nos travaux.

Dans cette partie, nous présentons les principales contributions de cette thèse. Le chapitre 3 a pour objectif de présenter notre approche de transfert d'apprentissage à base de filtre Monte Carlo pour la spécialisation d'un classifieur/détecteur entrainé sur une base générique vers une scène particulière. Nous rappelons tout d'abord le principe du filtre SMC dans un cadre général. Par la suite, nous nous intéressons à détailler notre proposition d'utiliser le SMC dans un cadre de transfert d'apprentissage via la description de chaque étape du filtre.

Nous présentons dans le chapitre 3 que la base spécialisée fournie par l'étape de prédiction contient un sous-ensemble d'échantillons qui est collecté automatiquement. Dans ce dernier sous-ensemble, chaque échantillon est dupliqué, il est associé une fois à l'étiquette positive et une autre à l'étiquette négative. Afin de choisir l'échantillon associé à la bonne étiquette nous avons essayé deux stratégies d'observation pour pondérer les échantillons de ce sous-ensemble que nous exposons dans le chapitre 4. La première stratégie d'observation, dite d'indices spatio-temporels OAS, calcule deux scores pour chaque échantillon puis elle lui affecte un poids en fonction de son étiquette et des scores calculés. La deuxième stratégie, dite stratégie de suivi KLT, attribue un poids à chaque échantillon en fonction du nombre et de la nature des points d'intérêts localisés à l'intérieur de la ROI associée à l'échantillon.

Chapitre 3

La spécialisation d'un classifieur à base d'un filtre séquentiel de Monte Carlo

Introduction

Dans ce chapitre, nous proposons une approche de transfert d'apprentissage transductif basée sur un filtre Séquentiel de Monte Carlo (SMC) pour spécialiser un classifieur générique à une scène cible spécifique. Cette approche estime itérativement la distribution cible comme étant un ensemble d'échantillons d'apprentissage. Ces derniers sont sélectionnés à la fois à partir de la source et des échantillons issus de la scène cible en fonction de leurs poids d'importance qui indiquent qu'ils appartiennent à la distribution cible estimée.

Nous donnons dans la première section de ce chapitre un rappel sur le filtre SMC utilisé généralement pour le suivi d'objets. Dans la deuxième section, nous décrivons en détails notre proposition de transfert d'apprentissage à base de filtre SMC. Nous définissons les notations utilisées et le principe de la méthode. Ensuite, nous détaillons le processus de chaque étape du filtre.

1 Rappel sur le filtre séquentiel de Monte Carlo

Le filtre Séquentiel de Monte Carlo, connu également par le nom "Filtre à particules", est une technique largement utilisée pour la résolution des problèmes de suivi d'objet dans des séquences vidéo et la localisation d'un robot mobile autonome [Isard et Blake, 1998, Smal *et al.*, 2007, Mei et Ling, 2011]. Le filtre SMC permet d'estimer une distribution de probabilités à l'aide d'un filtre Bayésien. La distribution est approchée par un ensemble d'échantillons (dits aussi particules) avec des poids associées. Chaque particule représente un état potentiel du système.

L'idée générale de l'approche Bayésienne récursive est d'estimer l'état courant d'un système $\mathbf{X}_k$, sachant toutes les observations précédentes soit $\mathbf{Z}_{1:k}$. Autrement dit, c'est déterminer sa distribution *a posteriori* $p(\mathbf{X}_k|\mathbf{Z}_{1:k})$ qui représente la densité de probabilité de l'état $\mathbf{X}_k$ connaissant les observations $\mathbf{Z}_{1:k}$. A chaque instant k, il y a une mise à jour de la distribution à l'aide des observations.

Cette distribution *a posteriori* peut être écrite en utilisant le théorème de Bayes à l'aide de l'équation (3.1).

$$p(\mathbf{X}_k|\mathbf{Z}_{1:k}) = \frac{p(\mathbf{Z}_k|\mathbf{X}_k)\ p(\mathbf{X}_k|\mathbf{Z}_{1:k-1})}{p(\mathbf{Z}_k|\mathbf{Z}_{1:k-1})} \tag{3.1}$$

Où $p(\mathbf{Z}_k|\mathbf{Z}_{1:k-1})$ est un facteur de normalisation indépendant de l'état $\mathbf{X}_k$, $p(\mathbf{Z}_k|\mathbf{X}_k)$ est le terme de vraisemblance et $p(\mathbf{X}_k|\mathbf{Z}_{1:k-1})$ est la prédiction.

Supposons que nous sommes dans le cadre d'un processus markovien pour lequel l'état $\mathbf{X}_k$ dépend uniquement de l'état précédent $\mathbf{X}_{k-1}$, l'équation (3.1) devient (3.2) :

$$p(\mathbf{X}_k|\mathbf{Z}_{1:k}) \propto p(\mathbf{Z}_k|\mathbf{X}_k) \int_{\mathbf{X}_{k-1}} p(\mathbf{X}_k|\mathbf{X}_{k-1})\ p(\mathbf{X}_{k-1}|\mathbf{Z}_{1:k-1}) d\mathbf{X}_{k-1} \tag{3.2}$$

Par la suite et en supposant que $p(\mathbf{X}_{k-1}|\mathbf{Z}_{1:k-1})$ est disponible à l'instant $(k-1)$, l'étape de prédiction s'engage pour obtenir la distribution *a priori* $p(\mathbf{X}_k|\mathbf{Z}_{1:k-1})$ via l'équation de Chapman-Kolmogorov (3.3) [Maskell et Gordon, 2001] :

$$p(\mathbf{X}_k|\mathbf{Z}_{1:k-1}) = \int_{\mathbf{X}_{k-1}} p(\mathbf{X}_k|\mathbf{X}_{k-1})\ p(\mathbf{X}_{k-1}|\mathbf{Z}_{1:k-1}) d\mathbf{X}_{k-1} \tag{3.3}$$

Où $p(\mathbf{X}_{k-1}|\mathbf{Z}_{1:k-1})$ représente la distribution résultante des calculs de l'instant précédent et $p(\mathbf{X}_{k+1}|\mathbf{X}_k)$ est la prédiction ou la dynamique du système.

En posant $C = p(\mathbf{Z}_k|\mathbf{Z}_{1:k-1})$ une constante, le remplacement de (3.3) dans (3.1) permet de formaliser la distribution Bayésienne récursive comme suit (3.4) :

$$p(\mathbf{X}_k|\mathbf{Z}_{1:k}) = C^{-1}\ p(\mathbf{Z}_k|\mathbf{X}_k) \times \int_{\mathbf{X}_{k-1}} p(\mathbf{X}_k|\mathbf{X}_{k-1})\ p(\mathbf{X}_{k-1}|\mathbf{Z}_{1:k-1}) d\mathbf{X}_{k-1} \tag{3.4}$$

A chaque instant et lorsqu'il y a un ensemble de nouvelles observations, le filtre Bayésien calcule la distribution à l'aide de l'équation (3.4).

Pour calculer une approximation de cette distribution, dans un domaine discret, la technique de Monte Carlo est utilisée. Soit $\mathbf{X}^n$ un échantillon de la distribution, $\pi_k^n \in [0,1]$ le poids associé à l'échantillon n à l'instant k et N le nombre total d'échantillons. Il est à noter que la somme des poids de tous les échantillons est égale à un (3.5).

$$\sum_{n=1}^{N} \pi_k^n = 1 \tag{3.5}$$

La technique de Monte Carlo approxime la distribution $p(\mathbf{X}_{k-1}|\mathbf{Z}_{1:k-1})$ à l'instant $(k-1)$ par un ensemble d'échantillons pondérés $\{\mathbf{X}_{k-1}^n, \pi_{k-1}^n\}_{n=1}^N$ selon l'équation (3.6) [Maskell et Gordon, 2001] :

$$p(\mathbf{X}_{k-1}|\mathbf{Z}_{1:k-1}) \approx \sum_{n=1}^{N} \pi_{k-1}^n\ \delta_{\mathbf{X}_{k-1}^n}(\mathbf{X}_{k-1}) \tag{3.6}$$

Où δ est la distribution de Dirac et les $\mathbf{X}_k^n$ sont les échantillons en posant $p(\mathbf{X}_k|\mathbf{X}_{k-1} = \mathbf{X}_{k-1}^n)$

Supposant que nous disposons de (3.6), alors après certaines simplifications l'équation (3.4) devient (3.7) :

$$p(\mathbf{X}_k|\mathbf{Z}_{1:k}) \approx \sum_{n=1}^{N} \pi_k^n\ \delta_{\mathbf{X}_k^n}(\mathbf{X}_k) \tag{3.7}$$

avec

$$\pi_k^n = \frac{\pi_{k-1}^n\ p(\mathbf{Z}_k|\mathbf{X}_k = \mathbf{X}_k^n)}{\sum_{n=1}^{N} \pi_{k-1}^n\ p(\mathbf{Z}_k|\mathbf{X}_k = \mathbf{X}_k^n)} \tag{3.8}$$

La littérature présente plusieurs variantes du filtrage particulaire. L'algorithme 1 rappelle une variante appelée CONDENSATION qui est proposée par Isard et Blake [Isard et Blake, 1998]. Nous invitons les lecteurs intéressés à consulter [Doucet *et al.*, 2001] et [Smith, 2007] qui présentent plus de détails concernant le filtre séquentiel de Monte Carlo.

Algorithme 1 Le filtrage particulaire CONDENSATION

`/* Étape 0:` *Initialisation* `*/`

Génération de N particules $\{\mathbf{X}_0^{(n)}, \pi_0^{(n)}\}_{n=1,..,N}$ et poser $\pi_0^{(n)} = \frac{1}{N}$

Pour $k = 1$ à T **faire**

`/* Étape 1:` *Prédiction* `*/`

Génération de N particules $\{\mathbf{X}_k^{'(n)}\}_{n=1,..,N}$ selon $p(\mathbf{X}_k|\mathbf{X}_{k-1} = \mathbf{X}_{k-1}^{(n)})$

`/* Étape 2:` *Mise à jour* `*/`

Mise à jour des poids des particules prédites en utilisant :

$$\pi_k^{(n)} = \frac{\pi_{k-1}^{(n)} p(\mathbf{Z}_k|\mathbf{X}_k = \mathbf{X}_k^{'(n)})}{\sum_{n=1}^{N} \pi_{k-1}^{(n)} p(\mathbf{Z}_k|\mathbf{X}_k = \mathbf{X}_k^{'(n)})} \tag{3.9}$$

`/* Étape 3:` *Ré-échantillonnage* `*/`

Sélection de $\{\mathbf{X}_k^{''(n)}, \pi_k^{(n)}\}_{n=1,..,N}$ à partir de $\{\mathbf{X}_k^{'(n)}, \pi_k^{(n)}\}_{n=1,..,N}$

Tirage par importance avec remise de N particules. Une particule avec un poids élevé peut être sélectionnée plusieurs fois. Par contre, une particule avec un poids faible est sélectionnée peu de fois.

Nous aurons $\{\mathbf{X}_k^{(n)}, 1/N\}_{n=1,..,N}$

$E[\mathbf{X}_k] = \sum_{n=1}^{N} \pi_k^{(n)} \mathbf{X}_k^{''(n)}$

FinPour

2 Transfert d'apprentissage transductif à base d'un filtre SMC

Dans cette section, nous présentons l'approche proposée qui concerne l'utilisation de filtre SMC dans un contexte de transfert d'apprentissage transductif. Dans notre approche, nous raisonnons par rapport à une itération k à la place d'un instant k.

2.1 Notations et définitions

Notre travail se base essentiellement sur l'hypothèse que la distribution conjointe inconnue entre les échantillons cibles et les étiquettes associées peut être approchée par un ensemble d'échantillons représentatifs.

Nous notons par :

— $\mathcal{D}_k \doteq \{\mathbf{X}_k^{(n)}\}_{n=1,..,N}$ la base spécialisée à l'itération k de taille N, où $\mathbf{X}_k^{(n)} \doteq (\mathbf{x}^{(n)}, y^{(n)})$ est l'échantillon numéro n avec $\mathbf{x}$ son vecteur de primitives et y son étiquette associée avec $y \in \mathcal{Y}$ (dans un cas de détection mono-objet $\mathcal{Y} = \{-1; 1\}$, 1 représente l'objet et -1 représente son absence).

— $\Theta_{\mathcal{D}_k}$ un classifieur spécialisé à l'itération k et entrainé sur la base spécialisée construite à l'itération $(k-1)$. Un classifieur associe une étiquette y à un vecteur de primitives $\mathbf{x}$. Nous utilisons un classifieur générique Θ_g à la première itération.

Nous disposons d'une base source $\mathcal{D}^s \doteq \{\mathbf{X}^{s(n)}\}_{n=1,..,N^s}$ de N^s échantillons étiquetés. Nous disposons également d'une base cible $\mathcal{D}^t \doteq \{\mathbf{x}^{t(n)}\}_{n=1,..,N^t}$ qui peut être de grande dimension mais composée uniquement par N^t échantillons non étiquetés qui proviennent d'une vidéo de la scène cible. Cette base est issue de l'application d'une stratégie de balayage multi-échelles par fenêtre fixe sur la vidéo cible.

La distribution cible est approchée par un ensemble d'échantillons de la base spécialisée. Ces échantillons sont initialement inconnus. Cependant, ils peuvent être déterminés à l'aide d'un processus d'observation issu de la séquence vidéo et des *a priori* sur la scène cible.

La mise en œuvre d'un filtre à particule nécessite la définition d'un vecteur d'état, d'un modèle d'observation et d'un modèle d'évolution qui exprime l'état d'un système à un instant donné en fonction de son état à l'instant précédent. Nous notons $\mathbf{X}_k$ un vecteur d'état aléatoire caché présentant une distribution conjointe entre les primitives de la base cible et les étiquettes associées à une itération k. Soit $\mathbf{Z}_k$ un vecteur aléatoire de mesure couplé à $\mathbf{X}_k$ et présentant une observation extraite de la séquence vidéo cible.

De cette façon, la distribution cible peut être approchée en appliquant de manière itérative l'approximation particulaire (3.10) :

$$p(\mathbf{X}_{k+1}|\mathbf{Z}_{0:k+1}) \approx \sum_{n=1}^{N} \pi_{k+1}^{n} \ \delta_{\mathbf{X}_{k+1}^{(n)}}(\mathbf{X}_{k+1}) \tag{3.10}$$

avec

$$\pi_{k+1}^{n} = \frac{\pi_k^n \ p(\mathbf{Z}_{k+1}|\mathbf{X}_{k+1} = \mathbf{X}_{k+1}^{(n)})}{\sum_{n=1}^{N} \pi_k^n \ p(\mathbf{Z}_{k+1}|\mathbf{X}_{k+1} = \mathbf{X}_{k+1}^{(n)})} \tag{3.11}$$

Dans notre cas, la méthode séquentielle de Monte Carlo qui approxime la distribution *a priori* $p(\mathbf{X}_k|\mathbf{Z}_k)$ par un jeu de N particules (échantillons dans notre cas) est écrite selon l'équation (3.12) :

$$p(\mathbf{X}_k|\mathbf{Z}_k) \approx \frac{1}{N} \sum_{n=1}^{N} \delta(\mathbf{X}_k^{(n)}) \approx \{\mathbf{X}_k^{(n)}\}_{n=1,..,N} \tag{3.12}$$

Le filtre SMC est utilisé pour estimer la distribution cible inconnue donc pour sélectionner les échantillons de l'ensemble d'apprentissage. Nous considérons les échantillons comme étant les particules du filtre et les informations extraites de la scène comme étant la fonction d'observation qui affecte un poids à chaque particule. Nous supposons que le processus de récursivité permet de mieux sélectionner les échantillons de la base spécialisée au fil des itérations donc de converger vers la vraie distribution cible et les classifieurs entrainés doivent être de plus en plus performants.

L'équation de récurrence (3.10) se résout en trois étapes : prédiction, mise à jour et ré-échantillonnage comme le cas général d'un filtre à particules utilisé pour le suivi d'objet ou localisation de robots.

2.2 Principe de la méthode

Le schéma bloc de la spécialisation proposée à une itération donnée k est illustré dans FIGURE 27 et Algorithme 2 donne un résumé de son processus.

L'algorithme 2 commence par une étape de prédiction qui entraine un classifieur sur la base créée dans l'itération précédente et l'applique sur un ensemble d'images extraites de la vidéo de la scène cible pour une nouvelle base spécialisée. Ensuite, la pertinence de chaque échantillon de la base créée est déterminée au cours de l'étape de mise à jour en utilisant des stratégies d'observation qui assignent un poids à chaque échantillon. L'étape de ré-échantillonnage fait un tirage par importance pour sélectionner des échantillons cibles avec un poids élevé et pour sélectionner des échantillons source qui sont visuellement proches des cibles sélectionnés. Les échantillons sélectionnés des deux bases de données cible et source sont combinés pour créer un nouvel ensemble de données spécialisé pour la prochaine itération. L'algorithme se termine une fois le critère d'arrêt atteint. Les entrées de notre algorithme sont une base source, un classifieur générique qui peut être entrainé sur cette base source et une séquence vidéo d'une scène cible et les sorties à fournir sont une base de donnée spécialisée à la scène cible et un classifieur associé.

Dans la suite, nous donnons une description détaillée de chaque étape du filtre proposé.

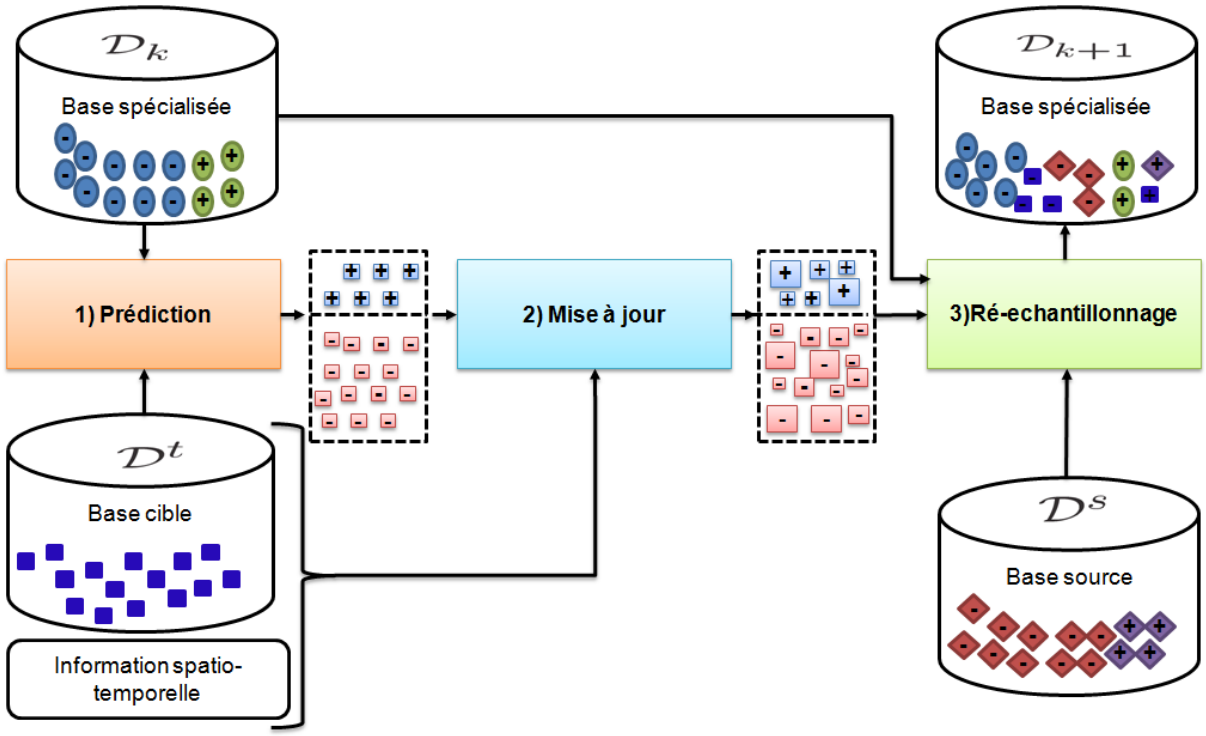


FIGURE 27 – Un schéma bloc de synthèse de la spécialisation séquentielle à base de filtre Monte-Carlo à une itération donnée k. 1) Étape de prédiction pour proposer des échantillons. 2) Étape de mise à jour pour pondérer les échantillons prédits. 3) Étape de ré-échantillonnage pour créer la base spécialisée

2.3 Étape de prédiction

L'objectif de cette étape est de proposer un ensemble d'échantillons à inclure dans la base spécialisée. Cette étape consiste donc à appliquer l'équation de Chapman-Kolmogorov (3.13) :

$$p(\mathbf{X}_{k+1}|\mathbf{Z}_{0:k}) = \int_{\mathbf{X}_k} p(\mathbf{X}_{k+1}|\mathbf{X}_k)p(\mathbf{X}_k|\mathbf{Z}_{0:k})d\mathbf{X}_k \tag{3.13}$$

L'équation de Chapman-Kolmogorov se base sur la dynamique du système $p(\mathbf{X}_{k+1}|\mathbf{X}_k)$ entre deux itérations successives pour proposer la base spécialisée $\mathcal{D}_{k+1} \doteq \{\mathbf{X}_k^{(n)}\}_{n=1,..,N^s}$ qui donne l'approximation (3.14) :

$$p(\mathbf{X}_{k+1}|\mathbf{Z}_{0:k}) = \sum_{n=1}^{N} \pi_k^n \ \delta_{\mathbf{X}_{k+1}^{(n)}}(\mathbf{X}_k) \approx \{\tilde{\mathbf{X}}_{k+1}^{(n)}\}_{n=1,..,\tilde{N}_{k+1}} \tag{3.14}$$

Nous notons par $\tilde{\mathcal{D}}_{k+1}$ la base spécialisée prédite à l'itération $(k+1)$ où $\tilde{\mathbf{X}}_{k+1}^{(n)}$ est un échantillon n et $\tilde{N}_{k+1}$ est le nombre des échantillons proposés par l'étape de prédiction. La FIGURE 28 résume le déroulement de l'étape de prédiction à une itération donnée $(k+1)$.

Algorithme 2 TTL-SMC pour la spécialisation d'un classifieur

Entrée: Base source $\mathcal{D}^s$
Classifieur générique Θ_g
Vidéo de la scène cible et la base associée $\mathcal{D}^t$
Nombre d'échantillons source N^s.
Paramètre α_s.
Sortie: Base spécialisée $\mathcal{D}$
Classifieur spécialisé $\Theta_{\mathcal{D}}$

$k \leftarrow 0$
$stop \leftarrow faux$
Tant que $stop \neq vrai$ **faire**

`/* Étape de prédiction */`
Si $(\mathcal{D}_k = \emptyset)$ **alors**
$Learn(\Theta_g, \mathcal{D}^s)^+$
Sinon
$Learn(\Theta_{\mathcal{D}_k}, \mathcal{D}_k)^+$
FinSi
$\tilde{\mathcal{D}}_{k+1} \leftarrow \{(\tilde{\mathbf{X}}_{k+1}^{(n)})\}_{n=1,..,\tilde{N}_{k+1}}$

Si $(|\tilde{\mathcal{D}}_{k+1}|/|\tilde{\mathcal{D}}_k| >= \alpha_s)$ **alors**
$stop \leftarrow vrai$
$Break$
FinSi

`/* Étape de mise à jour */`
$\breve{\mathcal{D}}_{k+1} \leftarrow \{(\breve{\mathbf{X}}_{k+1}^{(n)}, \breve{\pi}_{k+1}^{(n)})\}_{n=1,..,\breve{N}_{k+1}}$

`/* Étape de ré-échantillonnage */`
$\mathcal{D}_{k+1} \leftarrow \{(\mathbf{X}_{k+1}^{*(n)})\}_{n=1,..,N^s}$
$k \leftarrow k + 1$
FinTQ

$^+Learn(\Theta, \mathcal{D})$ une fonction qui entraine un classifieur Θ sur la base $\mathcal{D}$.

Dans notre cas comme il est visualisé dans la FIGURE 28, la base estimée est composée de trois sous-ensembles :

Un sous-ensemble 1 :

C'est un sous-échantillonnage de la base spécialisée précédente, utilisé pour propager la distribution (il s'agit d'un simple tirage aléatoire). Nous respectons le ratio entre les échantillons positifs et négatifs (généralement le même que la base source). Ce sous-ensemble approxime le terme $p(\mathbf{X}_k|\mathbf{Z}_{0:k})$ de l'équation (3.10) à l'itération $(k-1)$ conformément à l'équation (3.15) :

$$p(\mathbf{X}_k|\mathbf{Z}_{0:k}) \approx \{\mathbf{X}_{k+1}^{*(n)}\}_{n=1,..,N^*} \tag{3.15}$$

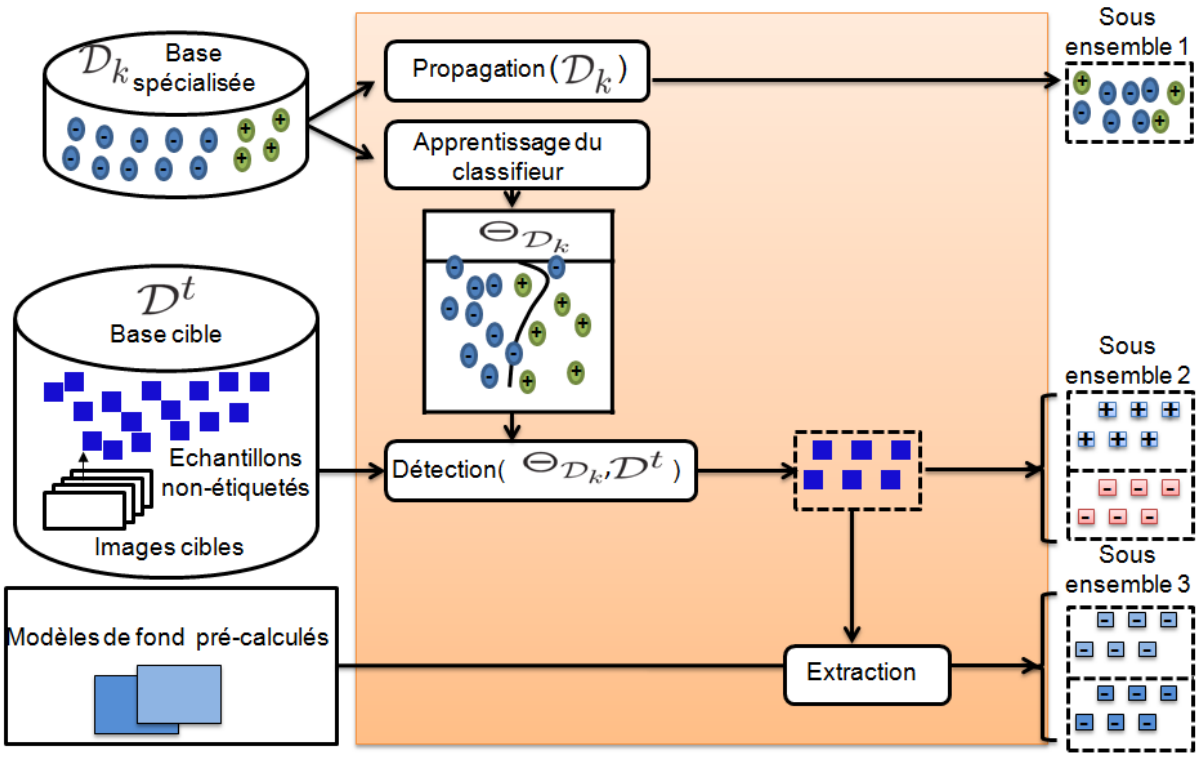


FIGURE 28 – Le processus de proposition d'échantillons lors de l'étape de prédiction. Proposition de trois sous-ensemble 1, 2, et 3 formant la base estimée par l'étape de prédiction du filtre

où $\mathbf{X}_{k+1}^{*(n)}$ est l'échantillon n sélectionné de la base $\mathcal{D}_k$ à l'itération $(k+1)$, N^* représente le nombre d'échantillons de ce sous-ensemble avec $N^* = \alpha_t N^s$, $(\alpha_t \in [0,1])$. Le paramètre α_t détermine le nombre d'échantillons à propager à partir de la base précédente.

Un sous-ensemble 2 :

Pour obtenir ce sous-ensemble, nous entrainons un classifieur spécialisé $\Theta_{\mathcal{D}_k}$ sur la base $\mathcal{D}_k$ et nous l'utilisons pour détecter les objets présents dans un ensemble d'images extraites uniformément de la séquence vidéo cible. Pour ce faire, nous avons recours à la technique de balayage multi-échelles par fenêtre fixe. Cette technique couvre une instance de l'objet d'intérêt par plusieurs boites englobantes, ainsi nous utilisons une fonction de regroupement spatial "mean-shift" pour fusionner les boîtes englobantes les plus proches. Par ailleurs, ce sous-ensemble contient certains échantillons présentant bien l'objet d'intérêt et d'autres échantillons qui sont des fausses détections. C'est pourquoi, nous supposons que chaque détection peut être un échantillon positif ou négatif. Ainsi, pour chaque détection, deux propositions sont faites : une positive et une négative. Ce sous-ensemble est renvoyé par l'équation (3.16).

$$\{\breve{\mathbf{X}}_{k+1}^{(n)}\}_{n=1,..,\breve{N}_k} \doteq \{(\mathbf{x}^{(n)}, y)\}_{y\in\mathcal{Y}\ ;\mathbf{x}^{(n)}\in\mathcal{D}^t/\Theta_{\mathcal{D}_k}(\mathbf{x}^{(n)})>0} \tag{3.16}$$

$\breve{\mathbf{X}}_{k+1}^{(n)}$ représente l'échantillon cible n proposé pour la base spécialisée de l'itération $(k+1)$ sachant qu'il est classé positif à l'itération k.

Ensuite, la décision entre les deux étiquettes est assurée dans l'étape de mise à jour par une stratégie d'observation qui attribuera un poids à chaque échantillon.

Il est à rappeler que la stratégie de balayage par fenêtre fixe permet d'extraire plein de propositions dont plusieurs sont redondants étant donnée que nous utilisons des images issues de la même scène.

Afin de réduire ce nombre, nous avons fait le choix de prendre uniquement les exemples classés comme positifs (c'est à dire les exemples qui ont un score de classification supérieur à zéro) par le classifieur utilisé. Ce choix est fait en se basant sur l'idée de l'étape "bootstrap" utilisée dans la littérature de création de détecteur générique. Notre choix donne la main à sélectionner les faux positifs connus par "hard-exemples" qui permettent l'amélioration de la performance du classifieur une fois qu'ils sont pris de nouveau dans l'apprentissage.

Un sous-ensemble 3 :

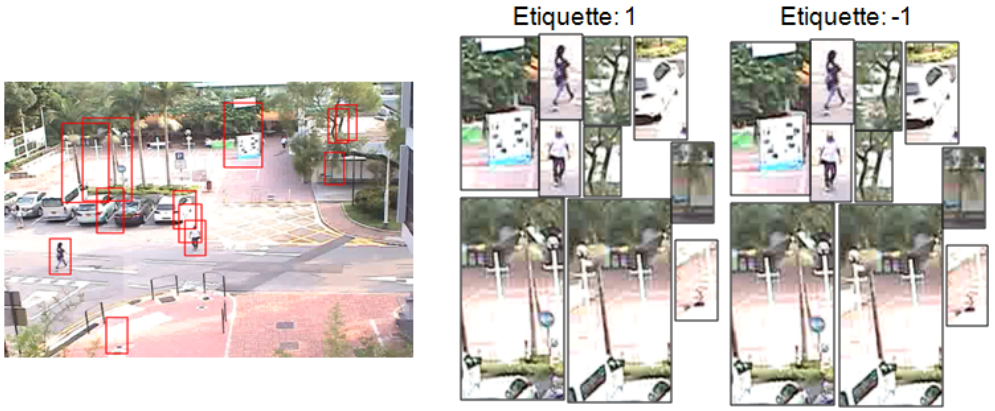
Dans certains cas, le classifieur spécialisé précédent n'arrive pas à détecter la plupart des instances de l'objet d'intérêt. C'est-à-dire les détections ratées sont nombreuses et les faux positifs peuvent être en nombre réduit. Dans d'autres cas, il est difficile de favoriser une étiquette pour plusieurs échantillons de sous-ensemble 2. Dans ces conditions, nous ne pouvons pas sélectionner suffisamment d'échantillons cibles négatifs pour spécialiser le classifieur cible. Pour éviter ces situations, nous utilisons des modèles de fond pré-calculés (dans notre cas, un fond médian et un fond moyen) pour fournir des échantillons cibles négatifs et produire le sous-ensemble 3 selon l'équation (3.17) :

$$\{\breve{\mathbf{X}}_{k+1}^{'(n)}\}_{n=1,..,\breve{M}_k} \doteq \cup \sum_{b_j in\{b1,...,bm\}} \{(\mathbf{x}^{'(n)}, -1)\}_{\mathbf{x}^{'(n)} \in b_j} \tag{3.17}$$

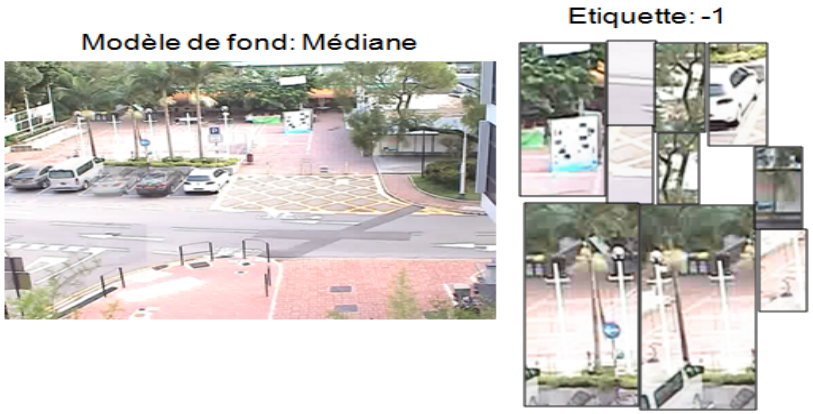
Où $(\mathbf{x}^{'(n)}, -1)$ est un échantillon prélevé à partir d'un modèle de fond pré-calculé et étiqueté négativement. $\breve{M}_k = m * \breve{N}_k$ est le nombre total des échantillons de fond.

La FIGURE 29 montre une illustration des stratégies de proposition des échantillons cibles qui forment le sous-ensemble 2 et le sous-ensemble 3 à partir d'une image de la scène cible. A la première itération, le sous-ensemble 1 est vide et les propositions qui composent le sous-ensemble 2 et le sous-ensemble 3 sont données à l'aide d'un détecteur générique formé sur la base INRIA.

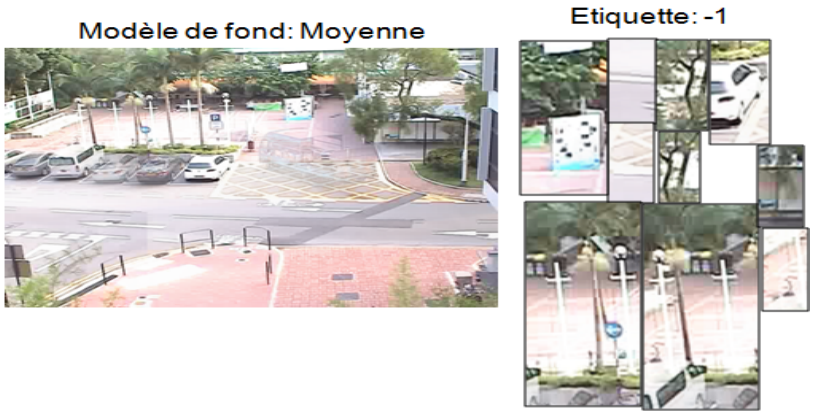
(a) Balayage multi-échelles pour la détection des piétons



(b) Regroupement spatial avec mean-shift et sélection des échantillons cibles



(c) Modèle de fond médiane : extraction des échantillons



(d) Modèle de fond moyenne : extraction des échantillons

FIGURE 29 – Illustration de la proposition des échantillons cibles lors de l'étape prédiction.

2.4 Étape de mise à jour

Cette étape détermine le terme de vraisemblance (3.18) en utilisant un ensemble de stratégies d'observation. Ces dernières permettent d'affecter un poids $\breve{\pi}_{k+1}^{(n)}$ à chaque échantillon $\breve{\mathbf{X}}_{k+1}^{(n)}$ retourné par le classifieur à l'étape de prédiction.

$$p\left(\mathbf{Z}_{k+1}|\mathbf{X}_{k+1}=\breve{\mathbf{X}}_{k+1}^{(n)}\right) \propto \breve{\pi}_{k+1}^{(n)} \tag{3.18}$$

Les stratégies d'observation utilisent des indices contextuels et des *a priori* extraits à partir de la séquence vidéo de la scène cible. Parmi ces informations, nous citons le mouvement d'objets, le suivi des points KLT, la soustraction fond-forme et/ou le chemin modèle de chaque objet. Ces stratégies seront détaillés dans le chapitre 4 page 69. La sortie de cette étape est un ensemble d'échantillons cibles pondérés que nous appelons par la suite base cible pondérée (3.19) :

$$\{(\breve{\mathbf{X}}_{k+1}^{(n)}, \breve{\pi}_{k+1}^{(n)})\}_{n=1,..,\breve{N}_{k+1}} \tag{3.19}$$

avec $(\breve{\mathbf{X}}_{k+1}^{(n)}, \breve{\pi}_{k+1}^{(n)})$ le couple formé d'un échantillon cible et de son poids associé et $\breve{N}_{k+1}$ représente le nombre total des échantillons de cette base.

2.5 Étape de re-échantillonnage

L'objectif de cette étape est de construire une nouvelle base de données spécialisée tout en décidant selon un tirage par importance quels sont les échantillons qui seront inclus dans la base spécialisée. Cette dernière approxime la distribution *a posteriori* $p(\mathbf{X}_{k+1}|\mathbf{Z}_{0:k+1})$ selon l'équation (3.20) :

$$p(\mathbf{X}_{k+1}|\mathbf{Z}_{0:k+1}) \approx \{\mathbf{X}_{k+1}^{*(n)}\}_{n=1,..,N^s} \tag{3.20}$$

où $\mathbf{X}_{k+1}^{*(n)}$ est l'échantillon n sélectionné pour être dans la base spécialisée $\mathcal{D}_{k+1}$; un échantillon peut être sélectionné à partir de la base source ou de la base cible.

Dans cette étape, nous faisons un tirage par importance pour approximer la distribution conditionnelle $p(\breve{\mathbf{X}}_{k+1}|\mathbf{Z}_{k+1})$ des échantillons cibles pondérés à l'aide des observations. Le tirage par importance donne un ensemble d'échantillons non-pondérés reflétant l'ensemble pondéré d'entrée et il nous permet de traiter les échantillons d'apprentissage en fonction de leur poids d'importance et ceci sans changer la fonction de décision (connue aussi par fonction objective d'apprentissage) pour intégrer les poids d'échantillons au cours de la phase d'apprentissage.

L'idée principale du tirage par importance consiste à remplacer les échantillons de poids fort par nombreux échantillons et à remplacer les échantillons associés à un poids faible par peu d'échantillons, à la fin tous les échantillons rendus ont un poids identique. En effet, ce tirage transforme le poids d'un échantillon en un nombre de répétitions.

La distribution conditionnelle $p(\breve{\mathbf{X}}_{k+1}|\mathbf{Z}_{k+1})$ est approchée, selon l'équation (3.21), en fusionnant la base cible non-pondérée issue du sous-ensemble 2 et une sélection aléatoire du sous-ensemble 3. La base cible non-pondérée est générée en appliquant un tirage par importance sur la base pondérée fournie par l'étape de mise à jour.

$$p(\breve{\mathbf{X}}_{k+1}|\mathbf{Z}_{k+1}) \approx \{\breve{\mathbf{X}}_{k+1}^{*(n)}\}_{n=1,..,\breve{N}_{k+1}^{*}} \cup \{\breve{\mathbf{X}}_{k+1}^{*'(n)}\}_{n=1,..,\breve{M}_{k+1}^{*}} \tag{3.21}$$

Où $\breve{\mathbf{X}}_{k+1}^{*(n)}$ et $\breve{\mathbf{X}}_{k+1}^{*'(n)}$ sont respectivement, les échantillons cibles sélectionnés pour l'itération $(k+1)$ à partir du sous-ensemble 2 et du sous-ensemble 3.

A ce niveau, la distribution $p(\mathbf{X}_{k+1}|\mathbf{Z}_{0:k+1})$ est approchée par les trois sous-ensembles 1, 2 et 3 selon l'équation (3.22) :

$$p(\mathbf{X}_{k+1}|\mathbf{Z}_{0:k+1}) \approx \{\mathbf{X}_{k+1}^{*(n)}\}_{n=1,..,N^*} \cup \{\breve{\mathbf{X}}_{k+1}^{*(n)}\}_{n=1,..,\breve{N}_{k+1}^*} \cup \{\breve{\mathbf{X}}_{k+1}^{*'(n)}\}_{n=1,..,\breve{M}_{k+1}^*} \tag{3.22}$$

Certains échantillons cibles sélectionnés peuvent être associés à de fausses étiquettes puisqu'ils sont associés de manière automatique. De plus, ils peuvent être insuffisants en nombre pour générer un classifieur robuste à la scène cible. Cependant, la base de données source contient uniquement des échantillons étiquetés et qui peuvent être bénéfiques à la spécialisation du classifieur. Ainsi, nous proposons d'utiliser la distribution source pour améliorer l'estimation de la cible en sélectionnant uniquement les échantillons sources qui dérivent de la même distribution cible (3.22). C'est-à-dire, nous cherchons à prendre les échantillons sources qui ressemblent visuellement le plus à ceux sélectionnés de la scène cible. La probabilité $\breve{\pi}_{k+1}^{s(n)}$ (poids) qui évalue l'appartenance de chaque échantillon source à la distribution cible $p(\mathbf{X}_{k+1}|\mathbf{Z}_{0:k+1})$ est calculée en utilisant une méthode non-paramétrique basée sur l'algorithme KNN [1]. Le processus de calcul de cette probabilité se déroule en deux phases :

1. Sélection des k plus proches voisins cibles à chaque échantillon de la base source. La distance $L2$ entre les vecteurs caractéristiques de deux échantillons est utilisée pour le calcul de "matrice source-cible".
2. Calcul de la probabilité d'appartenance à la distribution cible de chaque échantillon source. Elle est calculée en fonction des distances de k échantillons cibles sélectionnés.

Nous donnons dans Tableau 4 les instructions du pondération de la base source.

Tableau 4 – Pondération des échantillons de la base source

1- Remplir la "matrice source-cible" par les distances deux à deux.

Source \ Cible	$\mathbf{X}_t^0$	$\cdots$	$\mathbf{X}_t^{m1}$	$\cdots$	$\mathbf{X}_t^{N^t}$
$\mathbf{X}_s^0$	$d(\mathbf{X}_s^0, \mathbf{X}_t^0)$	$\cdots$	$d(\mathbf{X}_s^0, \mathbf{X}_t^{m1})$	$\cdots$	$d(\mathbf{X}_s^0, \mathbf{X}_t^{N^t})$
$\vdots$					
$\mathbf{X}_s^{m2}$	$d(\mathbf{X}_s^{m2}, \mathbf{X}_t^{m2})$	$\cdots$	$d(\mathbf{X}_s^{m2}, \mathbf{X}_t^{m1})$	$\cdots$	$d(\mathbf{X}_s^{m2}, \mathbf{X}_t^{N^t})$
$\vdots$					
$\mathbf{X}_s^{N^s}$	$d(\mathbf{X}_s^{N^s}, \mathbf{X}_t^0)$	$\cdots$	$d(\mathbf{X}_s^{N^s}, \mathbf{X}_t^{m1})$	$\cdots$	$d(\mathbf{X}_s^{N^s}, \mathbf{X}_t^{N^t})$

2- Ordonner en ordre croissant les échantillons cibles en fonction de leurs distances vis-à-vis de chaque échantillon source
3-Calculer les probabilités d'appartenance des échantillons sources à la distribution cible selon l'équation (3.23) :

Pour $n = 1$ **à** N^s **faire**

$$\pi_s^n = \sum_{j=0}^{k} \exp(-d(\mathbf{X}_s^n, \mathbf{X}_t^j)) \tag{3.23}$$

1. La bibliothèque FLANN (http ://www.cs.ubc.ca/research/ ann/) et une distance L2 sur les vecteurs caractéristiques sont utilisées

En se basant sur ces probabilités, nous faisons également un tirage par importance pour sélectionner les échantillons sources qui sont le plus proche de ceux sélectionnés de la scène cible et qui appartiennent à $p(\mathbf{X}_{k+1}|\mathbf{Z}_{0:k+1})$ selon l'équation (3.24) :

$$p(\mathbf{X}_{k+1}|\mathbf{Z}_{0:k+1}) \approx \{\mathbf{X}_{k+1}^{s*(n)}\}_{n=1,..,\breve{N}_{k+1}^{s*}} \tag{3.24}$$

Où $\mathbf{X}_{k+1}^{s*(n)}$ est un échantillon source n sélectionné pour être inclus, dans la base spécialisée de l'itération $(k+1)$ et $\breve{N}_{k+1}^{s*}$ est le nombre total des échantillons sources sélectionnés.

Ce nombre est déterminé à l'aide de l'équation (3.25) sachant que N^s, N^*, $\breve{N}_{k+1}^*$, $\breve{N}_{k+1}^*$ sont respectivement le nombre des échantillons de la base source, de sous-ensemble 1, de sous-ensemble 2 et de sous-ensemble 3 :

$$\breve{N}_{k+1}^{s*} = N^s - (N^* + \breve{N}_{k+1}^* + \breve{M}_{k+1}^*) \tag{3.25}$$

À la fin de cette étape, la nouvelle base spécialisée $\mathcal{D}_{k+1}$ est construite à partir des échantillons source et cible ((3.26), et elle est utilisée pour commencer l'itération suivante.

$$\mathcal{D}_{k+1} \doteq \{\mathbf{X}_{k+1}^{*(n)}\}_{n=1,..,N^*} \cup \{\breve{\mathbf{X}}_{k+1}^{*(n)}\}_{n=1,..,\breve{N}_{k+1}^*} \cup \{\breve{\mathbf{X}}_{k+1}^{*'(n)}\}_{n=1,..,\breve{M}_{k+1}^*} \cup \{\mathbf{X}_{k+1}^{s*(n)}\}_{n=1,..,\breve{N}_{k+1}^{s*}} \tag{3.26}$$

La FIGURE 30 résume le processus de l'étape de re-échantillonnage à une itération donnée.

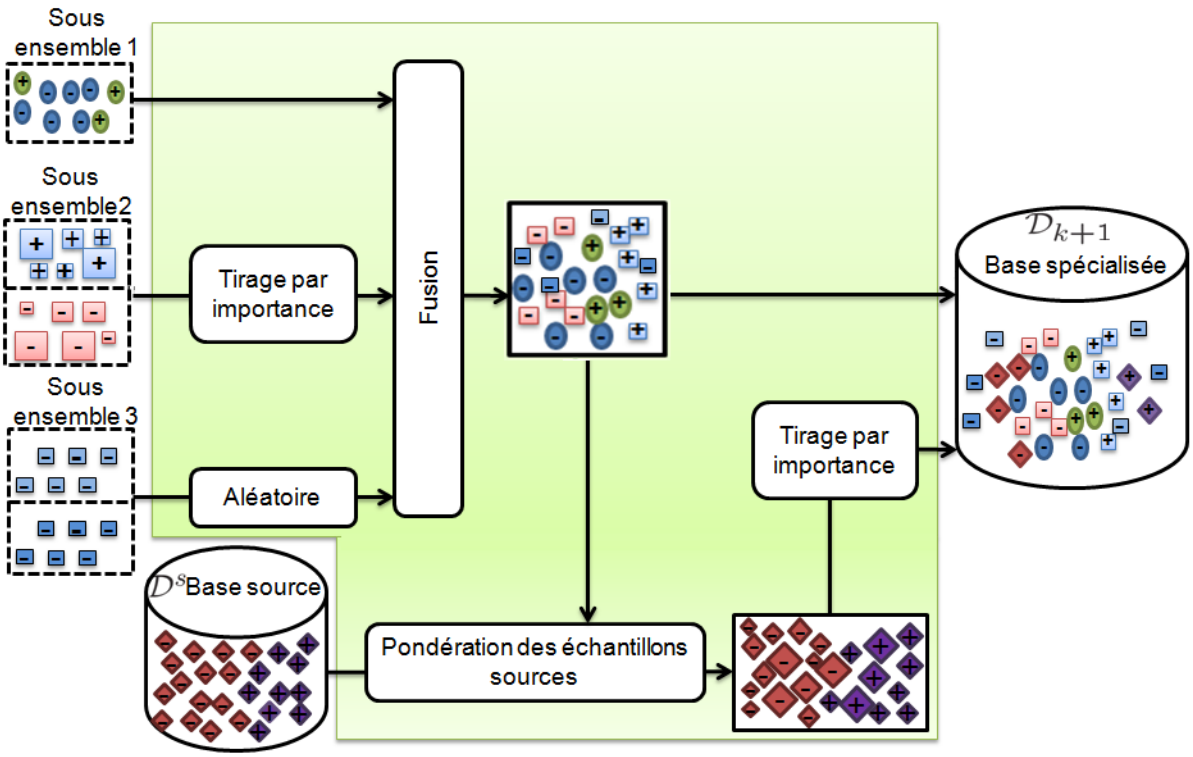


FIGURE 30 – Processus de l'étape de re-échantillonnage

Le processus de spécialisation s'arrête lorsque le rapport entre la cardinalité de deux bases prédites liés à deux itérations consécutives ($|\tilde{\mathcal{D}}_{k+1}|/|\tilde{\mathcal{D}}_k|$) dépasse un seuil α_s fixé précédemment (dans notre cas, nous utilisons $\alpha_s = 0.8$). $|.|$ représente la cardinalité d'une base de données. Le classifieur obtenu sera utilisé pour détecter l'objet d'intérêt, dans la scène cible, en se basant uniquement sur l'apparence de ce dernier.

Conclusion

Nous avons commencé ce chapitre par un rappel sur le principe de filtre Séquentiel de Monte Carlo. Ensuite, nous avons décrit l'approche proposée qui concerne l'utilisation de filtre SMC dans un contexte de transfert d'apprentissage transductif. Pour ce faire, nous avons donné au début de la deuxième section du chapitre, les notations et les définitions à utiliser par la suite dans le chapitre. Après, nous avons présenté le principe de la méthode à travers son schéma bloc à une itération donnée et à travers un algorithme récapitulatif. Puis, les différentes étapes du filtre sont détaillées.

Chapitre 4

Les stratégies d'observation

Introduction

Nous avons présenté dans le chapitre précédent, notre formalisation de transfert d'apprentissage à base du filtre SMC pour le spécialisation d'un classifieur (ou détecteur) vers une scène particulière. Un filtre SMC est composée de trois étape : prédiction, mise à jour et ré-échantillonnage. L'étape de prédiction sert à proposer une base spécialisée à l'itération suivante. L'étape de mise à jour cherche à pondérer les échantillons de cette base et évaluer leurs pertinences en utilisant une stratégie d'observation. L'étape de ré-échantillonnage génère une base spécialisée non-pondérée à partir de la sortie de l'étape mise à jour.

Au niveau de ce chapitre, nous détaillons deux stratégies d'observation utilisées dans l'étape de mise à jour du filtre pour pondérer les échantillons de la scène cible. La première stratégie qui calcule deux indices spatio-temporels pour attribuer un poids à un échantillon, est exposée dans la première section. Ensuite, le principe de la deuxième stratégie et son algorithme de pondération sont décrites dans la deuxième section. En particulier, ces stratégies cherchent à pondérer les échantillons du sous-ensemble 2 (proposé par l'étape de prédiction) afin de choisir la proposition correcte en utilisant des informations spatio-temporelles extraites de la scène cible.

Dans la suite de ce chapitre, nous raisonnons par rapport à un piéton comme objet d'intérêt, mais les stratégies peuvent être appliquées à d'autres types d'objets tels que voiture, moto, vélo, ...

1 Stratégie d'indices spatio-temporels OAS

La première stratégie de pondération est intitulée Overlap-Accumulation Scores (OAS). Elle est basée sur deux indices spatio-temporels simples : un overlap_score et un accumulation_score.

Pour calculer ces indices nous nous basons sur l'idée que dans une scène de circulation routière, il est rare que les piétons restent stables pendant une longue période de temps. Ainsi, une bonne détection se produit dans un blob de forme, alors que les fausses détections apparaissent comme un ensemble de régions d'intérêts (ROIs) qui se répètent au fil du temps au même endroit et avec une taille quasi-stable.

1.1 Indice 1 : overlap_score

Nous calculons pour chaque proposition un overlap_score qui compare le recouvrement du ROI associée à la proposition avec les blobs rendus d'un algorithme d'extraction fond-forme. Le overlap_score mesure le rapport entre la surface d'intersection et la somme des surfaces de deux rectangles. La Figure 31 illustre un exemple d'un overlap_score.

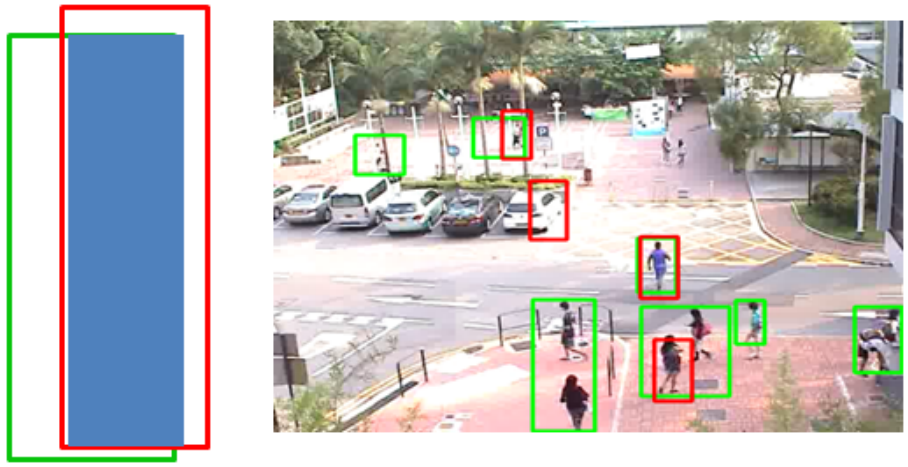


FIGURE 31 – Illustration d'un overlap_score. Un rectangle rouge représente une proposition rendue par le détecteur. Un rectangle vert est un blob retourné par un algorithme de soustraction fond-forme. Le rectangle rempli en bleu est l'intersection entre les deux que nous cherchons à mesurer le taux par rapport à la somme des deux surfaces

Nous notons par :

— ROI : Le premier rectangle qui présente une proposition du sous-ensemble 2.

— FG : Le deuxième rectangle qui décrit le blob fourni par la méthode d'extraction fond/forme.

Ainsi, Nous notons cet overlap_score par λ_o où $\lambda_o \in [0, 1]$ et nous faisons le calcul conformément à l'équation (4.1) :

$$\lambda_o(ROI) \doteq \frac{2(ROI \cap FG)\_AREA}{ROI\_AREA + FG\_AREA} \tag{4.1}$$

Avec $\lambda_o(ROI)$ est le taux de recouvrement relative à la (ROI) associée à un échantillon. ROI_AREA est la surface de la ROI, FG_AREA est la surface d'un FG présentant un blob de fond-forme et $(ROI \cap FG)\_AREA$ est la surface de l'intersection entre la ROI de l'échantillon et le blob FG

1.2 Indice 2 : accumulation_score

Nous calculons également pour chaque proposition un accumulation_score qui mesure le taux de détections trouvées au même emplacement à travers les images de spécialisation. Cet accumulation_score est noté λ_a (où $\lambda_a \in [0, 1]$) et il met en évidence les zones du fond de l'image pour lesquelles le classifieur répond positivement. Nous appliquons l'équation (4.2) pour calculer λ_a :

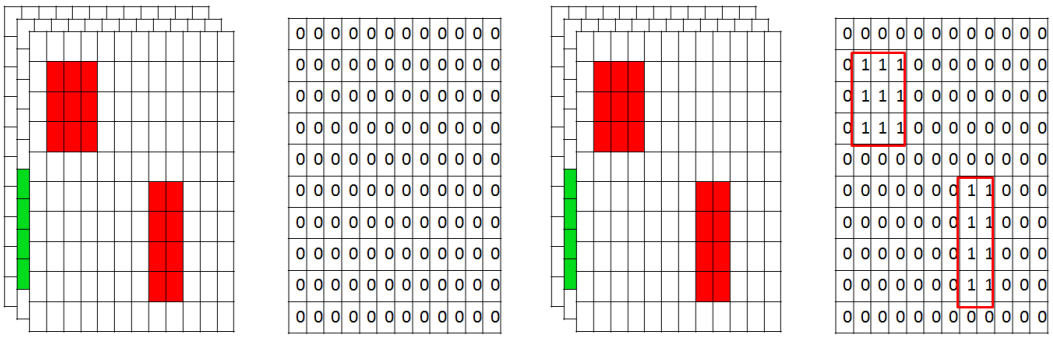
$$\lambda_a(ROI^1) \doteq \frac{min(NB\_FRAMES, \sum ROI\_SIZE\&LOC)}{NB\_FRAMES} \tag{4.2}$$

Avec $\lambda_o(ROI)$ est l'accumulation_score relative à la (ROI^1) associée à un échantillon 1, NB_FRAMES est le nombre des images de spécialisation. $\sum ROI\_SIZE\&LOC$ est la mesure qui prend en considération dans la somme de la taille de la ROI^1 toute les ROIs qui sont pratiquement dans la même position. Cette mesure est calculée selon (4.3) :

$$\sum ROI\_SIZE\&LOC = \frac{\sum ROI\_SIZE}{ROI\_AREA} \tag{4.3}$$

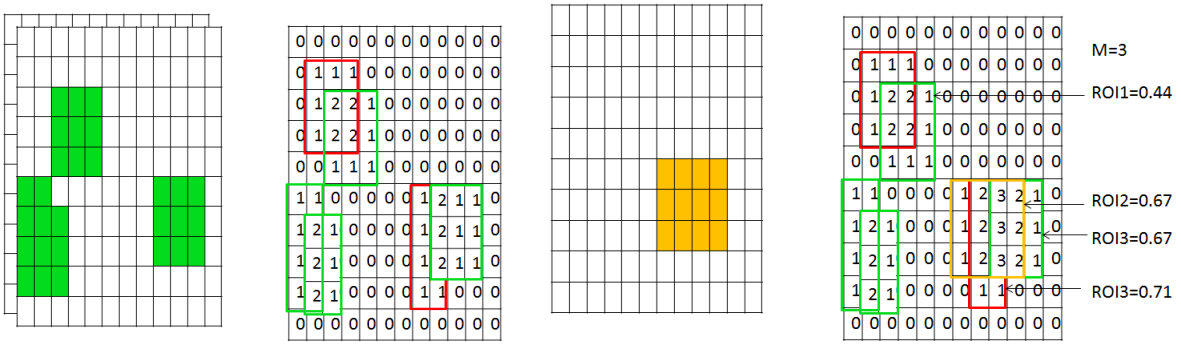
Où $\sum ROI\_SIZE$ c'est la somme de tout les pixels de la taille de la ROI après l'accumulation de toutes les ROIs détectées dans toutes les images de spécialisation.

(a) Illustration d'un accumulation_score.



(b) Initialisation à zéro d'une matrice de même taille qu'une image de la scène

(c) Traitement des ROIs de l'image 1



(d) Traitement des ROIs de l'image 2

(e) Traitement des ROIs de l'image 3 et Calcul d'un accumulation_score pour chaque ROI

FIGURE 32 – Illustration des étapes de calcul d'un accumulation_score

La FIGURE 32a donne une illustration sur un accumulation_score et donne un exemple des étapes de calcul pour M images de spécialisation. M représente le NB_FRAMES définit précédemment.

1.3 Algorithme de pondération selon la stratégie d'observation "OAS"

En se basant sur overlap_score et accumulation_score, attribuer un poids à un échantillon de sous-ensemble 2 et favoriser automatiquement la proposition associée à l'étiquette correcte devient plus facile et peut se faire en implémentant Algorithme 3. Le Tableau 5 présente quelques notations utilisées dans Algorithme 3.

Tableau 5 – Notations et fonctions utilisées dans Algorithme 3

Notation : Définition
- $\mathbf{p}$: C'est la position d'une ROI associée à un échantillon dans la séquence vidéo cible ($\mathcal{D}^t$). - *compute_overlap*$(\mathbf{p}, \mathcal{D}^t)$: Fonction qui calcule un overlap_score d'une ROI représentée par sa position $\mathbf{p}$. - *compute_accumulation*$(\mathbf{p}, \mathcal{D}^t)$: Fonction qui calcule un accumulation_score d'une ROI représentée par sa position $\mathbf{p}$.

Algorithme 3 Stratégie d'observation 1 : OAS

Entrée: Sous-ensemble 2 $\{\breve{\mathbf{X}}_{k+1}^{(n)}\}_{n=1,..,\breve{N}}$ avec les positions ROIs et les tailles associées dans la séquence vidéo cible$\{\mathbf{p}_i\}_{i=1,..,\breve{N}}$
La séquence vidéo cible et la base associée $\mathcal{D}^t$
α_p : Seuil de recouvrement
Sortie: L'ensemble $\{\pi_i\}_{i=1..,\breve{N}}$ des poids attribués aux échantillons

Pour $i = 1$ à $\breve{N}$ **faire**
  $\pi_i \leftarrow 0$
  `/* Calcul des indices spatio-temporels */`
  $\lambda_o \leftarrow$ *compute_overlap*$(\mathbf{p}_i, \mathcal{D}^t)$
  $\lambda_a \leftarrow$ *compute_accumulation*$(\mathbf{p}_i, \mathcal{D}^t)$
  `/* Attribution des poids */`
  Si $(\breve{y}_i = piéton)$ **alors**

    Si $(\lambda_o >= \alpha_p)$ **alors**
      $\pi_i \leftarrow \lambda_o$
    FinSi
  Sinon
    Si $((\lambda_o = 0.0)\&(\lambda_a > 0.0))$ **alors**
      $\pi_i \leftarrow \lambda_a$
    FinSi
  FinSi
FinPour

Un échantillon positif sera associé à un poids égal à son overlap_score si λ_o dépasse ou égale un seuil α_p [1] déterminé empiriquement. Sinon, il sera associé à zéro. Un raisonnement similaire est fait dans le cas d'un échantillon négatif ; Il aura son accumulation_score comme un poids si son λ_o est nul et son λ_a est supérieur à zéro. Sinon, il aura un poids nul. Tout échantillon associé à un poids nul sera rejeté lors du ré-échantillonnage.

1. Dans nos expérimentations, nous avons fixé ce seuil à la valeur qui donne le moins de fausses détections et qui garantit un nombre acceptable d'échantillons positifs sur 9 valeurs (0.5, 0.55, 0.60, 0.65, 0.70, 0.75, 0.80, 0.85, 0.90) testées pour chaque base

2 Stratégie de suivi KLT

Nous proposons une deuxième stratégie pour la pondération des échantillons du sous-ensemble 2, qui utilise le suivi des points d'intérêts KLT (dits aussi des points caractéristiques et connu en anglais par *KLT feature tracker*) [Tomasi et Kanade, 1991, Shi et Tomasi, 1994].

2.1 Description du principe de la stratégie

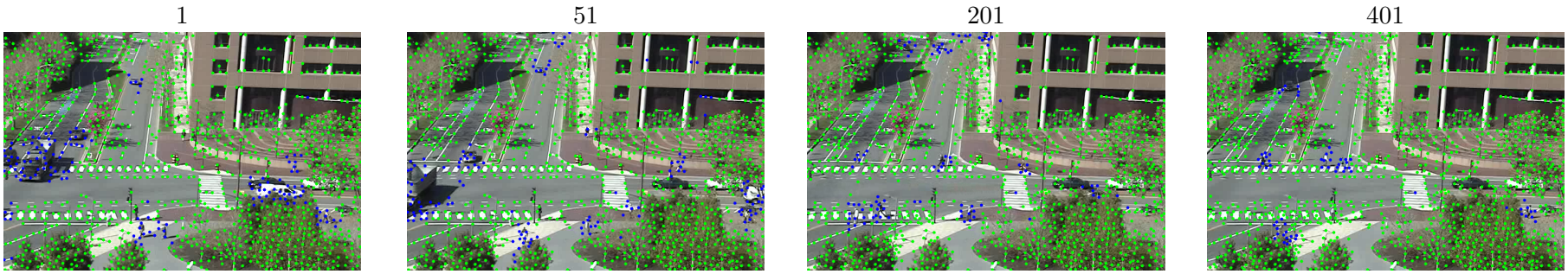
La technique de suivi KLT vise à trouver pour chaque point d'intérêt, détecté sur l'image (i) de la vidéo, un point d'intérêt correspondant, détecté sur l'image $(i+1)$.

Dans un premier temps, nous utilisons les informations de correspondance entre des images consécutives pour : (1) Attribuer un identificateur (ID) pour chaque point d'intérêt détecté et suivi jusqu'à l'image courante (i) de la vidéo et (2) Mettre à jour et sauvegarder les trois paramètres $Life$, $AmpX$ et $AmpY$ associés à chaque point. Le paramètre $Life$ indique le nombre des images contenant ce point d'intérêt jusqu'à atteindre l'image courante de la vidéo. Ce paramètre présente la durée de vie d'un point. $AmpX$ représente l'amplitude du déplacement latérale (suivant l'axe x). Et $AmpY$ est l'amplitude du déplacement verticale (suivant l'axe y). Ensuite et une fois que toute la vidéo est traitée, nous re-propageons pour chaque point, les valeurs de ses paramètres de la dernière image vers la première. La FIGURE 33 visualise l'idée principale de cette stratégie et l'algorithme 4 donne les instructions de calcul de ces paramètres pour chaque point.

Algorithme 4 Calcul des paramètres relatives aux points d'intérêts

Entrée: La séquence vidéo cible et la base associée $\mathcal{D}^t$
Nombre des images de spécialisation : NB_Frames
Pas entre les images de spécialisation : pas
Numéro de première image de spécialisation : $init$
Sortie: Liste des points d'intérêts $\{FPts_j\}$ de chaque image j, $\{FPts_j\}_{j=1..,L}$

$L \leftarrow init + (NB\_Frames \times pas)$
$\{FPts_j\}_{j=1..,L} \leftarrow \emptyset$
Pour $i = 1$ à L **faire**
  `/* Sens 1 */`
  $T[i] \leftarrow$ Détecter les points d'intérêts $\{FPts_i\}$ de i
  $Status[i-1] \leftarrow$ Calculer le flux optique entre $T[i-1]$ et $T[i]$
  Pour $j = 1$ à $Size(Status[i-1])$ **faire**
    Si $Status[i-1][j] == 1$ **alors**
      Chercher ID de $T[i-1][j]$
      Calculer $Life$, $AmpX$, $AmpY$ de $T[i][j]$
      Ajouter le point $\{FPts_i\}$
    FinSi
  FinPour
FinPour
`/* Sens 2 */`
Pour $i = L-1$ à 1 **faire**
  Si ID de $T[i]$ dans $T[i+1]$ **alors**
    Mettre à jour $Life$, $AmpX$, $AmpY$ de $T[i][j]$ par les les valeurs dans $T[i+1]$
  FinSi
  $\{FPts_j\}_{j=1..,L} \leftarrow \{FPts_j\}_{j=1..,L} + \{FPts_i\}$
FinPour



ID	$Life$	$AmpX$	$AmpY$
0	2	-0.0001	-0.0002
500	2	-0.0005	-7.e-005
801	2	-0.236	0.329
1000	2	0.175	0.0192
1274	2	0	6.1-005

ID	$Life$	$AmpX$	$AmpY$
0	52	-11.9	-11.6
500	52	-0.0914	0.104
801	52	-0.7716	0.75
1000	52	0.662	0.0652
1275	22	0.0401	0.0618

ID	$Life$	$AmpX$	$AmpY$
0	202	-33.3	-33.3
500	202	-0.0901	-0.0917
801	202	-0.0115	0.0065
1000	202	0.63	0.642
1276	172	67.9	67.8

ID	$Life$	$AmpX$	$AmpY$
0	402	-33.2	-33.2
500	402	0.19	0.18
801	402	-0.745	-0.734
1000	402	11.9	11.3
1278	372	-3.89	-3.89

ID	$Life$	$AmpX$	$AmpY$
0	7539	60.2	60.2
500	7539	-12.4	-12.4
801	2352	-37.8	-37.7
1000	5434	34.2	33.9
1274	36	-0.424	-0.465

ID	$Life$	$AmpX$	$AmpY$
0	7539	60.2	60.2
500	7539	-12.4	-12.4
801	2352	-37.8	-37.7
1000	5434	34.2	33.9
1275	72	-0.745	-0.649

ID	$Life$	$AmpX$	$AmpY$
0	7539	60.2	60.2
500	7539	-12.4	-12.4
801	2352	-37.8	-37.7
1000	5434	34.2	33.9
1276	189	72.6	71.6

ID	$Life$	$AmpX$	$AmpY$
0	7539	60.2	60.2
500	7539	-12.4	-12.4
801	2352	-37.8	-37.7
1000	5434	34.2	33.9
1278	7509	222	222

FIGURE 33 – Stratégie de suivi KLT des points d'intérêts. Un point vert présente un faible mouvement entre deux images successives et un point bleu présente un déplacement plus important

Dans un deuxième temps, nous exploitons les trois paramètres sauvegardés de chaque point d'intérêt pour le classer en tant qu'un point caractéristique mobile (point de forme) ou en tant qu'un point statique (point de fond). Pour ce faire, nous nous basons sur deux hypothèses :

— Classification en fonction de paramètre $Life$: Si un point d'intérêt pt est suivi sur un nombre très petit d'images alors il est probable que son mouvement est du à un bruit (c'est du bruit). Sinon, si pt est présent sur un nombre très grand d'images alors ce point est un point de fond. Un point d'intérêt sera considéré comme un point mobile si son paramètre $Life$ a une valeur dans l'intervalle $[minlife, maxlife]$.

— Classification en fonction de l'amplitude de déplacement $AmpX$ et $AmpY$: Si un point d'intérêt pt a fait un déplacement très faible alors il est probablement qu'il est un point statique (un point de fond). Par contre, si son déplacement est assez grand il est supposé être un point de bruit. Un point d'intérêt sera considéré comme un point mobile si ses paramètres $AmpX$ ou $AmpY$ ont des valeurs dans $[minamp, maxamp]$.

FIGURE 34 présente les deux hypothèses de classification des points d'intérêts. Dans nos expérimentations, les valeurs de $minlife$, $maxlife$, $minamp$, $maxamp$ sont déterminés empiriquement [2] pour chaque séquence vidéo et elles sont données comme entrées lors de la classification des points d'intérêts.

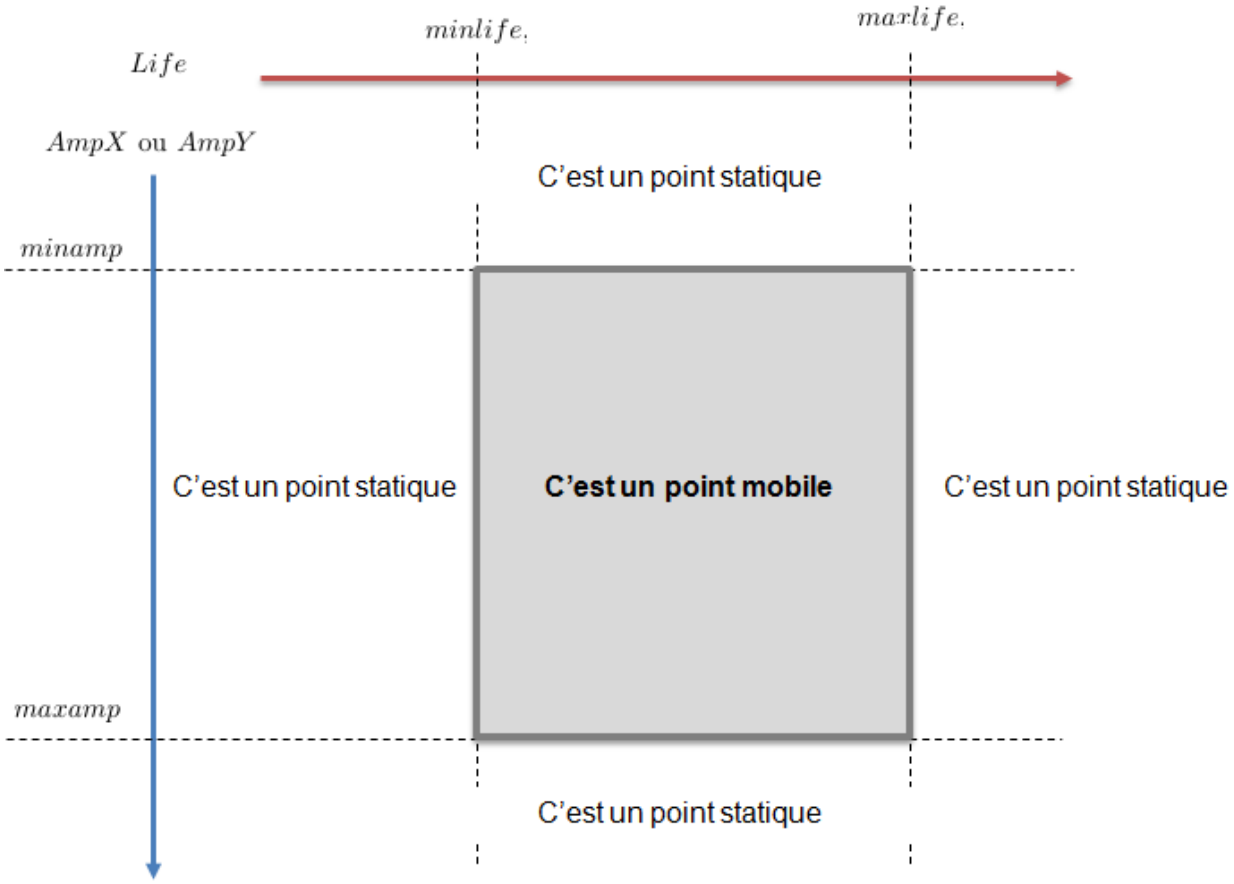


FIGURE 34 – Hypothèses de classification d'un point d'intérêt. Dans notre implémentation, un point dont les paramètres sont à l'intérieur de rectangle gris est classé un point mobile, sinon c'est un point de fond statique.

2. C'est la moyenne de toutes les valeurs enregistrés pour chaque paramètre sur les 100 premiers images de spécialisation

2.2 Algorithme de pondération selon la stratégie d'observation "Suivi KLT"

Pour pondérer une proposition du sous-ensemble 2 suite à l'étape prédiction, nous supposons qu'un échantillon positif est un vrai positif si sa ROI contient un nombre de points d'intérêts mobiles plus élevé que le nombre de points statiques de fond. Par contre, un échantillon négatif est un vrai négatif si sa ROI ne contient que des points caractéristiques statiques ou bien le nombre de ces point est nettement supérieur de celui des points mobiles.

L'Algorithme 5 résume le principe de pondération selon la stratégie de suivi KLT. Il prend en compte le type de point caractéristique dans la ROI associée à l'échantillon et l'étiquette prédite de ce dernier pour lui attribuer un poids. Le Tableau 6 présente les notations utilisées dans l'Algorithme 5.

Tableau 6 – Notations et fonctions utilisées dans Algorithme 5

Notation : Définition
- $\mathbf{p}$: Position d'une ROI associée à un échantillon dans la séquence vidéo cible ($\mathcal{D}^t$).
-$compute\_FRPts(\mathbf{p}_i, \{FPts_j\}_{j=1..,L})$: Fonction qui calcule les points d'intérêts de forme (points mobiles) à l'intérieur d'une ROI représentée par sa position $\mathbf{p}$.
-$compute\_BKPts(\mathbf{p}_i, \{FPts_j\}_{j=1..,L})$: Fonction qui calcule les points d'intérêts de fond (points statiques) à l'intérieur d'une ROI représentée par sa position $\mathbf{p}$.
-FR_FPts : Le nombre des points de forme dans une ROI.
-BK_FPts : Le nombre des points de fond dans une ROI.

Algorithme 5 Stratégie d'observation 2 : Suivi KLT

Entrée: Sous-ensemble 2 $\{\breve{\mathbf{X}}_{k+1}^{(n)}\}_{n=1,..,\breve{N}}$ avec les positions ROIs et les tailles associées dans la séquence vidéo cible $\{\mathbf{p}_i\}_{i=1,..,\breve{N}}$
La séquence vidéo cible et la base associée $\mathcal{D}^t$
Liste des données : $minlife$, $maxlife$, $minamp$ et $maxamp$
Liste des points d'intérêts $\{FPts_j\}$ de chaque image j, $\{FPts_j\}_{j=1..,L}$
Sortie: L'ensemble $\{\pi_i\}_{i=1..,\breve{N}}$ des poids attribués aux échantillons

---

Pour $i = 1$ to $\breve{N}$ **faire**
  $\pi_i \leftarrow 0$
  `/* Calcul et classification des points d'intérêts */`
  $FR\_FPts \leftarrow compute\_FRPts(\mathbf{p}_i, \{FPts_j\}_{j=1..,L})$
  $BK\_FPts \leftarrow compute\_BKPts(\mathbf{p}_i, \{FPts_j\}_{j=1..,L})$
  `/* Attribution des poids */`
  Si $((\breve{y}_i = piéton)\&(FR\_FPts > BK\_FPts))$ **alors**
    $$\pi_i \leftarrow \frac{FR\_FPts}{FR\_FPts + BK\_FPts}$$
  Sinon Si $((\breve{y}_i \neq piéton)\&(FR\_FPts < BK\_FPts))$ **alors**
    $$\pi_i \leftarrow \frac{BK\_FPts}{FR\_FPts + BK\_FPts}$$
  FinSi
FinPour

Un échantillon positif sera associé à un poids égal au nombre des points formes divisé par le nombre total (points de formes et de fond) des points enregistrés à l'intérieur de sa ROI si FR_FPts dépasse BK_FPts. Sinon, il sera associé à zéro. De même, un échantillon négatif sera associé à un poids égal au nombre des points de fond divisé par le nombre total (points de formes et de fond) des points enregistrés à l'intérieur de sa ROI si BK_FPts dépasse FR_FPts. Sinon, il aura un poids nul. Toute proposition d'échantillon associée à un poids nul sera rejetée lors de l'étape de ré-échantillonnage.

Conclusion

Nous avons exposé, dans ce chapitre, deux stratégies d'observation pour la pondération des échantillons cibles et la favorisation automatique des propositions qui possèdent des étiquettes correctes.

Dans la première section, nous avons décrit les deux scores de la stratégie d'indices spatio-temporels OAS. Ensuite, nous avons donné l'algorithme qui utilise les deux scores overlap_score et accumulation_socre pour attribuer un poids à chaque proposition.

Dans la deuxième section, nous avons présenté le principe de notre utilisation de la technique de suivi KLT pour la classification des points d'intérêts en points mobiles et en points statiques. Puis, nous avons exposé l'algorithme de pondération des échantillons selon la stratégie de suivi KLT.

Dans le chapitre suivant, nous allons présenter les expérimentations effectuées pour la validation de l'approche proposée.

Troisième partie

Expérimentations et Implémentation

Dans la première partie, nous avons situé cette thèse dans son cadre général. Ensuite, nous avons détaillé dans la deuxième partie : (i) Une contribution principale qui sert à spécialiser automatiquement un classifieur à base d'un filtre SMC. (ii) Deux autres contributions qui représentent les deux stratégies d'observation pour l'étape mise à jour de filtre. Dans cette partie, nous exposons une contribution pratique qui consiste à présenter une application de l'approche proposée pour la détection multi-objets.

Plus précisément, cette partie a pour objectif de valider notre approche de spécialisation et les deux stratégies d'observation présentées dans un cadre de détection multi-objets.

Cette troisième partie du manuscrit est composée également de deux chapitres. Le chapitre 5 décrit une série d'expérimentations permettant de valider les points clefs de notre approche telles que (i) l'amélioration de performance par rapport au détecteur générique, (ii) l'apport de notre approche comparant aux autres méthodes de l'état de l'art et (iii) la généricité de l'approche proposée pour intégrer différentes stratégies d'observation et pour spécialiser différents types de classifieurs. Le chapitre 6 présente principalement le logiciel OD SOFT de la société Logiroad, l'implémentation réalisée pour intégrer les détecteurs spécialisés fournis par notre approche dans OD SOFT et les résultats des comparaisons entre la méthode de détection développée initialement par Logiroad et les détecteurs spécialisés obtenus.

Chapitre 5

Expérimentations et Résultats

Introduction

Ce chapitre présente les différentes expérimentations réalisées et les résultats des tests effectués pour mettre en évidence l'apport de spécialisation proposée. Nous appliquons l'approche proposée pour la spécialisation d'un détecteur de piétons et d'un détecteur de voiture vers des séquences vidéos de trafic routier provenant d'une caméra statique.

Le déroulement de la spécialisation est tout d'abord rappelé dans la section 1. Dans la section 2, nous présenterons les différents détails des détecteurs génériques HOG-SVM crées. Ensuite, dans la troisième section, nous évaluons notre approche à travers plusieurs expérimentations tout en interprétant les résultats obtenus. Dans la quatrième section, nous comparons notre approche à d'autres méthodes de l'état de l'art. Nous terminons le chapitre par une démonstration de la généricité de la méthode via une spécialisation d'un détecteur à base d'apprentissage profond.

1 Spécialisation d'un détecteur vers une scène particulière

L'objectif principal de notre travail est de spécialiser un détecteur vers une scène spécifique. Pour ce faire, nous avons considéré les échantillons de la base de données d'apprentissage comme des réalisations de la distribution de probabilité conjointe entre les vecteurs caractéristiques d'échantillons et les classes d'objets. De cette façon, notre objectif peut être vu comme une estimation d'une distribution cible cachée en utilisant une distribution source dans laquelle nous avons un ensemble d'échantillons annotés, afin de donner une distribution cible estimée en sortie.

L'approximation de la distribution est réalisée à l'aide d'un processus récursif. Un schéma de synthèse correspondant est illustré dans la Figure 35. Au cours de la première itération, un détecteur générique est appliqué sur un ensemble d'images extraites de la scène cible pour chercher et proposer des échantillons cibles et des échantillons de fond. Ensuite, la pertinence des propositions est déterminée durant l'étape de mise à jour à l'aide d'une des stratégies d'observation (décrites dans le chapitre précédent 4) qui attribue un poids pour chaque échantillon retourné par le détecteur générique. L'étape de ré-échantillonnage consiste à faire un tirage par importance pour sélectionner des échantillons cibles possédant un poids élevé et pour désigner et choisir les échantillons sources visuellement proches des cibles sélectionnés. Les échantillons sélectionnés à la fois à partir de la base source et de la base cible sont combinés pour créer la première base spécialisée pour l'itération suivante.

Le processus est le même à une itération k différente de zéro, mais l'étape de prédiction utilise un détecteur spécialisé qui est entrainé sur la base de données spécialisée construite à l'itération précédente, pour proposer de nouveaux échantillons issus de la distribution cible. L'étape de ré-échantillonnage crée une base de données d'apprentissage à partir de la base spécialisée précédente, 2) la scène cible (images de la scène et modèles de fond pré-calculés) et 3) la base source.

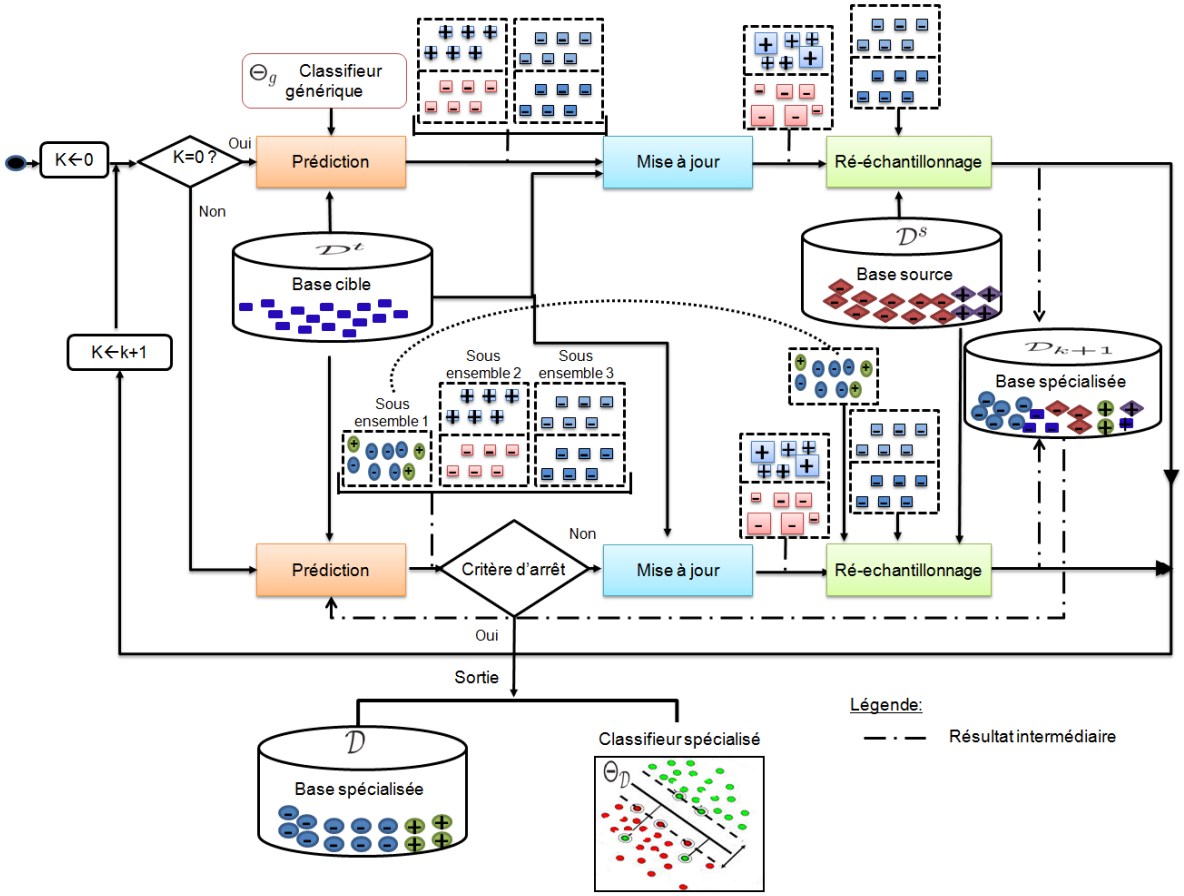


FIGURE 35 – Schéma général du processus de la méthode de spécialisation. La distribution cible est estimée par une base spécialisée à l'aide de filtre SMC.

La nouvelle base spécialisée approxime de mieux en mieux la distribution cible à travers les itérations. Une fois le critère d'arrêt atteint, nous aurons en sortie le dernier classifieur (ou détecteur) spécialisé et la base de données spécialisée associée.

2 Différents détecteurs génériques

Un détecteur HOG-SVM générique, tel qu'il est présenté dans la section 4 (page 40) du chapitre 2, est conçu suite à une phase d'apprentissage sur un grand nombre d'imagettes reparties en échantillons positifs présentant l'objet d'intérêt et d'autres échantillons négatifs pouvant avoir tout autre contenu sauf l'objet d'intérêt.

Dans notre cas, nous avons besoin d'un détecteur générique pour chaque objet de trafic routier recherché. Les objets de trafic recherché sont piétons, voitures, bus, camions, motos et vélos, et véhicules utilitaires. La littérature présente particulièrement deux bases génériques pour la détection de piétons qui sont INRIA Person Dataset et la base Caltech [Dollár *et al.*, 2009]. Par contre d'aprés notre recherche bibliograohique, il n'y a pas de grandes bases publiques pour les autres objets ou bien il y en a certaines de tailles très petites ou présentant des points de vues particuliers. Nous citons à titre d'exemple la base Caltech [Philip et Updike, 2001] et la base UIUC [Agarwal *et al.*, 2004] de voitures. La base Caltech présente des voitures en vue de face et en vue d'arrière seulement et la base UIUC contient des images en niveau de gris dont seulement 550 échantillons de voitures avec point de vue de profil (gauche <–> droite) et qui ont subi une normalisation particulière pour l'apprentissage.

Dans la sous-section 2.1, nous présentons un outil simple développé pour la création automatique d'une base générique. Ensuite, nous décrivons les différentes bases créées et les détecteurs associés dans la sous-section 2.2

2.1 Utilitaire de création de base générique

L'idée principale ici est de créer un logiciel utilitaire capable d'importer des vidéos et des dossiers d'images pour que l'utilisateur puisse englober l'instance de l'objet recherché par un rectangle afin de réaliser un découpage selon le rectangle dessiné et de sauvegarder l'imagette obtenue dans un dossier spécifique indiquant le ratio du rectangle qui a bien entouré l'objet. Ainsi ce dossier signale le point de vue de l'objet(ou bien l'orientation de l'objet) dans l'image : bas-haut, gauche-droite, incliné.

L'utilitaire vise a faciliter la collecte des imagettes qui répondent à certaines recommandations :

— Une imagette doit présenter l'objet d'intérêt recherché au centre.
— Une imagette doit contenir le minimum possible de fond.
— Une imagette doit posséder une taille parmi une liste des ratios possibles. Nous avons cinq ratios qui sont : 0.5, 1, 1.25, 1.6, 1.8. Ces ratios reflètent les dimensions de fenêtres 64×128, 64×64, 80×64, 128×80 et 112×64.

Il est à noter qu'un ratio dans notre cas représente le rapport entre la largeur d'une imagette et sa hauteur. La FIGURE 36 illustre l'interface de logiciel [1] développé.

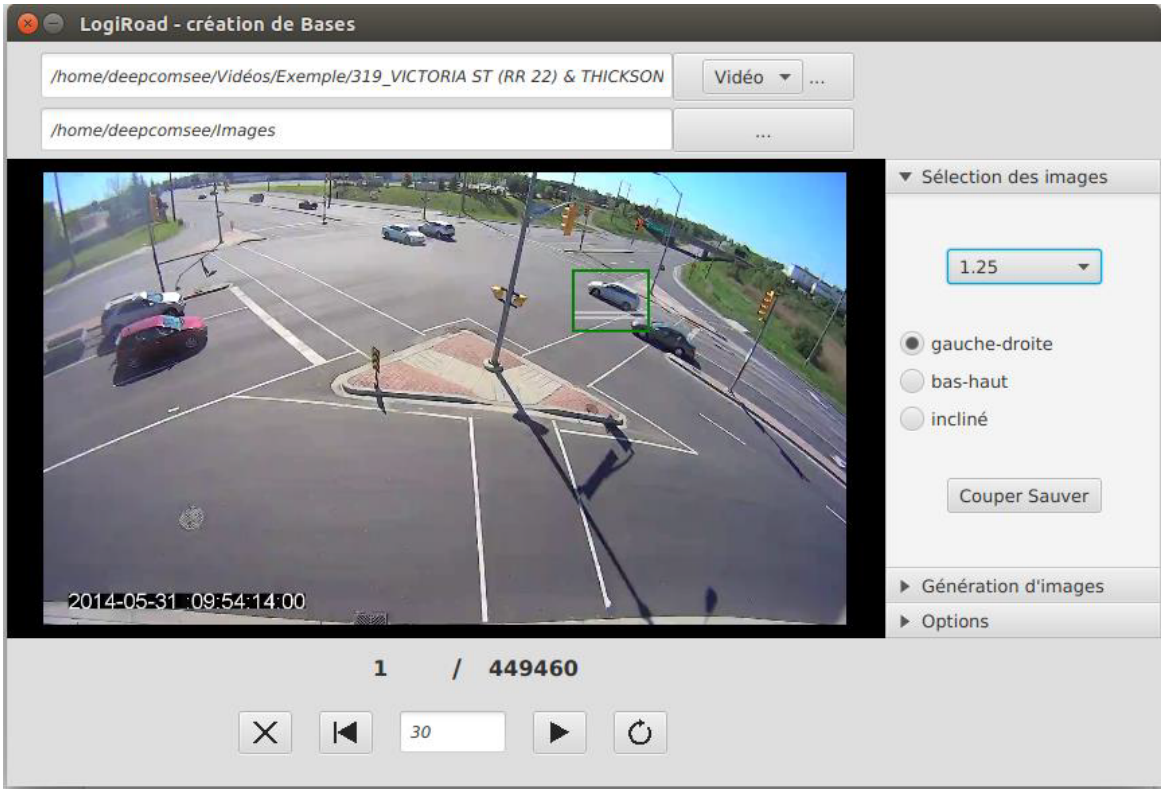


FIGURE 36 – Logiciel utilitaire de création de base données

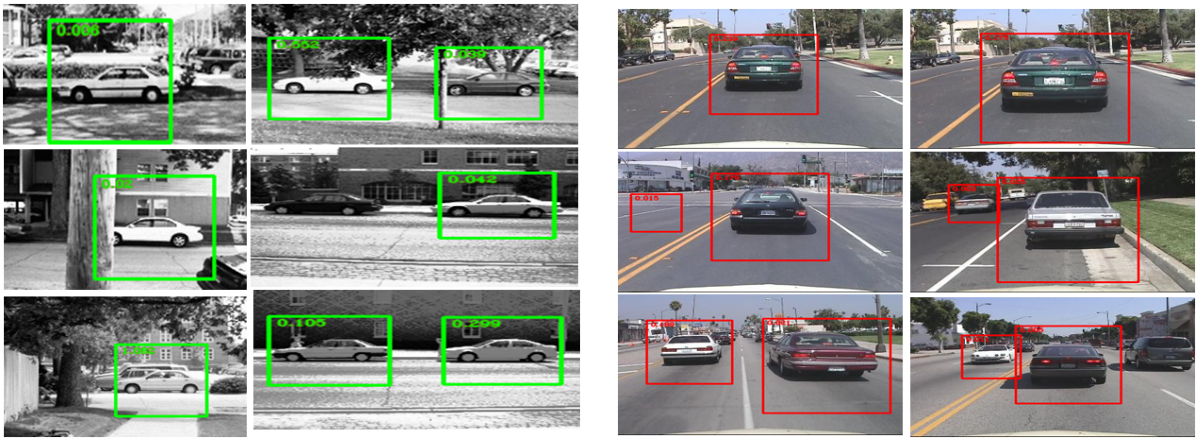
2.2 Différentes bases de données et détecteurs associés

Nous avons utilisé le logiciel (FIGURE 36) pour collecter une base d'apprentissage de voiture dont chaque imagette contient une voiture au centre et a un ratio égal à 1. Nous avons extrait les échantillons

1. Ce Logiciel est développé par Alexander Boyer lors d'un stage de 10 semaines

positifs à partir d'un ensemble de vidéos de la société Logiroad. Nous avons collecté 1050 imagettes sur lesquelles nous avons appliqué une réflexion horizontale pour avoir un nombre total de 2100 imagettes de voitures. Les échantillons négatifs sont extraits à partir des images négatifs de la base INRIA Person Dataset et de la base INRIA Car dataset [Carbonetto *et al.*, 2008].

Nous avons entraîné un détecteur de voiture tout en respectant le rapport entre les échantillons positifs (2 100) et négatifs (12 000) que Dalal et Triggs ont utlisé dans [Dalal et Triggs, 2005]. Ensuite, nous avons effectué une étape bootstrap sur les images négatives de la base INRIA Person dataset et nous avons ré-entrainé le détecteur sur la première base et les exemples difficiles issus de l'étape de bootstrap. Les figures 37a et 37b illustrent respectivement, des exemples de détections effectuées par notre détecteur de voiture source sur les bases de données de voitures UIUC [Agarwal *et al.*, 2004] et Caltech 2001 (rear)[Philip et Updike, 2001].



(a) La base UIUC (b) La base Caltech 2001 (rear)

FIGURE 37 – Résultat de détections à l'aide de détecteur générique HOG-SVM de voiture

Nous avons utilisé également le logiciel décrit dans la section 2.1 pour créer d'autres bases d'imagettes de voitures, camions, bus, motos et vélos, et véhicules utilitaires que nous utilisons pour entraîner des détecteurs génériques. Le Tableau 7 résume le nombre d'imagettes collectées pour chaque catégorie selon le ratio (rapport) correspondant. Au total, nous avons un ensemble de quatre détecteurs génériques pour ce qui concerne les voitures et deux détecteurs génériques pour les bus, les camions, les motos&vélos, et les véhicules utilitaires.

Tableau 7 – Les différentes bases génériques par rapport d'imagettes

Rapports	0.5	1	1.25	1.6	1.8
Bus			1000		1002
Camions			1038		1012
Motos&Vélos	1002	1032			
Voitures		2100	4954	3158	7200
Utilitaires			1004		2010
Négatifs	12000	12000	12000	12000	12000

3 Évaluation de la spécialisation proposée

Dans cette section, nous nous intéressons à valider les performances de la spécialisation à travers un certains nombre de tests. Dans la suite de ce chapitre, l'indication d'un taux de détection correspondra toujours à un seul faux positif par image (FPPI = 1).

Dans nos expérimentations, nous avons utilisé l'implémentation SVMLight [2], pour l'apprentissage des détecteurs HOG-SVM génériques et spécialisés. Nous avons considéré la vérité du terrain fournie par Wang et Wang dans [Wang et Wang, 2011] (notée MIT_P) et par Wang et al. (notée CUHK_P) dans [Wang *et al.*, 2012a], pour tester les résultats de la détection des piétons sur la base de données MIT Traffic et la base de données CUHK_Square, respectivement. Pour tester la détection des voitures, nous avons proposé des annotations relatives aux voitures présentes dans les images de test de la base MIT Traffic et la base Logiroad Traffic. Nous avons noté ces derniers MIT_C et LOG_C, respectivement.

Les expérimentations réalisées pour l'évaluation de notre approche sont regroupées en cinq catégories. La première catégorie étudie l'effet de la variation du paramètre α_t et valide le critère d'arrêt du processus de spécialisation à travers l'étude de la convergence de la méthode. La deuxième catégorie de tests effectués sert à identifier l'effet de la nature des échantillons sélectionnés. Nous évaluons la performance de la spécialisation en fonction de la stratégie de collecte des échantillons à partir de la scène cible. La troisième catégorie d'expérimentations évalue la performance de l'approche en fonction de la stratégie d'observation utilisée pour la pondération des échantillons cibles.

Dans la quatrième catégorie d'expérimentations, nous comparons la meilleure performance obtenue par notre approche avec les performances d'autres méthodes de l'état de l'art. La dernière catégorie d'expérimentations est faite pour étudier la généricité de l'approche spécialisée. Elle se résume dans un test de spécialisation d'un détecteur basé sur l'apprentissage profond et une comparaison entre les performances de spécialisation d'un détecteur HOG-SVM et un détecteur Faster R-CNN. Le tableau 8 résume les objectifs visés de chacune des catégories d'expérimentations et les tests effectués.

2. http ://svmlight.joachims.org/

Tableau 8 – Les différentes catégories d'expérimentations

	Objectifs	Tests effectués
Catégorie 1	- Détermination de nombre d'échantillons à propager d'une itération à une autre	- Variation du valeur de α_t et comparaison des performances des détecteurs obtenus
	- Étude de convergence au fil des itérations - Détermination de critère d'arrêt du processus de spécialisation	Comparaison des performances des détecteurs à travers les itérations -Calcul de divergence D_KL entre les échantillons positifs extraits manuellement et les échantillons positifs de la base spécialisée à travers les itérations -Suivi de la variation du nombre d'échantillons de sous-ensemble 2
Catégorie 2	Étude d'effet des échantillons de fonds	- Comparaison entre les cas où la base spécialisée contenant des échantillons de fond SMC_B et les cas sans ces échantillons SMC_WB
Catégorie 3	Étude de généricité de la méthode à travers l'intégration de différentes stratégies d'observation	- Comparaison des performances suites à différentes stratégies
Catégorie 4	Étude de l'apport de la spécialisation par rapport à l'état de l'art	Comparaison de performances avec d'autres algorithmes
Catégorie 5	Étude de généricité de la méthode proposée à travers la spécialisation de différents détecteurs	- Spécialisation d'un Faster R-CNN -Comparaison entre la performance de spécialisation de HOG-SVM et de Faster R-CNN

3.1 Effet de paramètre α_t

Le sous-ensemble 1 produit dans l'étape de prédiction sert à sauvegarder une mémoire sur l'état de la distribution cible estimée à travers les itérations. Il est à noter que dans un cas de filtre SMC utilisé pour le suivi l'effet de mémoire existe par défaut parce que le déplacement d'un objet d'une position à une autre garde une trace d'évolution ou de dégradation par rapport à l'estimation précédente.

Le nombre d'échantillons qui forme ce sous-ensemble 1 est ajusté à l'aide de paramètre α_t. Nous avons fait cinq tests tout en faisant varier la valeur de α_t (en pourcentage) pour déterminer le nombre d'échantillons à propager d'une itération à une autre. Nous avons résumé les résultats obtenus dans le Tableau 9.

Tableau 9 – Performance de détection de différents détecteurs pour différentes valeurs du paramètre α_t à FPPI = 1

α_t	**0.1**	**0.25**	**0.5**	**0.75**	**0.9**
Performance en %	67,2	70,2	73,9	**77**	75,38

Ce tableau présente les performances des détecteurs pour différentes valeur de α_t. Il présente un maximum pour une valeur de α_t égale à 0.75. A partir de ces résultats, nous constatons qu'il sera mieux

de garder 75% des échantillons de la base spécialisée créée à l'itération précédente et de renouveler 25% avec des échantillons à sélectionner des deux autres sous-ensembles 2 et 3.

3.2 Évaluation de convergence

Dans cette sous-section, nous utilisons la base CUHK_square pour décrire les différents tests effectués et les résultats enregistrés.

Afin de déterminer le critère d'arrêt du processus de spécialisation, nous avons étudié la convergence de notre approche à l'aide de trois tests. Au cours du premier, nous comparons les performances des détecteurs spécialisés à travers les itérations. Dans le deuxième test, nous calculons la divergence entre des échantillons extraits manuellement et les échantillons positifs de la base spécialisée. Dans le troisième test, nous avons étudié la variation du nombre d'échantillons collectés pour le sous-ensemble 2 au fil des itérations.

La comparaison des performances de détecteur spécialisé à plusieurs itérations et les performances du détecteur générique montre que notre approche de TTL-SMC génère une augmentation dans le taux de détection dés la première itération. La FIGURE 38 montre que les performances des détecteurs spécialisés passent de 26,6% à 60% à la première itération, et de 60% à plus de 70% à la quatrième itération. Les expériences montrent que la performance est légèrement améliorée pour un FPPI inférieur ou égale 0.6 et légèrement diminuée pour un FPPI supérieur à 0.6 au cours des itérations 5 à 10. Dans la FIGURE 38, nous avons limité la visualisation du courbe ROC à la dixième itération pour des raisons de clarté et pour illustrer le gain de performance obtenu après six itérations en plus de la quatrième itération qui a marqué l'itération d'arrêt dans nos expériences.

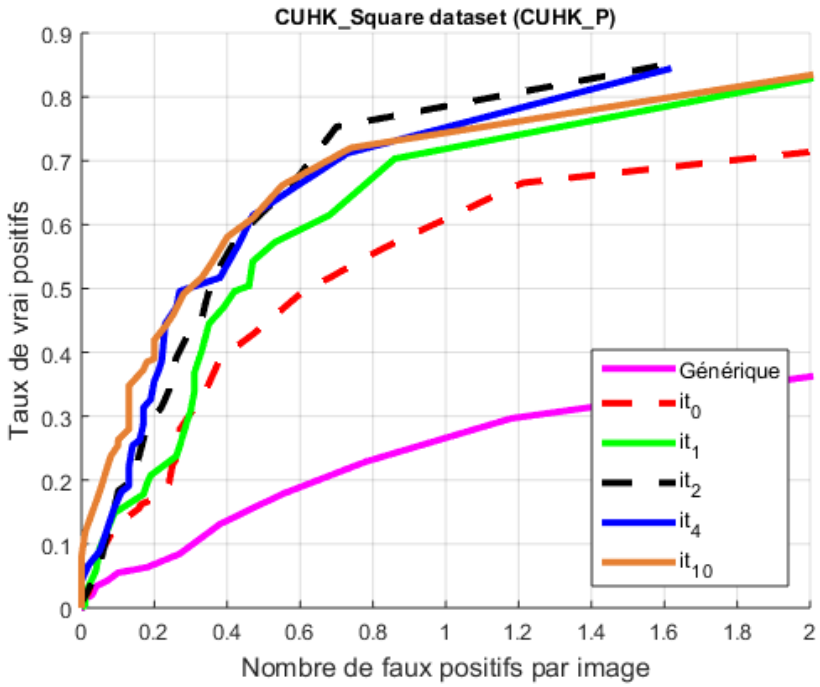


FIGURE 38 – Comparaison des performances entre le détecteur générique et les détecteurs spécialisés à travers plusieurs itérations, sur les images de test de la scène CUHK_Square

Nous avons utilisé également une autre métrique d'évaluation, la divergence de Kullback-Leibler (D_KL), pour mesurer la convergence de la distribution estimée vers la vraie distribution cible. Nous avons calculé la D_KL entre un ensemble de piétons extraits manuellement à partir des images de spécialisation et des échantillons positifs de la base de données spécialisée produite à chaque itération. Le calcul de D_KL entre deux ensembles d'échantillons est fait conformément au travail de Boltz *et al.* [Boltz *et al.*, 2009].

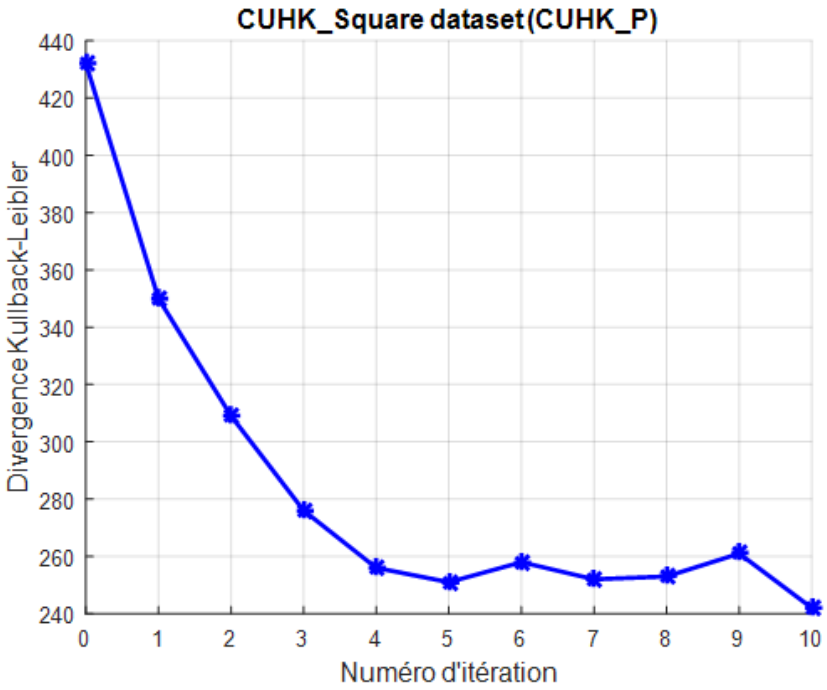


FIGURE 39 – Divergence D_KL entre les échantillons positifs de la base spécialisée à chaque itération et les instances de piétons des images de spécialisation

La FIGURE 39 montre que la D_KL entre les échantillons positifs de la base spécialisée et les instances de piétons de la partie spécialisation de la scène CUHK diminue au cours des itérations jusqu'à attendre une valeur relativement stable à partir de l'itération 4 qui correspond à l'itération d'arrêt.

Toujours dans le but de déterminer le critère d'arrêt, nous avons étudié la variation du nombre d'échantillons du sous-ensemble 2 retourné à chaque itération. La FIGURE 40 indique que le nombre d'échantillons du " sous-ensemble 2" se stabilise à partir de l'itération 4. Cette dernière marque la validation du critère d'arrêt.

La convergence de notre processus de spécialisation est pratiquement déterminée lorsque le paramètre α_s atteint la valeur 0.8. Ce paramètre α_s reflète le rapport entre le nombre d'échantillons du sous-ensemble 2 retourné par le détecteur à l'itération courante et le nombre d'échantillons du sous-ensemble 2 retourné par le détecteur de l'itération précédente.

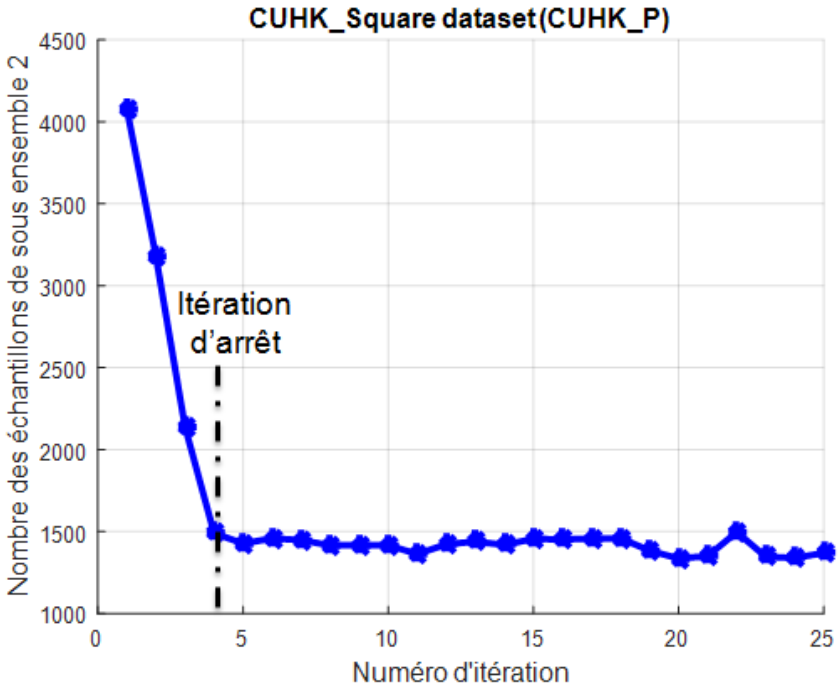


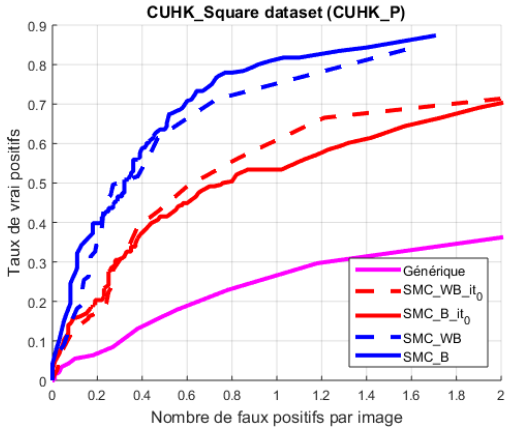
FIGURE 40 – Nombre d'échantillons du sous-ensemble 2 au cours des itérations

3.3 Évaluation des stratégies de proposition d'échantillons

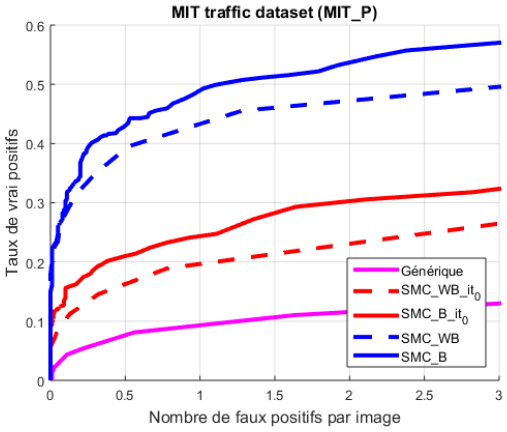
L'étape de prédiction consiste principalement à proposer des échantillons pour créer la base spécialisée. Dans la sous-section 3.1, nous avons étudié l'effet de la variation du nombre d'échantillons du sous-ensemble 1 pour garder une mémoire de la distribution estimée à travers les itérations et dans cette sous-section nous étudions la composition de toute la base. Nous avons testé deux stratégies de proposition d'échantillons : la première stratégie, notée SMC_B, propose une base composée de trois sous-ensembles selon la description faite dans la sous-section 2.3, page 59. Et une deuxième stratégie, notée SMC_WB, forme une base composée à partir des deux premiers sous-ensembles sans utiliser des échantillons extraits des modèles de fond.

Pour bien évaluer l'apport des échantillons proposés par ces deux stratégies : (i) Nous avons utilisé la stratégie d'indices spatio-temporels OAS comme une stratégie d'observation. (ii) Nous avons spécialisé deux détecteurs ; un détecteur de piétons et un détecteur de voitures. (iii) Nous avons fait le test sur deux bases pour chaque détecteur. La FIGURE 41 présente les résultats obtenus, uniquement les performances du détecteur spécialisé à la première et à la dernière itération sont reportées et comparées à la performance du détecteur générique. Les Figures 41a et 41b illustrent respectivement les performances de détection de piétons sur la base CUHK_Square et la base MIT Traffic. Ces figures nous permettent de constater que l'utilisation d'échantillons du sous-ensemble 3 (extraits des modèles de fond pré-calculés automatiquement) conduisent à une amélioration de 6% (à un FPPI =1) du taux de détection final des piétons sur les deux bases par rapport aux performances enregistrées avec uniquement sous-ensemble 1 et sous-ensemble 2.

Pour le cas de la détection des voitures, nous avons noté que les deux stratégies (SMC_B et SMC_WB) donnent des résultats comparables sur la base MIT Traffic (FIGURE 41c). Pour le test sur la base Logiroad Traffic visualisé par la FIGURE 41d, les courbes ROC du taux de détection montrent que les deux stratégies ont la même performance à un FPPI = 1 à la première itération, mais la stratégie SMC_B s'améliore de 19% en performance par rapport à la SMC_WB à l'itération de convergence.

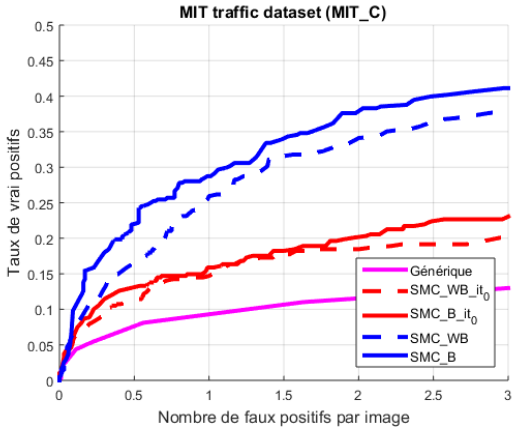


(a) CUHK_Square dataset

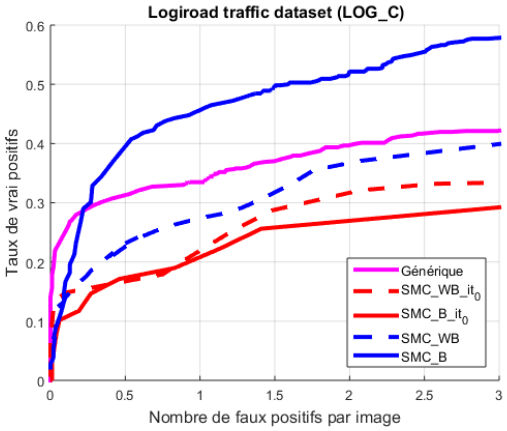


(b) MIT Traffic dataset

(a) & (b) Détection de piétons



(c) MIT Traffic dataset



(d) Logiroad Traffic dataset

(c) & (d) Détection de voitures

FIGURE 41 – Comparaison des performances des stratégies de proposition d'échantillons

Par ailleurs, nous avons remarqué que le processus de spécialisation converge pratiquement avec le même nombre d'itérations dans la plupart des cas, mais la stratégie SMC_B nécessite un peu plus de temps supplémentaire à une itération donnée. Le Tableau 10 décrit le temps moyen nécessaire d'une itération de spécialisation par rapport à la stratégie utilisée (sélection d'échantillons et apprentissage de détecteur) sur une machine avec un processeur Intel (R) Core i7-3630QM 2.4G pour chaque base de données tout en indiquant le nombre et la taille d'images traitées.

Tableau 10 – Durée moyenne d'une itération de spécialisation par rapport à la stratégie de proposition d'échantillons et selon différentes bases de données

Base	Nb. images	Taille	SMC_WB	SMC_B
CUHK_Squre	352	1440×1152	60 min	84 min
MIT Traffic (Piétons)	420	4320×2880	210 min	285 min
MIT Traffic (Voitures)	420	720×480	14 min	28 min
Logiroad Traffic	600	864×486	22 min	36 min

3.4 Évaluation des stratégies d'observation

Nous comparons les deux stratégies d'observation (décrites dans le chapitre précédent) utilisées pour la pondération des échantillons de sous-ensemble 2. Une stratégie d'indices spatio-temporels OAS qui calcule un Overlap_score et un accumulation_score pour chaque échantillon afin d'attribuer un poids qui permet la sélection des propositions correctes. Et une stratégie de suivi KLT qui fait la classification des points d'intérêts à l'intérieur de la *ROI* associée à chaque échantillon en un point mobile ou un point statique. Ensuite, en fonction du nombre de points et de leurs classes un poids est calculé pour chaque proposition.

Cette comparaison permet de visualiser le gain en performance du détecteur spécialisé par rapport à celle du détecteur générique et la généricité de l'approche de spécialisation proposée qui peut intégrer différentes stratégies d'observation.

Afin d'évaluer les performances des deux stratégies d'observation, nous adoptons la stratégie SMC_B dans l'étape de prédiction. Cette dernière a donné la meilleure performance dans les différents tests présentés dans la sous-section 3.3. Nous notons par SMC_B_OAS un détecteur spécialisé en appliquant la stratégie SMC_B comme une stratégie de proposition d'échantillons et la stratégie OAS comme une stratégie d'observation. En outre, SMC_B_KLT indique que c'est la stratégie SMC_B est utilisée dans l'étape prédiction et la stratégie de pondération selon le suivi KLT est utilisée dans l'étape mise à jour.

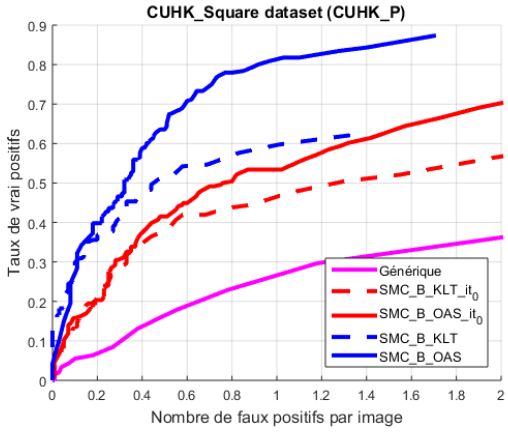
La FIGURE 42 examine l'efficacité des deux stratégies d'observation et compare les performances des détecteurs spécialisés aux performances du celui générique. Les figures 42a et 42b représentent les résultats de la détection des piétons sur la base CUHK_Square et la base MIT Traffic, respectivement. Les figures 42c et 42d présentent les résultats de la détection des voitures sur la base MIT Traffic et Logiroad Traffic.

Les figures 42a, 42b et 42c montrent que le détecteur spécialisé à l'aide de notre approche, génère une augmentation dans le taux de détection dés la première itération avec les deux stratégies d'observation utilisées. Cependant, la FIGURE 42d illustre une diminution à la première itération.

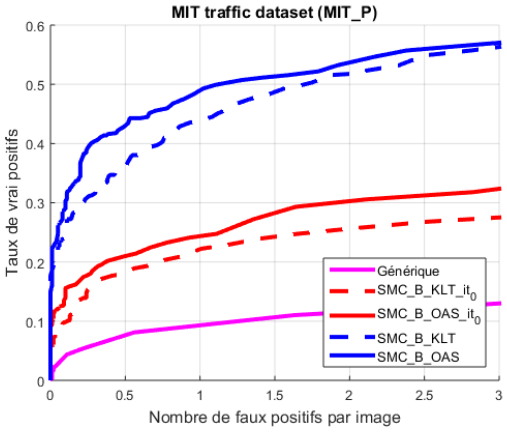
En particulier, la performance du détecteur SMC_B_OAS spécialisé (cas de la base CUHK_Square dans la FIGURE 42a) dépasse celle du générique de plus de 27% dès la première itération. De plus, les courbes montrent que la spécialisation converge avec un taux de vrais positifs égal à 81%. Par contre, la spécialisation du détecteur SMC_B_KLT a donné une augmentation de 34% du taux de détection par rapport à celui du détecteur générique.

Pour la base MIT Traffic, cas des piétons (FIGURE 42b), le détecteur SMC_B_OAS a amélioré la détection de 10% à 24% à la première itération et il a convergé à partir de la quatrième itération avec 49% de bonnes détections. Cependant, le détecteur SMC_B_KLT converge avec une augmentation de 32% par rapport aux performances du détecteur générique. Dans le cas de détection des voitures (FIGURE 42c), nous avons enregistré, pour les deux détecteurs SMC_B_OAS et SMC_B_KLT une augmentation du taux de détection de 5% à la première itération, par rapport à celle du détecteur générique. Le taux de détection du SMC_B_OAS augmente d'environ 30% à la quatrième itération contre une augmentation de 9% à 24% enregistrée par le détecteur SMC_B_KLT.

Concernant la base de données Logiroad (FIGURE 42d), le détecteur générique présente un taux de détection égal à 32%. Par contre, notre détecteur SMC_B_OAS spécialisé a donné un taux de détection égal à 20% à la première itération, puis a convergé avec 45% à la quatrième itération. Et la performance du détecteur SMC_B_KLT a diminué jusqu'à 16% à la première itération et passe ensuite à 47% à l'itération d'arrêt. Nous pouvons expliquer cette chute de performance à la première itération par l'injection de certaines instances des voitures comme des échantillons négatifs dans la base spécialisée. Cette injection est due à l'échec de nos stratégies à pondérer les voitures qui sont temporairement stationnaires. Ces instances sont détectées par le détecteur mais la stratégie d'observation a favorisé les propositions avec les fausses étiquettes, ce qui a perturbé le processus de spécialisation.

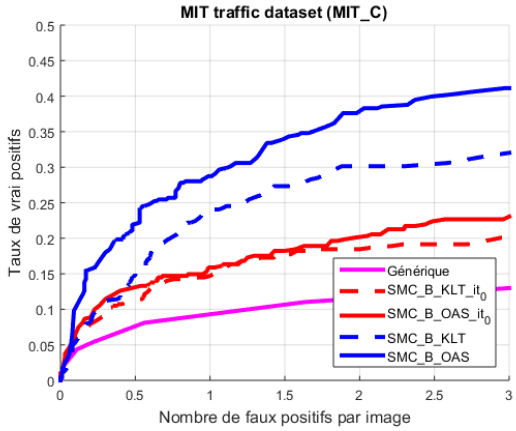


(a) CUHK_Square dataset

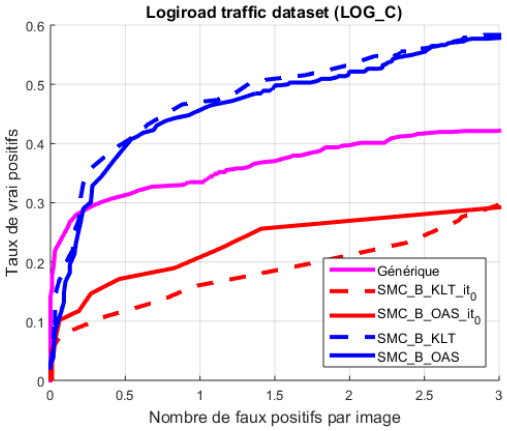


(b) MIT Traffic dataset

(a) & (b) Détection de piétons



(c) MIT Traffic dataset



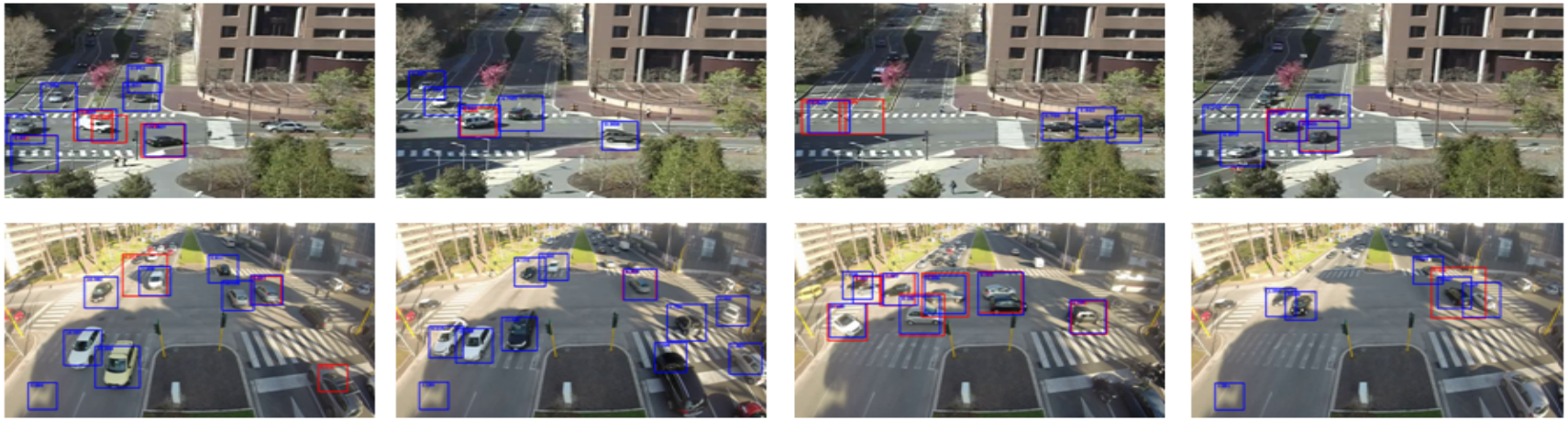
(d) Logiroad Traffic dataset

(c) & (d) Détection de voitures

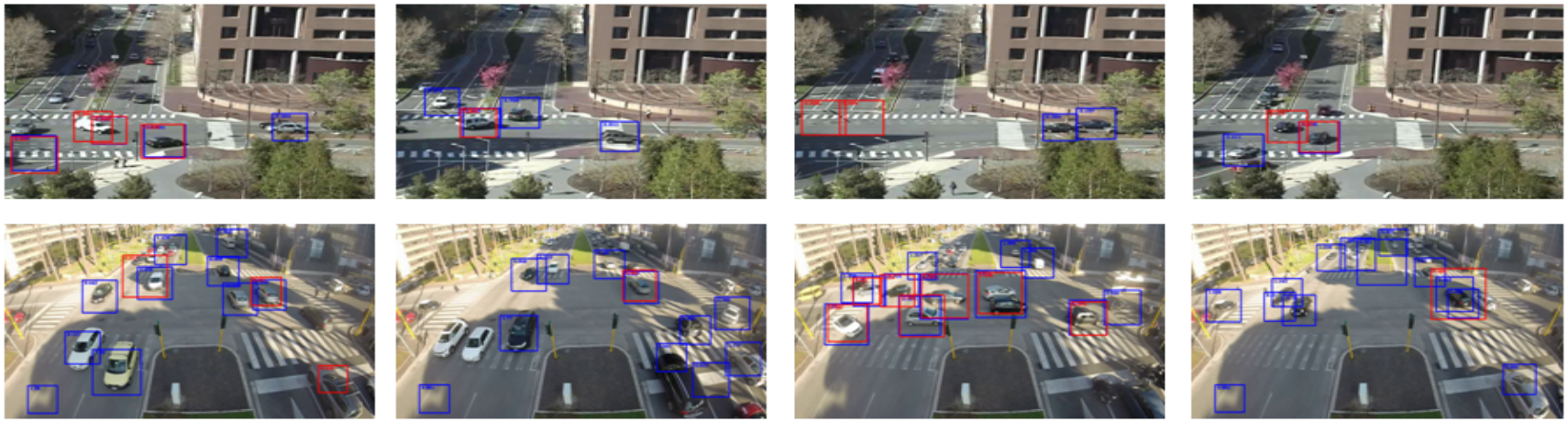
FIGURE 42 – Comparaison des performances des stratégies d'observation

Nous rappelons que la performance monte faiblement après la quatrième itération correspondant à l'itération d'arrêt dans nos expériences. Ainsi, nous avons remarqué des performances légèrement faibles dans la plupart des taux de détection finale du détecteur SMC_B_KLT, par rapport à ceux enregistrés par le détecteur SMC_B_OAS. Mais, il est clair qu'il y a une amélioration générée par notre spécialisation proposée, dans les deux figures 41 et 42, indépendamment des stratégies utilisées à l'étape de prédiction ou l'étape de mise à jour.

En outre, les courbes ROC des détecteurs de voiture présentent une amélioration légère des taux de détection suite aux itérations de la spécialisation sur les deux bases de données MIT Traffic et Logiroad Traffic. Ceci est noté pour les deux stratégies d'observation et peut être expliqué par la difficulté d'avoir un taux de chevauchement supérieur ou égal à 0.5 entre la vérité du terrain et la fenêtre carrée de détection. Nous avons choisi d'utiliser une fenêtre de ratio 1 pour détecter à la fois des voitures avec vue gauche-droite (ou vue de profil) et vue bas-haut (ou dite vue face-arrière). La FIGURE 43 visualise des résultats de détection de voitures sur les deux bases MIT Traffic et Logiroad Traffic. Cette figure compare visuellement les performances du détecteur générique et celles des détecteurs spécialisés relatives aux deux stratégies d'observation.



(a) Stratégie d'observation 1 : OAS



(b) Stratégie d'observation 2 : Suivi KLT

FIGURE 43 – Illustration de détection de voitures ; détecteur générique (rouge) et détecteur spécialisé (bleu). La base MIT Traffic (en haut) et la base Logiroad Traffic (en bas)

3.5 Combinaison des deux stratégies d'observation

Nous avons ajouté un autre test où nous avons appliqué simultanément les deux stratégies d'observation sur l'ensemble des échantillons rendus par l'étape de prédiction selon la stratégie SMC_B. Ensuite, nous combinons les données pondérées comme une seule entrée à l'étape de ré-échantillonnage.

Tableau 11 – Performance de détection (en %) de plusieurs détecteurs spécialisés selon la stratégie d'observation utilisée (pour FPPI=1)

Specialised detector			Generic	OAS	KLT	Fusion
Pedestrian	CUHK	it_f	26.6	53.7	46.5	**66.5**
		it_c		**81.3**	59.6	76.7
	MIT	it_f	10	24.2	22	**26.3**
		it_c		**49**	44.1	45.8
Car	MIT	it_f	9	15.8	14.7	**17.2**
		it_c		28.7	23.8	**31.5**
	Logiroad	it_f	33.5	20.8	16	**25.8**
		it_c		45.6	**47**	46.8

Le Tableau 11 compare le taux de détection de plusieurs détecteurs spécialisés et celui du détecteur générique avec un seul faux positif par image. Les notations OAS, KLT et Fusion correspondent respectivement à la stratégie d'observation 1 : OAS, à la stratégie d'observation 2 : suivi KLT et à leur combinaison. En outre, nous avons utilisé it_f et it_c pour désigner la première itération et l'itération de convergence.

Le Tableau 11 prouve une nouvelle fois que l'approche de spécialisation proposée est générique et qu'elle peut être appliquée en utilisant toute stratégie d'observation. Le tableau montre également que la combinaison des deux stratégies d'observation améliore légèrement la performance du détecteur spécialisé, mais dans certains cas, une seule stratégie d'observation donne un meilleur taux de détection que celui obtenu par la Fusion.

4 Comparaison avec l'état de l'art

Dans notre travail, nous supposons que la scène cible est surveillée par une caméra statique. Cette hypothèse nous aide à extraire nos indices contextuels. Tout en prenant en compte l'hypothèse fixée, nous avons besoin de séquences vidéo annotées, qui sont enregistrées par une caméra fixe afin de comparer notre approche aux algorithmes de l'état de l'art.

Dans cette sous section, nous évaluons la performance globale de la spécialisation proposée dans un cas de détection de piétons. Cette évaluation est réalisée par rapport aux méthodes de l'état de l'art sur deux bases : CUHK_Square et MIT Traffic.

Les principaux détecteurs d'état de l'art utilisés pour l'évaluation sont :

- **Générique [Dalal et Triggs, 2005] :** Un détecteur HOG-SVM qui a été entrainé sur la base générique INRIA Person dataset avec les détections de l'étape bootstrap tout en suivant la méthode de Dalal et Triggs dans [Dalal et Triggs, 2005].
- **Détecteur avec étiquetage de données manuel :** Ce détecteur sera noté par la suite "Det_manu". C'est un détecteur HOG-SVM cible qui a été entrainé sur un ensemble d'échantillons cibles. Ce dernier est composé de tout l'ensemble des piétons (qui sont extraits manuellement des images de spécialisation) et un ensemble d'échantillons négatifs extraits également des images de spécialisation en tenant compte qu'il n'y a pas de chevauchement avec les boîtes englobantes des piétons.

- **Nair 2004** [Nair et Clark, 2004] **:** Il s'agit d'un détecteur HOG-SVM qui est créé de manière similaire à celle proposée dans [Nair et Clark, 2004], mais le descripteur HOG a été utilisé comme vecteur de primitives et le classifieur SVM à la place du classiffieur Winnow. Ce détecteur est la sortie d'une approche d'adaptation automatique qui sélectionne un ensemble d'échantillons cibles à ajouter dans la base d'apprentissage initiale tout en se basant sur le retour d'un algorithme d'extraction fond-forme.
- **Wang 2014** [Wang *et al.*, 2014b] **:** Un détecteur spécifique à la scène cible qui est entrainé sur des échantillons de la base INRIA et des échantillons extraits et étiquetés automatiquement à partir des images de la scène cible. Les échantillons sources et cibles possédant des scores de confiance élevés seront sélectionnés. Les scores sont calculés en se basant sur plusieurs indices contextuels et la sélection sera effectuée par la méthode dite "confidence-encoded SVM" qui favorise les échantillons avec score élevé en pénalisant gravement l'erreur de classification de ces échantillons lors de la phase d'apprentissage. Cette variante de SVM intègre le score de confiance dans la fonction objective du classifieur.
- **Mao 2015** [Mao et Yin, 2015] **:** Un détecteur entrainé sur des échantillons cibles étiquetés automatiquement en utilisant des petites chaînes de suivi dites "Tracklets" et en propageant de l'information à partir des Tracklets étiquetées vers celles non-étiquetées où il y a une incertitude dans l'étiquette attribuée.

Dans un premier temps, nous interprétons les résultats de ces détecteurs sur la base CUHK_Square et la base MIT Traffic. Dans un deuxième temps, nous discutons la comparaison des mêmes détecteurs à travers les bases.

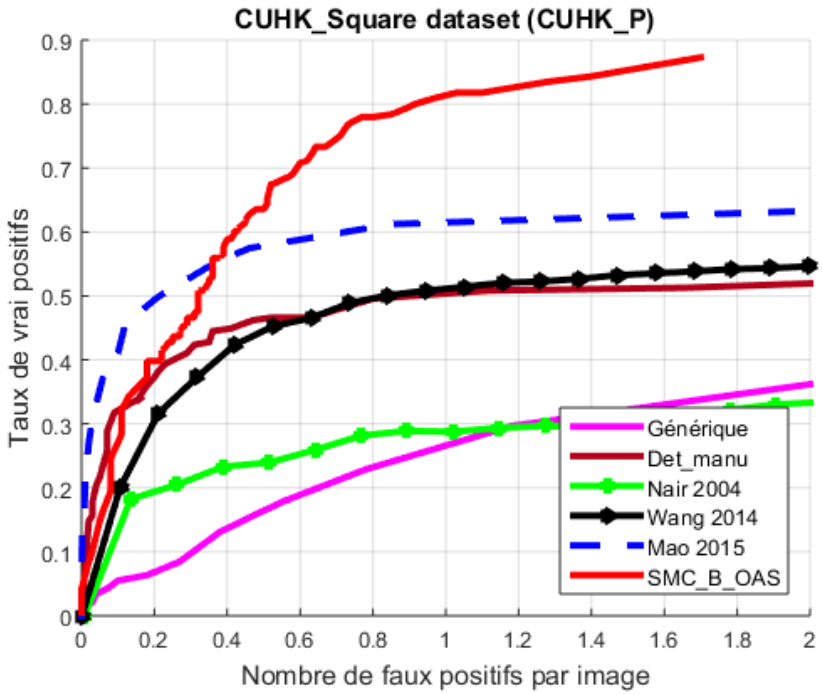


FIGURE 44 – Performances globales sur la scène CUHK_Square : Comparaison de SMC_B_OAS avec d'autres détecteurs de l'état de l'art

La FIGURE 44 montre que le détecteur spécialisé SMC_B_OAS dépasse nettement le générique sur la base CUHK_Square. Sa performance atteint 81% contre 26,6% enregistré par le détecteur générique. Le SMC_B_OAS surmonte aussi le détecteur "Det_manu" qui est entrainé avec des échantillons cibles étiquetés manuellement, avec une augmentation d'environ 31% à un FFPI = 1. Par contre, ce dernier

dépasse légèrement notre détecteur spécialisé pour un FPPI inférieur à 0.2.

Ainsi, notre détecteur SMC_B_OAS dépasse également les trois autres détecteurs spécialisés de Nair 2004, Wang 2014 et Mao 2015 respectivement par 45,57%, 23,25% et 20%. Il est à noter que Mao 2015 est meilleur que le détecteur spécialisé SMC_B_OAS pour un FFPI inférieur à 0.4.

Pour la base de données MIT Traffic (FIGURE 45), le taux de détection augmente de 10% à 47%. Également, le détecteur spécialisé SMC_B_OAS à la scène MIT dépasse par 21% le détecteur "Det_manu". Comparant par rapport au détecteur de Nair 2004, notre détecteur SMC_B_OAS spécialisé donne un meilleur taux de détection que celui proposé par Nair et Clark.

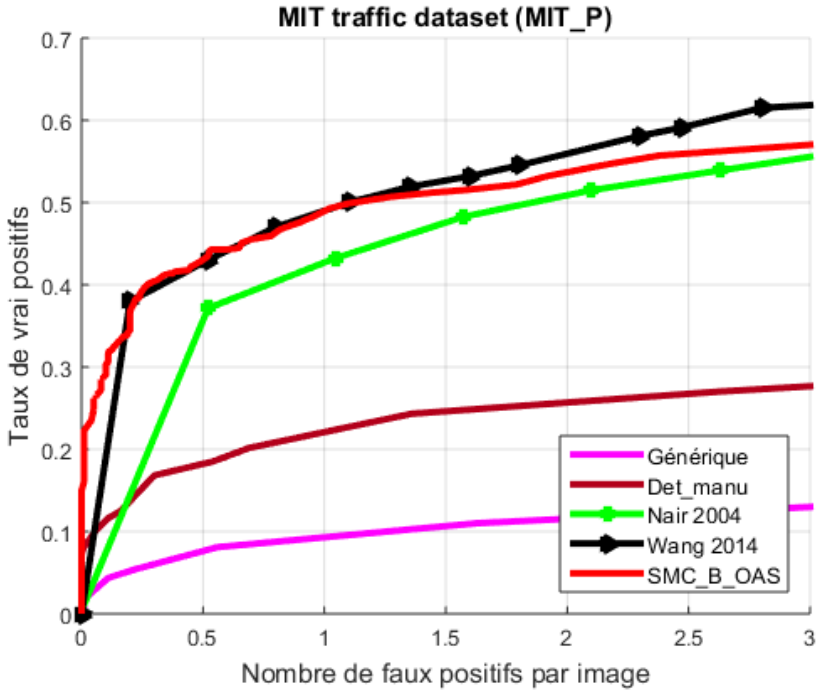


FIGURE 45 – Performances globales sur la scène MIT Traffic : Comparaison de notre SMC_B_OAS avec d'autres détecteurs de l'état de l'art

Les courbes ROC de notre détecteur spécialisé SMC_B_OAS et le détecteur spécialisée de Wang montrent que ces deux détecteurs ont des taux de détection très similaires. Néanmoins, il est nécessaire de mentionner que les ombres présentes dans la scène MIT, ont perturbé la pondération et la sélection des échantillons positifs cibles au moment de nos expérimentations.

Nous résumons dans le Tableau 12, le taux de détection des piétons de nos détecteurs spécialisés et plusieurs détecteurs de l'état de l'art en fonction de la base de données. Nous donnons également le gain entre notre détecteur spécialisé SMC_B_OAS et le détecteur générique dans la dernière ligne du tableau.

Afin de comparer la performance du même détecteur à travers les bases, nous illustrons, dans la FIGURE 46, les résultats des détecteurs génériques, des détecteurs de Wang 2014 et nos détecteurs spécialisés SMC_B_OAS selon les bases de données. Nous limitons la visualisation à ces trois détecteurs pour garder la clarté de l'image.

La FIGURE 46 montre que le détecteur générique a donnée une meilleure performance sur la base CUHK_Square comparée à celle obtenue sur la base MIT. La même interprétation est vraie pour le détecteur SMC_B_OAS. Toutefois, Wang 2014 possède pratiquement les mêmes performances sur les deux bases. Nous constatons que plus le détecteur générique est performant plus le spécialisé est performant.

Tableau 12 – Comparaison de la performance de détection avec celles des détecteurs de l'état de l'art pour un FPPI=1

Détecteur	Bases	
	CUHK (%)	MIT (%)
Générique [Dalal et Triggs, 2005]	26.60	9.80
Det_manu	50.36	22.01
Nair 2004 [Nair et Clark, 2004]	28.80	42.70
Wang 2014[Wang *et al.*, 2014b]	51.12	**49.00**
Mao 2015 [Mao et Yin, 2015]	61.50	-
Notre SMC_B_OAS	**81.35**	48.97
Gain (SMC_B_OAS / générique)	205.82	399.63

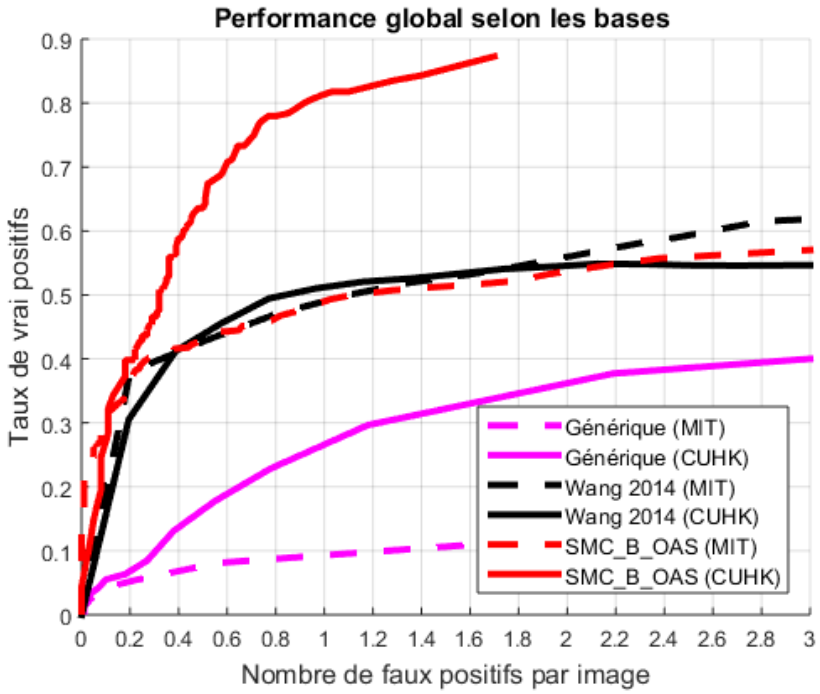


FIGURE 46 – Performance globale de même détecteur par rapport aux bases de données

Les différentes expérimentations que nous avons mené, ont montré que le processus de spécialisation SMC converge après seulement quelques itérations dans quatre cas : deux pour la détection des piétons et deux pour la détection des voitures. Différentes stratégies ont été utilisées dans l'étape de prédiction et l'étape de mise à jour, pour valider l'apport de notre approche et monter son aspect générique.

5 Généricité de la spécialisation avec un détecteur à base d'apprentissage profond

Récemment, les travaux basés sur l'apprentissage profond émergent de manière exponentielle. Ils présentent de hautes performances à la fois dans la classification et la détection. Pourtant, il est connu que ces modèles nécessitent de grandes bases de données et ont beaucoup de paramètres à entrainer. Afin de profiter de ces classifieurs, certains travaux ont proposé de transférer le CNN appris sur un grand ensemble de données sources à un domaine cible avec une base de données de taille réduite.

Parmi ces travaux, nous citons le travail de Grishick *et al.*[Girshick *et al.*, 2013] qui ont traité l'idée d'entraîner un réseau CNN quand les données étiquetées sont insuffisants. Ils ont fait l'apprentissage supervisé d'un réseau pour une tâche source avec des données abandantes (classification d'images). Ensuite, ils ont raffiné le réseau pré-entrainé pour la tâche de détection où il y a des données rares. Oquab *et al.*[Oquab *et al.*, 2014] ont copié les poids d'un classifieur CNN entrainé sur la base ImageNet à un réseau cible avec des couches supplémentaires pour la classification des images de la base Pascal VOC. Dans [Li *et al.*, 2015], Li *et al.* ont proposé d'adapter un détecteur générique ConvNet de voiture à une scène spécifique en réservant des filtres partagés entre les données sources et cibles et en mettant à jour les filtres non partagés. Contrairement à [Oquab *et al.*, 2014, Li *et al.*, 2015] qui nécessitent la présence de certains échantillons annotés dans le domaine cible, Zeng *et al.*[Zeng *et al.*, 2014] ont appris la distribution du domaine cible tout en optant pour l'approche de Wang *et al.*[Wang *et al.*, 2014b] comme une entrée à leur modèle profond et pour pondérer les échantillons des deux domaines sans étiquetage manuel des données de la scène cible.

En suivant le même principe, nous avons proposé, dans cadre d'une collaboration avec Ala Mhalla [3], de spécialiser un détecteur Faster R-CNN avec notre méthode de spécialisation. Faster R-CNN est un travail de Ren *et al.*[Ren *et al.*, 2015] qui a marqué la détection d'objets dans des images avec un taux de précision moyenne égale 70.0% sur la base Pascal VOC 2007. Ce détecteur combine deux réseaux de neurones profond ; le premier RPN sert à proposer des régions susceptible de contenir des objets et le deuxième réseau est le réseau Fast R-CNN [Girshick, 2015] qui se charge de la classification et la mise à jour de la taille des régions proposés par le premier. Le modèle Faster R-CNN utilise à la fois les caractéristiques DCNN pour la proposition des régions et pour la classification d'objets.

Pour ce faire, nous avons adapté l'architecture de notre spécialisation pour prendre en considération les propriétés du détecteur de type Faster R-CNN. Le schéma bloc de la version modifiée est présenté dans FIGURE 47. Et le Tableau 13 résume les principales modifications entre la spécialisation d'un détecteur HOG-SVM et celle d'un détecter Faster R-CNN à une itération k donnée.

3. Doctorant qui travaille dans l'institut Pascal et le laboratoire LATIS

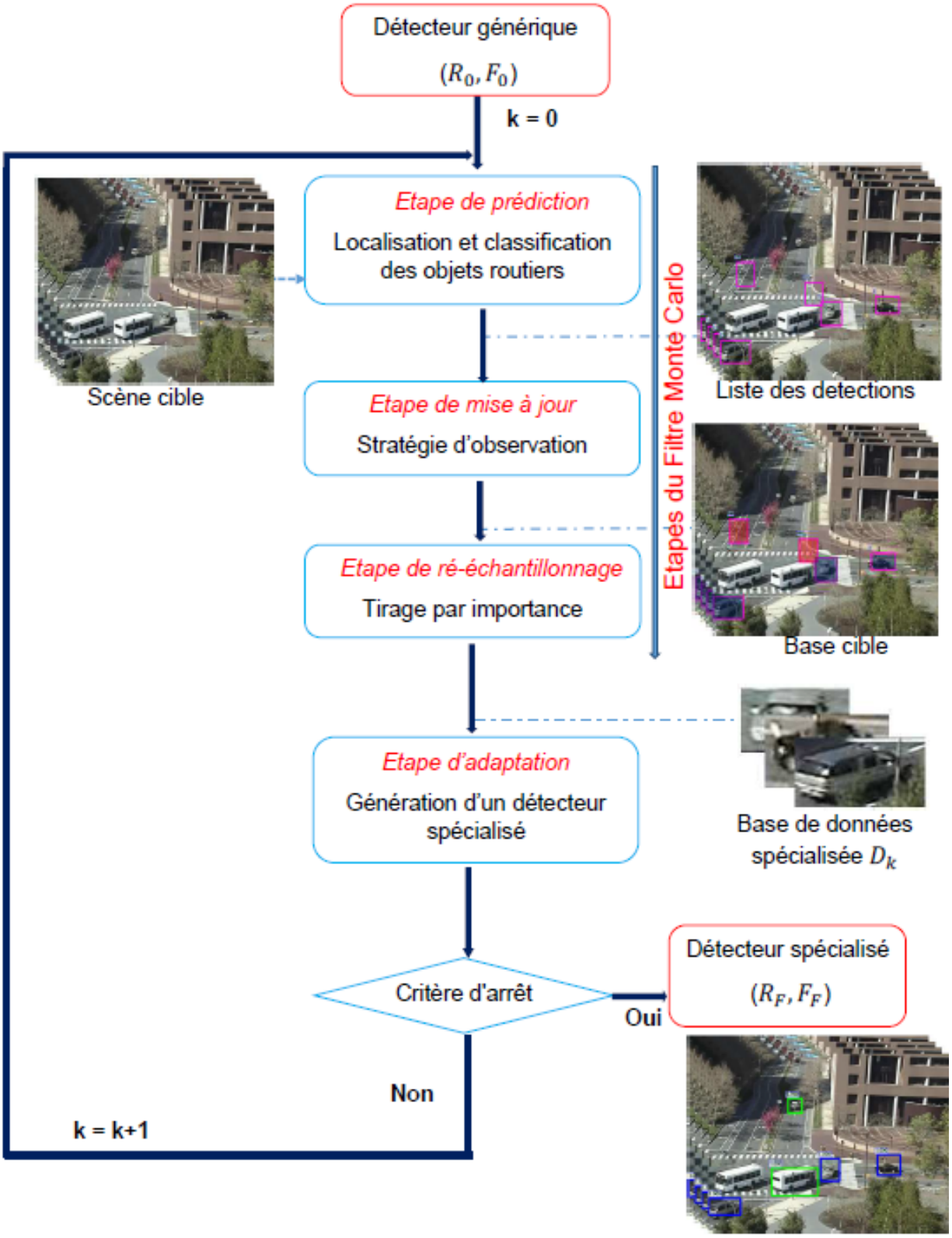


FIGURE 47 – Schéma bloc de la spécialisation d'un détecteur Faster R-CNN

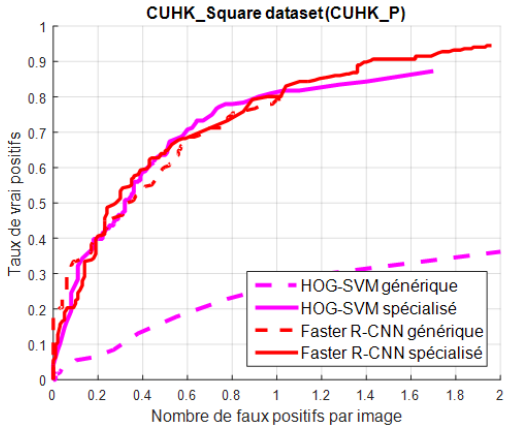
Tableau 13 – Différences entre la spécialisation d'un HOG-SVM et la spécialisation d'un Faster R-CNN

	Spécialisation d'un HOG-SVM	Spécialisation d'un Faster R-CNN
Entrées	- HOG-SVM générique - Base générique - Modèles de fond pré-calculés	- Faster R-CNN générique (architecture et poids)
Étape prédiction	- Apprentissage d'un HOG-SVM sur la base spécialisée précédente - Recherche et proposition d'échantillons cibles (positifs et négatifs)	- Recherche et proposition d'échantillons cibles (positifs)
Étape mise à jour	- Pondération des échantillons cibles en utilisant une stratégie d'observation	
Étape de ré-échantillonnage	- Pondération des échantillons sources - Sélection des échantillons sources et cibles pour créer une base spécialisée	- Sélection des échantillons cibles positifs pour créer une base spécialisée
Étape de fine-tuning (ajustement fin)		Ajustement fin des poids du réseau Faster R-CNN (donc ajustement des poids des deux sous Réseaux RPN et Fast R-CNN)
Données transférées	- Échantillons sources (positifs et négatifs) proches visuellement des cibles	- Architecture et poids du réseau Faster R-CNN
Sorties	- Détecteur HOG-SVM spécialisé - Base spécialisée	- Détecteur Faster R-CNN spécialisé (architecture et poids)

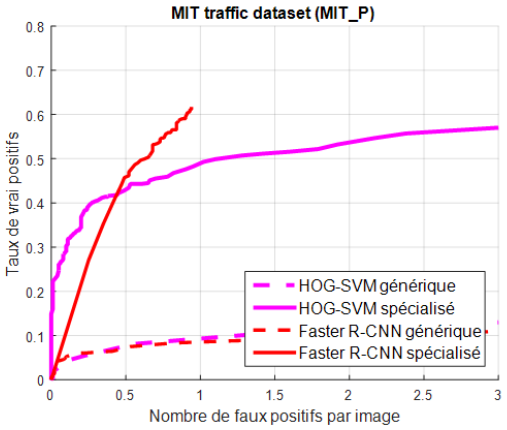
Après avoir donné une idée sur les différences entre la spécialisation d'un HOG-SVM et la spécialisation d'un Faster R-CNN, nous présentons dans la FIGURE 48 les résultats obtenus suite à une expérimentation faite pour montrer la généricité de notre approche de spécialisation et son apport à adapter un détecteur à base d'apprentissage profond. Dans ces expérimentions : (i) Nous avons fixé d'utiliser la stratégie d'indices spatio-temporels OAS comme une stratégie d'observation. (ii) Nous avons spécialisé deux détecteurs Faster R-CNN ; un détecteur de piétons et un détecteur de voitures. (iii) Nous avons fait le test sur deux bases pour chaque détecteur.

Pour la FIGURE 48a ; nous avons enregistré que les détecteurs Faster R-CNN générique et spécialisé ont des performances comparables et qui sont très proches de HOG-SVM spécialisé. Le détecteur Faster R-CNN générique fonctionne bien dès le départ sur la scène CUHK et arrive à détecter la plupart des échantillons positifs dans la scène. Donc, la spécialisation n'a pas introduit d'ajustement des poids du réseau.

Pour le cas de détection de piétons sur la base MIT Traffic (FIGURE 48b), les deux détecteurs génériques HOG-SVM et Faster R-CNN donnent pratiquement le même taux de détection. Et après la spécialisation, le Faster R-CNN spécialisé donne un gain dans le taux de détection supérieur à 50% avec moins d'un seul faux positif par image. Tandis que, le détecteur HOG-SVM spécialisé a enregistré un gain d'environ 40% avec un seul faux positif par image.

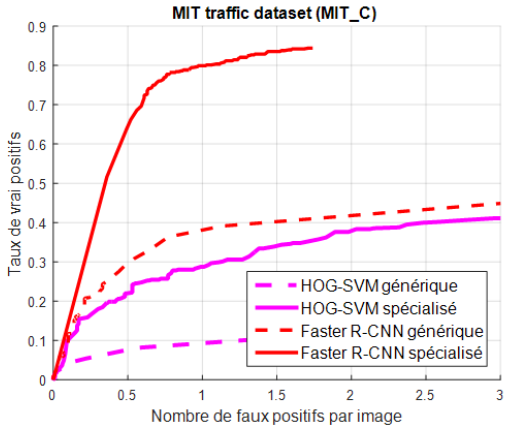


(a) CUHK_Square dataset

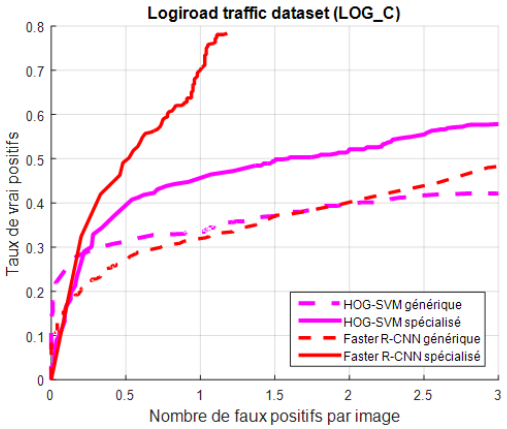


(b) MIT Traffic dataset

(a) & (b) Détection de piétons



(c) MIT Traffic dataset



(d) Logiroad Traffic dataset

(c) & (d) Détection de voitures

FIGURE 48 – Comparaison des performances des détecteurs HOG-SVM et Faster R-CNN

Cependant, les résultats de détection de voitures sur la base MIT Traffic (FIGURE 48c) sont assez différentes. La spécialisation de détecteur HOG-SVM a apporté uniquement 18% par rapport au détecteur générique. Alors que le détecteur Faster R-CNN générique a détecté presque 40% des voitures et sa version spécialisée a détecté 80% des voitures toujours avec un seul faux positif par image.

Concernant le cas de la scène Logiroad Traffic (FIGURE 48d), le HOG-SVM générique et le Faster R-CNN générique donnent pratiquement le même résultat à un FPPI=1. Mais le détecteur Faster R-CNN spécialisé a une performance égal à 70% par rapport 45% de performance qui a été enregistrée par le détecteur HOG-SVM spécialisé.

Cette série d'expérimentations a mis en évidence de nouveau la généricité de notre méthode, non seulement pour utiliser différents stratégies de proposition et d'observation mais aussi pour spécialiser différents classifieurs.

Conclusion

Dans la première section de ce chapitre, le processus complet de notre méthode de spécialisation d'un détecteur générique vers une scène vidéo cible est exposé. Ensuite, une description des différents détecteurs générique HOG-SVM est faite dans section 2.

Dans la troisième section du chapitre, une évaluation rigoureuse est réalisée. Nous avons étudié le choix du paramètre α_t et nous avons validé le critère d'arrêt fixé via l'étude de la convergence du détecteur au fur et à mesure de la spécialisation. Nous avons évalué la performance de l'approche via l'intégration de différentes stratégies de collecte d'échantillons et d'observation. Ainsi nous avons testé sa performance, dans la quatrième section, vis-à-vis autres algorithmes de l'état de l'art. Dans la dernière section, nous avons montré la généricité de notre approche par un test de spécialisation d'un détecteur à base d'apprentissage profond (le Faster R-CNN).

Dans le chapitre suivant, nous allons présenter le logiciel OD SOFT et les étapes d'implémentation et d'intégration de notre travail dans OD SOFT ainsi que les comparaisons effectuées.

Chapitre 6

Implémentation

Introduction

L'entreprise Logiroad est une société d'édition des logiciels d'aide à la décision dans le domaine de trafic routier. Logiroad vend cinq produits qui sont L^2R Mesure, L^2R Base, L^2R Programme, OD Record et OD Soft. Les trois premiers sont destinés principalement à l'entretien des routes et les deux derniers sont destinés à l'exploitation des informations collectées sur le trafic routier. Les produits de Logiroad permettent de fournir : des statistiques précises et rapides sur la densité de la circulation routière, prévention et élimination des cas de congestion, estimation des budgets des entretiens des routes, etc. Les principaux clients de logiroad, nationaux et/ou internationaux, sont les gestionnaires des routes.

Nous nous intéressons particulièrement au logiciel OD SOFT que nous allons présenter dans la première section de ce chapitre. La deuxième section décrira sa configuration et son fonctionnement. La troisième section sera réservée à exposer l'intégration des détecteurs spécialisés dans le logiciel OD soft. Dans la dernière section, nous comparons la méthode Vu-mètre de Logiroad aux deux détecteurs HOG-SVM et Faster R-CNN spécialisés par notre approche.

Les parties de ce chapitre, qui reprennent les bases de logiciel OD SOFT, sont fortement inspirées de site web de Logiroad [1], et de Manuel OD Soft V1.3 [Pitard et Goyat, 2015].

1 Présentation OD SOFT

Le logiciel OD SOFT (Figure 49), est un programme d'analyse et d'étude de trafic routier. Il permet plus précisément de connaitre le flux des véhicules selon leur catégorie par vidéo, de générer automatiquement des rapports de résultats telles que des matrices Origines / Destinations, des comptage, etc.

Le logiciel permet de faire la classification des véhicules en cinq catégories :

— 2 R : C'est la classe des véhicules à 2 roues qui regroupe les instances de motos et vélos.
— VL : C'est la classe des véhicules légers à 4 roues.
— PL : C'est la classe des véhicules de poids lourds.
— Bus : C'est la classe des véhicules de type Bus.
— PLG : C'est la classe qui représente des véhicules de poids lourds avec remorque.

L'utilisateur du logiciel intervient pour spécifier les catégories à classer dans chaque séquence vidéo, pour fixer les valeurs de certaines paramètres et pour déterminer le type des résultats à fournir après le traitement.

1. http ://www.logiroad.fr/

FIGURE 49 – Interface utilisateur du logiciel OD SOFT.

2 Configuration et fonctionnement de OD SOFT

Le processus d'analyse de vidéo effectué par le logiciel est composé principalement de deux étapes :

— Configuration : Cette étape consiste à créer un projet pour définir des paramètres spécifiques à un site, et de sauvegarder les éléments modifiés par l'utilisateur.

— Traitement et exportation du résultat : Cette étape consiste à détecter, suivre les objets et déterminer leurs trajectoires. En plus, il présente le résultat selon le choix fixé par l'utilisateur au cours de l'étape de configuration.

2.1 Configuration

Au cours de cette étape, l'utilisateur fait le paramétrage des éléments à partir de l'interface en plus de la configuration par défaut des paramètres avancés. L'utilisateur peut modifier par la suite les paramètres avancés si il y a besoin.

Tout d'abord, l'utilisateur doit définir l'emplacement de la vidéo à analyser. Cette dernière est l'élément clé du traitement. Celle-ci doit être de la meilleure qualité possible afin d'avoir un traitement optimal. Le logiciel supporte plusieurs résolutions d'image en sachant que plus la résolution sera élevée, plus le traitement sera de meilleure qualité. La résolution préconisée est 640 * 480 pixels (VGA).

Ensuite, il est important de définir une vitesse moyenne (en km/h) relative à chaque site (ou scène) lorsque les véhicules sont détectés. C'est la vitesse à la détection et non pas la vitesse légale du site.

Après, l'utilisateur doit cocher toutes les catégories qu'il souhaite utiliser lors du traitement. Si aucune catégorie n'est sélectionnée, une catégorie "Toutes catégories confondues" sera alors créée automatiquement.

Par défaut, il y a des valeurs des dimensions pour chaque catégorie que l'utilisateur peut les modifier en cas de besoin. La FIGURE 50 présente l'interface OD SOFT de l'onglet "choix des catégories". L'interface donne aussi la main pour ajouter une nouvelle catégorie.

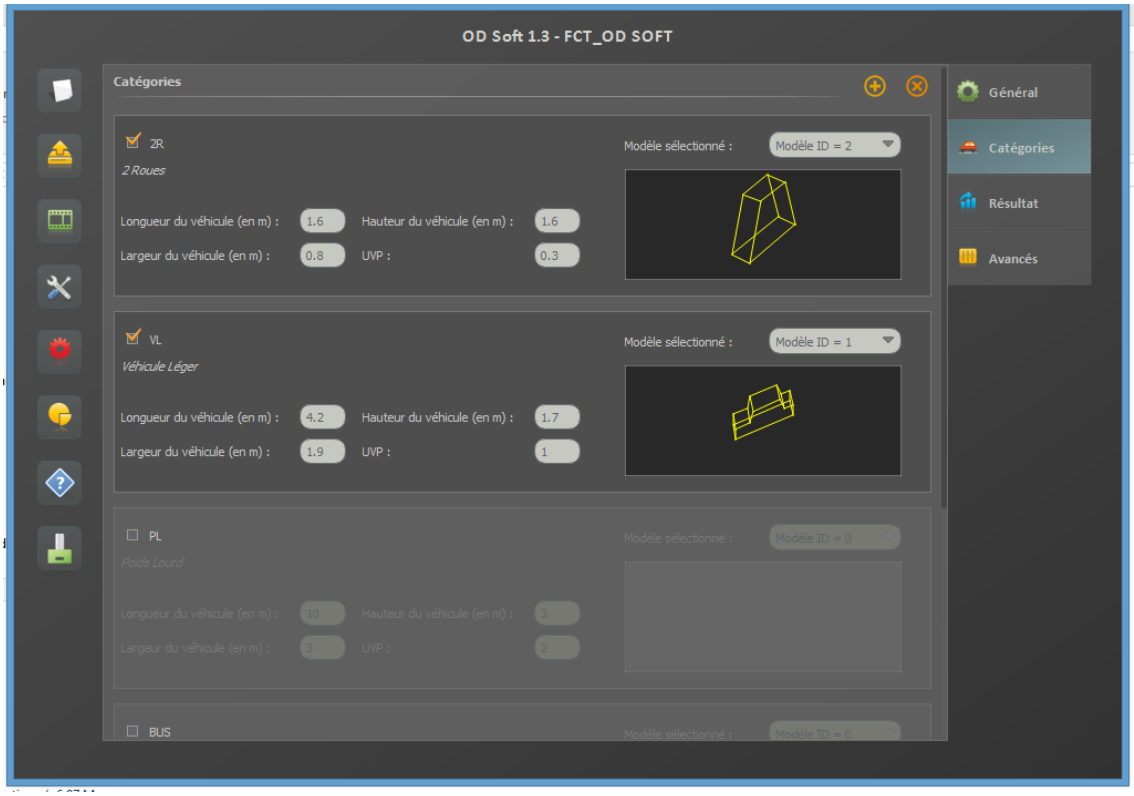


FIGURE 50 – Onglet de configuration des catégories via l'interface de OD SOFT. Configuration des catégories à utiliser dans le traitement. Exemple de projet où l'utilisateur cherche à détecter des objets de type 2R et VL

Il faut également que l'utilisateur spécifie les modèles de sorties qu'il souhaite générer. Par défaut, le logiciel propose de fournir les résultats sous forme de matrices O/D, triées par classe de véhicules et par plage horaire. Cependant, les matrices ne prendront pas en compte la catégorisation s'il n'y a pas de catégorie sélectionnée.

A part les spécifications listées précédemment, l'utilisateur doit définir la zone de masquage, l'emplacement des zones d'entrées et de sorties, de comptage, puis le calibrage de la scène.

La zone de masquage, c'est une zone marquée par l'utilisateur qui ne sera pris en compte lors de traitement (FIGURE 51). Le masquage accélère le temps de calcul et supprime les pixels parasites qui pourraient perturber le bon suivi des véhicules (arbres qui bougent, changement de luminosité dans une zone masquée, etc.)

La zone d'entrée présente la zone où le suivi d'un objet démarre. Elle est représentée par un polygone bleu. En général, il faut placer une zone d'entrée à un endroit où les véhicules ne sont pas à l'arrêt (exp. après le passage des piétons).

La zone de sortie c'est la zone où il faut stopper le suivi d'un véhicule. Elle est représentée par un polygone jaune. De façon générale, on place ce type de zones à la limite de la branche de sortie.

FIGURE 51 – Définition de la zone de masquage. Les zones jaunes sont les zones de masque qui ne seront pas traitées

La définition de ces zones (voir FIGURE 52) permet de :

* Éviter les détections parasites
* Éviter le suivi d'un véhicule sortie de la scène
* Réduire le temps de calcul
* Créer une matrice origine / destination

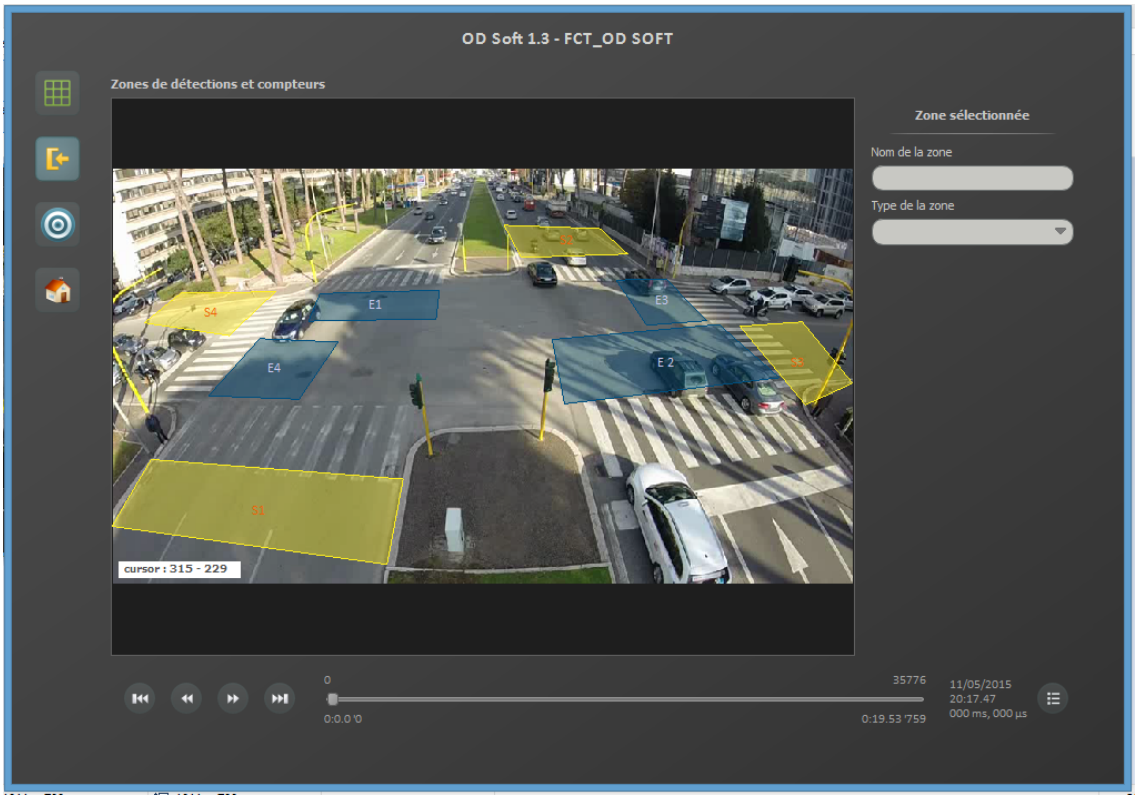


FIGURE 52 – Fixation des zones d'entrées et de sorties. Les zones bleues sont les zones d'entrées et les jaunes sont les zones de sorties

Une **zone de comptage** (ou compteur) est représentée par un polygone vert. Le passage d'un véhicule par cette zone, incrémente le compteur associé. Ce type de zone ne permet pas de faire un suivi complet du véhicule.

Le calibrage d'une scène (FIGURE 53), c'est une autre étape de configuration, permet le traitement des données dans le repère métrique. Il est utile pour unifier les calculs indépendamment de la scène, ainsi pour fournir une catégorisation correcte des véhicules. En effet, les coordonnées métriques permettent de déterminer le gabarit d'un véhicule en fonction de sa taille. Pour ce faire, il faut définir les quatre points au sol de la scène. Puis, il faut définir les points en hauteur correspondant au quatre points définis antérieurement. Ensuite, il faut donner les valeurs métriques de la longueur de la scène, sa largeur et sa hauteur.

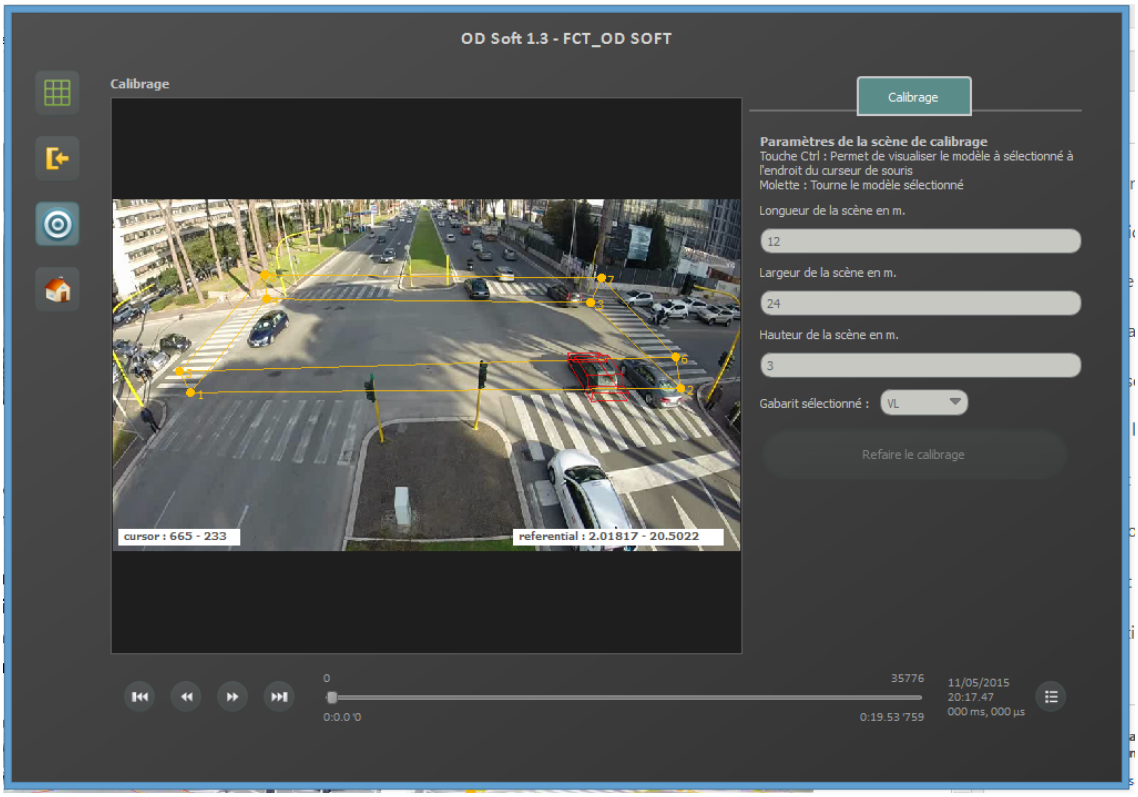


FIGURE 53 – Définition d'une scène de calibrage pour permettre le passage des coordonnées pixels aux coordonnées métriques et inversement

2.2 Traitement et exportation du résultat

Une fois toutes les éléments précédents sont configurés, le traitement automatique de la vidéo peut être lancé. Ce dernier c'est une boucle qui traite la vidéo image par image. Au niveau d'une image les étapes de traitement sont :

1. Extraction des blobs correspondant aux formes. C'est l'étape de détection d'objets à base d'extraction fond-forme.
2. Prédiction des positions possibles des formes précédemment suivies.
3. Association entre les positions prédites et les blobs extraits.
4. Détermination des meilleures associations.
5. Mise à jour des véhicules associés.
6. Gestion des formes non associées.
7. Gestion des véhicules suivis.

La gestion des formes non-associées consiste principalement a créé des nouveaux véhicules pour les formes qui sont détectées dans des zones d'entrées. Par contre, la gestion des véhicules suivis consiste à déterminer les véhicules abandonnés, enregistrer les véhicules associés dans la base de données et gérer les zones de sorties.

En effet, le but de l'analyse est d'extraire des statistiques qui décrivent quantitativement et / ou qualitativement l'état de la circulation routière dans la vidéo étudiée. Par défaut, le résultat retourné a la fin du traitement est une base de données contenant l'ensemble des trajectoires. Ainsi, d'autres types de résultats peuvent être établis en fonction des choix fixés par l'utilisateur.

Il est à signaler que le logiciel donne la possibilité d'exporter les résultats brutes sous formes de tableaux, de matrice O / D, des graphiques,...

3 Intégration des détecteurs spécialisés dans OD SOFT

Notre approche de spécialisation proposée sert à améliorer la première étape du traitement à savoir l'extraction des blobs relatives aux formes. Dans notre travail, nous nous basons sur l'idée que l'amélioration de la détection d'objets induit une amélioration dans les étapes suivantes du traitement.

Cette section présente l'intégration d'un détecteur spécialisé avec notre approche de spécialisation dans le logiciel OD SOFT. Dans 3.1 nous décrivons un cas de détecteur HOG-SVM et dans 3.2 un détecteur de type Faster R-CNN est exposé.

3.1 Détecteur HOG-SVM

Dans la pratique, un détecteur HOG-SVM rendu par la spécialisation est un fichier qui présente l'ensemble des caractéristiques de ce dernier. Le FIGURE 54 présente un exemple de ce fichier. Parmi ces caractéristiques, nous citons principalement 'state', 'windows_size', 'category', et 'coefficients' qui détermine respectivement si le détecteur sera active pour la détection ou non, la taille de fenêtre de balayage pour la recherche de l'objet, la classe de l'objet cherché et les coefficients du modèle HOG-SVM.

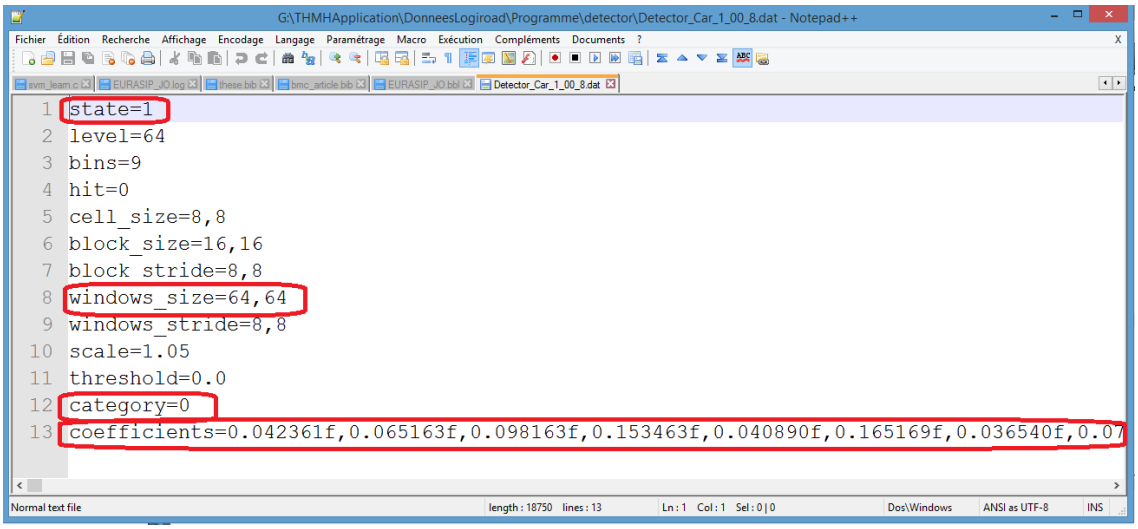


FIGURE 54 – Exemple d'un détecteur HOG-SVM

Pour intégrer ce détecteur dans OD SOFT nous avons ajouté :

— Une option à activer via l'interface de logiciel par l'utilisateur s'il veut utiliser la détection par détecteurs HOG-SVM. Par défaut, cette option est non active.
— Une fonction qui charge la liste des détecteurs qui peuvent être sélectionnés par l'utilisateur.
— Une fonction qui charge les caractéristiques d'un détecteur à partir d'un fichier comme celui présenté dans la FIGURE 54.
— Une fonction qui prend en entrée la liste des détecteurs dont le paramètre "state" est mis à 1 et l'image de la vidéo à traiter. Cette dernière rend en sortie un vecteur de détections où chaque détection est représentée par six paramètres "roi.x", "roi.y", "roi.w", "roi.h", "score" et "category". les paramètres "roi.x", "roi.y" sont les coordonnées pixels du point gauche haut du rectangle englobant l'objet, "roi.w" et "roi.h" sont respectivement la largeur et la hauteur du rectangle. Le score c'est le poids de confiance de la classification calculé par le détecteur et "category" représente la classe de détecteur qui a retourné la détection. Le nombre minimal d'éléments de la liste de détecteurs activés doit être 1 et peut être égal à six au maximum. C'est à dire un détecteur par catégorie d'objet est activé.

La FIGURE 55 présente l'interface OD SOFT suite à l'intégration du détecteur HOG-SVM.

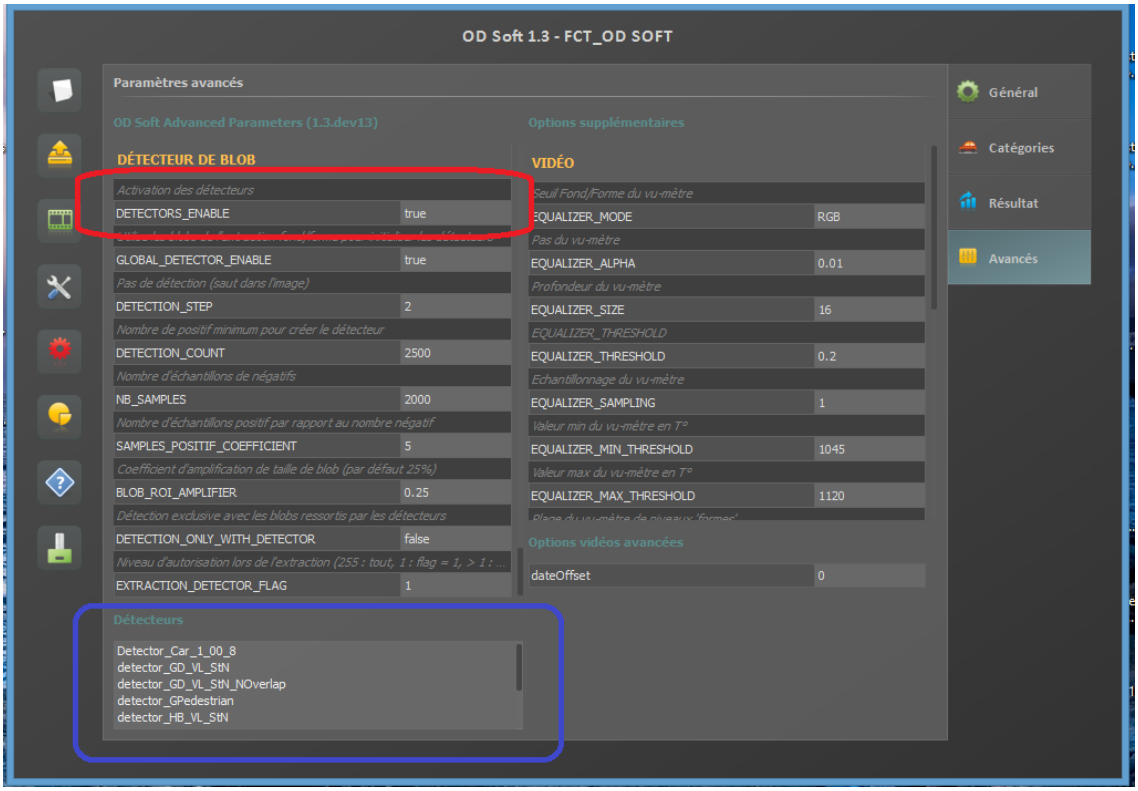


FIGURE 55 – Intégration de HOG-SVM dans OD SOFT : Le rectangle rouge présente l'option d'activation du détecteur HOG-SVM. Le rectangle bleu visualise la liste des détecteurs qui sont chargés par le logiciel et qui peuvent être activés par l'utilisateur

Il est indispensable de signaler que la détection multi-objets dans ce cas est effectuée par l'activation de plusieurs détecteurs mono-objets à la fois. Ces détecteurs mono-objets s'exécutent en parallèle lors de la recherche d'objets dans une image.

3.2 Détecteur Faster R-CNN

Dans le cas des expérimentations illustrées dans la section 5 (page 97), nous avons spécialisé le détecteur Faster R-CNN pour détecter un seul objet à chaque vidéo. Néanmoins, le détecteur Faster R-CNN est initialement un détecteur multi-objets. Donc, il est possible d'être spécialisé avec la même approche de spécialisation vers une scène particulière pour détecter plusieurs objets à la fois.

Pour ce faire, dans la phase de spécialisation d'un détecteur multi-objets, nous donnons un certain taux de confiance à la classification retournée par le Faster R-CNN générique ainsi nous ajoutons d'autres indices visuels qui permettent la distinction entre les objets par les stratégies d'observation tels que la taille du détection, la vitesse de mouvement,...

L'intégration proprement dite de ce type de détecteur n'est pas encore réalisée, mais l'idée consiste à :

— Ajouter une option à activer via l'interface de logiciel par l'utilisateur pour lancer une détection utilisant un détecteur Faster R-CNN.

— Ajouter une fonction qui prend en entrée un fichier d'architecture réseau du détecteur de type Faster R-CNN, un fichier contenant les poids du détecteur spécialisé multi-objets et l'image de la vidéo à traiter. Cette fonction doit rendre en sortie un vecteur de détections qui a les

mêmes éléments et la même structure que celui rendu par la fonction de détection utilisant un détecteur HOG-SVM.

Le détecteur Faster R-CNN spécialisé présente principalement deux avantages qui sont un taux de détection nettement supérieur en lui comparant à un Faster R-CNN générique et une détection multi-objets par le même détecteur en comparant par rapport au détecteur HOG-SVM.

4 Comparaison de détection entre Vu-mètre et nos détecteurs

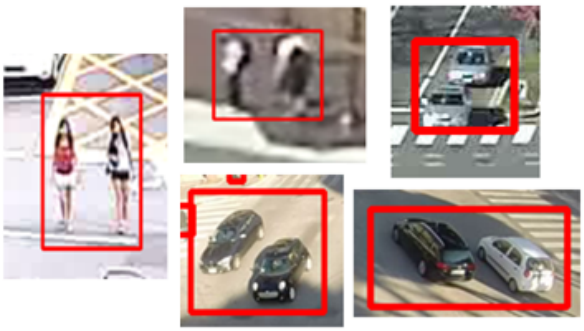
Dans cette section, nous comparons la méthode de détection Vu-mètre (basé sur l'extraction fond-forme) développée initialement par Logiroad, les détecteurs génériques utilisés dans cette thèse et les détecteurs spécialisés obtenus suite à notre spécialisation. Pour ce faire, nous avons fixé quatre critères à évaluer sur les 100 images de chaque base de test étudiée [2]. Ces quatre critères sont :

- Nombre des bonnes détections : Une bonne détection c'est un rectangle qui englobe uniquement une seule instance de l'objet recherché. La figure 56a présente certaines bonnes détections.
- Nombre des détections perdues : C'est le nombre d'instances de l'objet recherché qui ne sont pas détectées.
- Nombre des fausses détections : Une fausse détection est un rectangle dont le contenu n'est pas une instance de l'objet recherché, ou bien la détection est assez large que le rectangle de la vérité terrain. Les figures 56b, 56c, 56d et 56e montrent certains cas de fausses détections.
- Nombre des regroupements : Un regroupement est un cas particulier des fausses détections. C'est une détection qui englobe un ensemble d'instances de l'objet recherché. Nous avons séparés les regroupements en trois cas : Un regroupement de deux instances, un regroupement de trois instances et un regroupement de quatre instances ou plus dans un seul rectangle. Les figures 56c, 56d et 56e présentent des exemples des trois cas de regroupements.

2. Nous étudions quatre cas de détections : Deux cas de détections des piétons sur la base CUHK_Square et la base MIT Traffic et deux cas de détections des voitures sur la base MIT Traffic et la base Logiroad Traffic.

(a) Exemples de bonnes détections

(b) Des mauvaises localisations



(c) Regroupements : 2 instances

(d) Regroupements : 3 instances

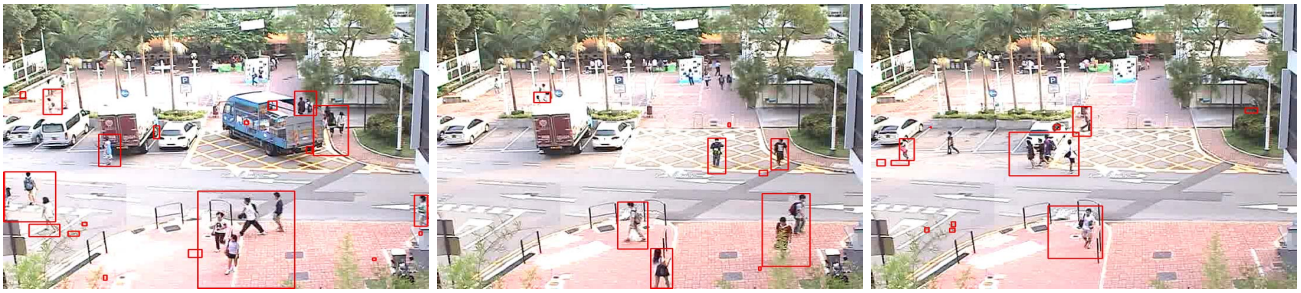
(e) Regroupements : 4 et plus d'instances

Exemples de fausses détections : (b), (c), (d) et (e)

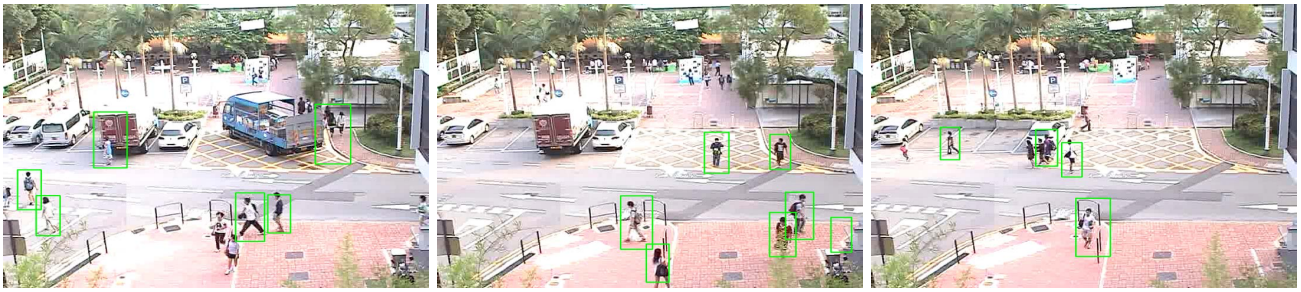
FIGURE 56 – Critères de comparaison entre la Vu-mètre et les détecteurs basés sur l'apprentissage

Dans la suite de cette section, nous donnons respectivement les résultats de comparaisons sous forme d'une illustration et d'un tableau récapitulatif sur la base CUHK_Square, sur la base MIT Traffic (cas des piétons), sur la base MIT Traffic (cas des voitures) et sur la base Logiroad Traffic. L'illustration visualise la détection avec la méthode Vu-mètre, le détecteur HOG-SVM spécialisé et le détecteur Faster R-CNN spécialisé. Le tableau présente une comparaison quantitative entre la méthode Vu-mètre, ainsi les détecteurs génériques HOG-SVM et Faster R-CNN, et les détecteurs spécialisés HOG-SVM et Faster R-CNN.

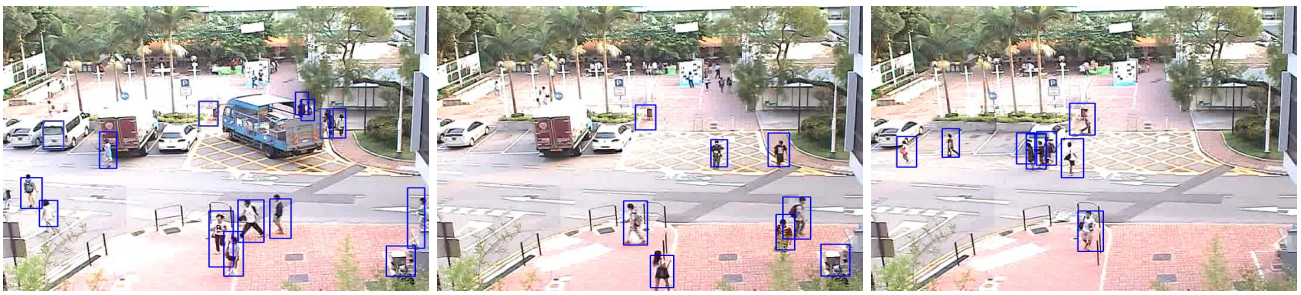
Résultat de comparaison sur la base CUHK_Square



(a) Vu-mètre (méthode initiale de Logiroad)



(b) HOG-SVM spécialisé



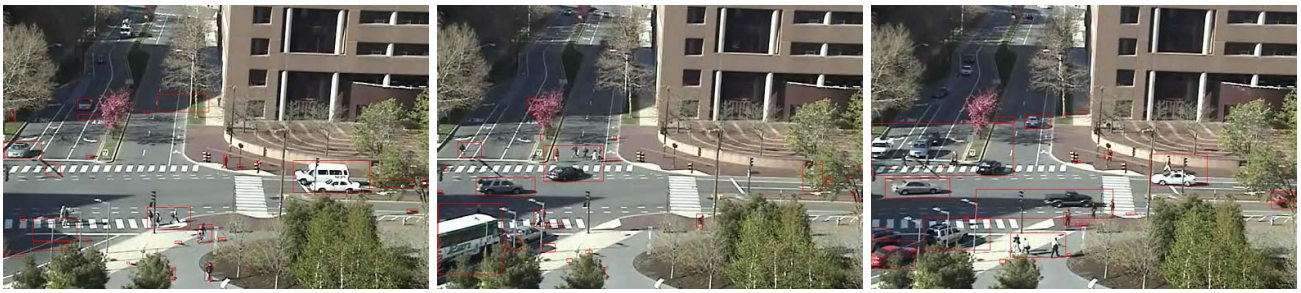
(c) Faster R-CNN spécialisé

FIGURE 57 – Illustration d'exemples détection des piétons sur la base CUHK_Square rendus par Vu-mètre, HOG-SVM spécialisé et par Faster R-CNN spécialisé

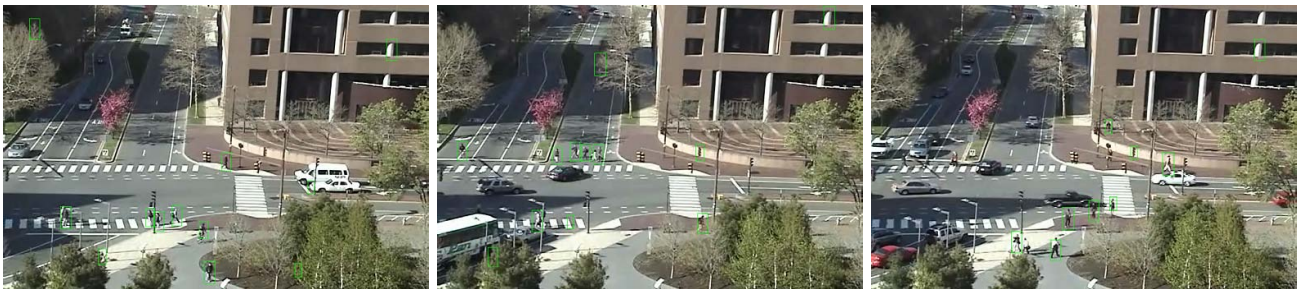
Tableau 14 – Comparaison de performance de détection sur la base CUHK_Square

		Vu-mètre	HOG-SVM		Faster R-CNN	
			Générique	Spécialisé	Générique	Spécialisé
Bonnes Détections		148 (62.7 %)	137 (58.1%)	206 (87.3%)	163 (69.06%)	221 (93.64%)
Détections perdues		88 (37,7 %)	99 (41.9%)	30 (13.7%)	73 (30.94%)	15 (6.36%)
Fausses Détections		956	1177	190	65	143
Regroupements	2 instances	29	10	10	13	17
	3 instances	3	2	2	2	0
	4 instances et plus	7	2	0	1	0

Résultat de comparaison sur la base MIT Traffic (cas des piétons)



(a) Vu-mètre (méthode initiale de Logiroad)



(b) HOG-SVM spécialisé

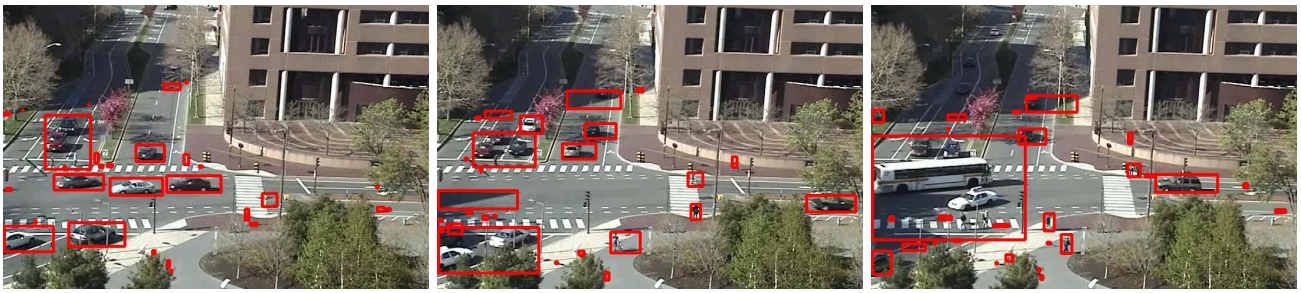
(c) Faster R-CNN spécialisé

FIGURE 58 – Illustration d'exemples de détection des piétons sur la base MIT Traffic rendus par Vu-mètre, HOG-SVM spécialisé et par Faster R-CNN spécialisé

Tableau 15 – Comparaison de performance de détection sur la base MIT Traffic (cas des piétons)

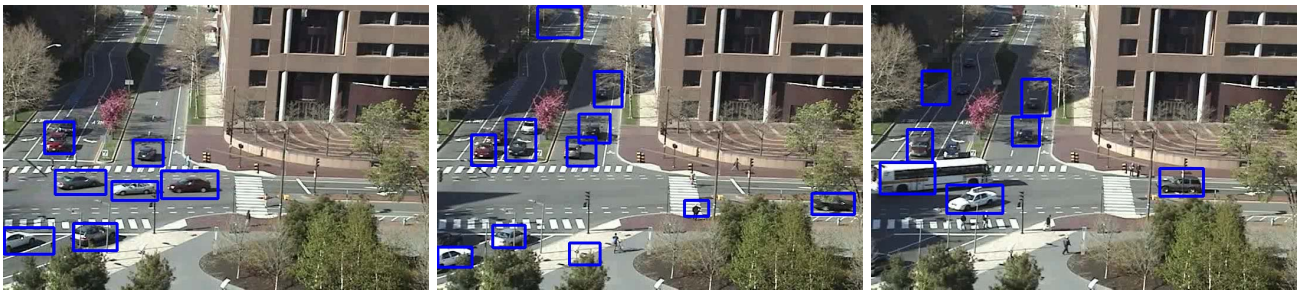
		Vu-mètre	HOG-SVM		Faster R-CNN	
			Générique	Spécialisé	Générique	Spécialisé
Bonnes Détections		195 (40.54 %)	208 (43.2 %)	308 (64,03%)	27 (5.61%)	297 (61.75%)
Détections perdues		286 (59.46%)	273 (56.8 %)	137 (35.97 %)	454 (94,39%)	184 (38.25%)
Fausses Détections		2212	4246	340	25	94
Regroupements	2 instances	29	5	5	4	23
	3 instances	7	1	0	0	1
	4 instances et plus	3	0	0	0	0

Résultat de comparaison sur la base MIT Traffic (cas des voitures)



(a) Vu-mètre (méthode initiale de Logiroad)

(b) HOG-SVM spécialisé



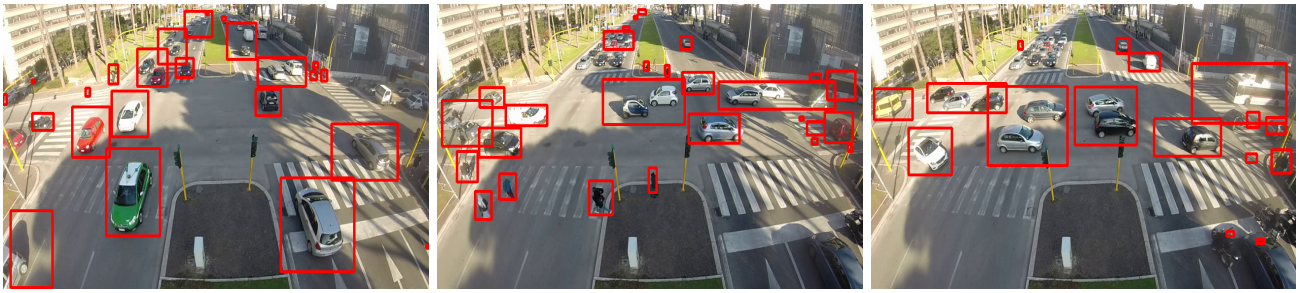
(c) Faster R-CNN spécialisé

FIGURE 59 – Illustration d'exemples de détection des voitures sur la base MIT Traffic rendus par Vu-mètre, HOG-SVM spécialisé et par Faster R-CNN spécialisé

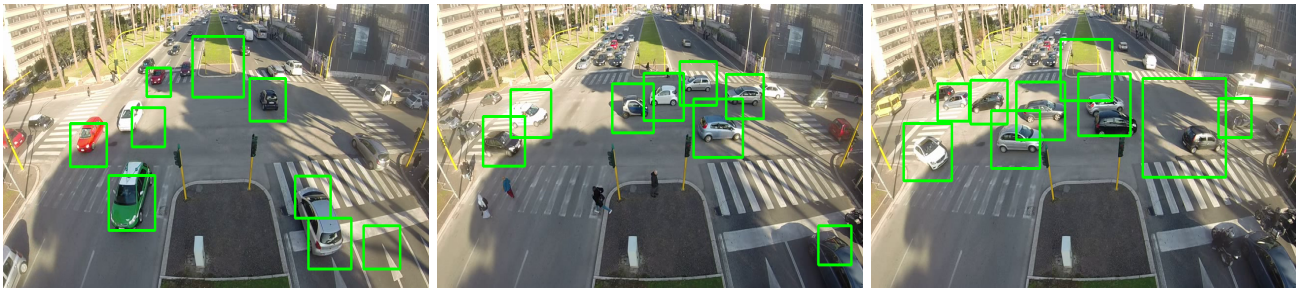
Tableau 16 – Comparaison de performance de détection sur la base MIT Traffic (cas des voitures)

		Vu-mètre	HOG-SVM		Faster R-CNN	
			Générique	Spécialisé	Générique	Spécialisé
Bonnes Détections		253 (59.11%)	62 (14.49%)	173 (40.42%)	230 (53.74%)	362 (84.58%)
Détections perdues		175 (40.89%)	366 (85.51%)	255 (59.58%)	198 (46.26%)	66 (15.42%)
Fausses Détections		2161	75	88	590	174
Regroupements	2 instances	33	0	2	12	25
	3 instances	10	0	0	1	0
	4 instances et plus	17	0	0	0	0

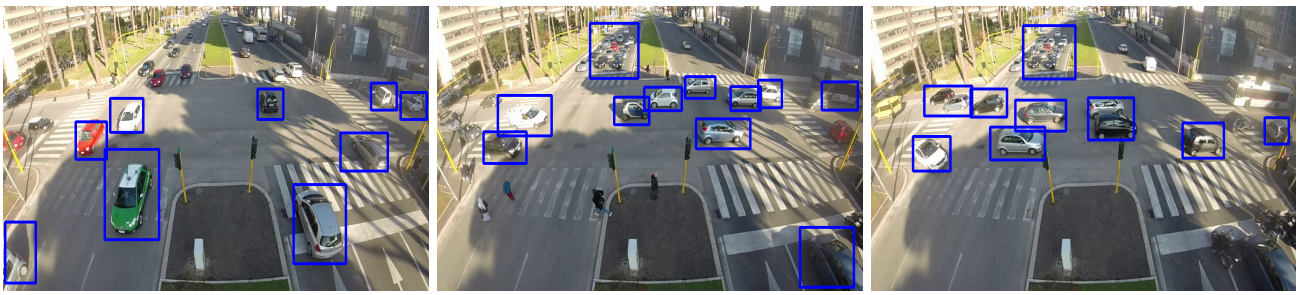
Résultat de comparaison sur la base Logiroad Traffic



(a) Vu-mètre (méthode initiale de Logiroad)



(b) HOG-SVM spécialisé



(c) Faster R-CNN spécialisé

FIGURE 60 – Illustration d'exemples de détection des voitures sur la base Logiroad Traffic rendus par Vu-mètre, HOG-SVM spécialisé et par Faster R-CNN spécialisé

Tableau 17 – Comparaison de performance de détection sur la base Logiroad Traffic

		Vu-mètre	HOG-SVM		Faster R-CNN	
			Générique	Spécialisé	Générique	Spécialisé
Bonnes Détections		323 (49.85%)	433 (66.82%)	445 (68.67%)	339 (52.31%)	508 (78.39%)
Détections perdues		325 (50.15%)	215 (33.17%)	203 (31.33%)	309 (47.69%)	140 (21.61%)
Fausses Détections		1268	1431	680	402	118
Regroupements	2 instances	55	16	11	8	15
	3 instances	23	1	1	6	1
	4 instances et plus	17	0	1	3	0

Les figures 57, 58, 59, 60 et les tableaux 14, 15, 16, 17 montrent encore une fois l'apport de spécialisation en comparant les détecteurs génériques et spécialisés. Ainsi, ils constituent une démonstration de gain de performance ajouté par rapport à la sortie brute de la méthode Vu-mètre de Logiroad. En particulier, ces figures et ces tableaux exposent que la spécialisation des détecteurs a permis :

— L'augmentation du taux des bonnes détections et par la suite, elle a permis la réduction du nombre des détections perdues.
— La diminution du nombre des fausses détections.
— La réduction à la fois du nombre de regroupements et du nombre d'objets regroupés ensemble dans un seul rectangle par rapport au résultat obtenu par la méthode Vu-mètre d'extraction fond-forme de Logiroad.

Conclusion

Nous avons présenté, dans ce chapitre, les détails du logiciel OD SOFT de Logiroad. Dans la première section, nous avons présenté l'objectif de ce logiciel et son interface graphique. La deuxième section a été réservée à la description des différentes configurations nécessaires pour l'utilisation de ce logiciel et à la description de son traitement.

Ensuite, nous avons exposé l'intégration d'un détecteur HOG-SVM dans OD SOFT et le principe d'intégration d'un détecteur de type Faster R-CNN. Dans la dernière section du chapitre, nous avons comparé les détecteurs génériques et spécialisés par rapport à la méthode Vu-mètre développée par Logiroad sur quatre base de données du trafic routier. Nous avons conclu que notre approche ajoute un gain de performance et corrige considérablement le problème des regroupements de la méthode Vu-mètre.

Conclusion générale et perspectives

Les travaux effectués dans cette thèse traitent la problématique de détection multi-objets dans des séquences vidéo en utilisant des détecteurs à base d'apprentissage. L'application étudiée est la détection des objets dans des scènes de circulation routière. Étant donné la chute significative des performances d'un détecteur générique lorsqu'il est appliqué à une scène spécifique, nous proposons une spécialisation automatique d'un détecteur générique vers une scène particulière. La baisse de performance est due principalement aux différences visuelles entre les échantillons d'apprentissage et les échantillons de la scène cible.

La spécialisation développée s'appuie sur une formalisation originale d'un filtre SMC dans un contexte d'apprentissage. Cette formalisation approxime itérativement une distribution cible inconnue comme une base de données d'apprentissage en utilisant une distribution source dans laquelle nous avons un ensemble d'échantillons annotés. Les échantillons de la base d'apprentissage sont considérés comme des réalisations de la distribution de probabilité conjointe entre les vecteurs caractéristiques d'échantillons et les classes d'objets. Ces échantillons sont sélectionnés à la fois à partir de la base source et de la scène cible. Deux stratégies d'observation ont été proposées pour la pondération des échantillons cibles et la détermination des échantillons pertinents sans intervention manuelle. A notre connaissance, il s'agit de la première contribution qui formule le problème d'apprentissage par transfert en utilisant un filtre Monte Carlo. Nous proposons une approche de transfert d'apprentissage générique dans laquelle toute stratégie d'observation peut être intégrée. Ainsi, notre approche peut être appliquée pour spécialiser tout type de classifieur/de détecteur.

Nous avons commencé ce manuscrit par une introduction dans laquelle nous avons particulièrement présenté les deux contextes scientifique et applicatif de nos travaux.

Le premier chapitre a été dédié à dresser un état de l'art sur la détection d'objets, sur l'apprentissage semi-supervisé et sur le transfert d'apprentissage. Dans ce chapitre nous avons présenté une revue de littérature sur les techniques de détection d'objets et sur les différents méthodes d'apprentissage semi-supervisé. Ensuite, nous avons présenté le transfert d'apprentissage naturel et le principe d'apprentissage artificiel ainsi que les avantages de ce dernier et ses différents types. Nous avons détaillé les principales catégories d'approches traitant le transfert d'apprentissage. Nous avons aussi cité certaines applications de détection d'objets à base de transfert d'apprentissage.

Dans le deuxième chapitre, nous avons commencé par définir l'analyse automatique d'une scène de trafic routier. Nous avons décrit par la suite les descripteurs les plus utilisés pour la détection d'objets de vidéo-surveillance routière. Ensuite, nous avons présenté le détecteur d'objet HOG-SVM retenu pour nos travaux tout en détaillant le calcul des primitives HOG et l'algorithme SVM d'apprentissage. De plus, nous avons montré le besoin et les avantages de spécialiser un détecteur générique vers une scène particulière. A la fin du chapitre, nous avons présenté l'outil d'évaluation et les bases de données qui ont été utilisées dans les expérimentations de nos travaux.

Tenant compte de l'application visée, des avantages du transfert d'apprentissage et de la nécessité de spécialisation d'un détecteur, nous avons proposé dans le troisième chapitre, une approche de transfert d'apprentissage à base d'un filtre SMC pour spécialiser un détecteur vers une scène particulière. Nous avons donné dans la première section, un rappel du filtre SMC dans un cadre général. Ensuite,

nous avons décrit le principe du filtre SMC proposé pour le transfert d'apprentissage et nous avons détaillé le processus de chacune de ses étapes.

Dans le quatrième chapitre, nous avons exposé deux stratégies d'observation. Une première stratégie consiste à calculer un overlap_score et un accumulation_score pour valoriser les échantillons proposés par l'étape de prédiction. Pour le même objectif, nous avons proposé également d'utiliser la technique de suivi KLT dans une deuxième stratégie d'observation. Cette dernière classe les points d'intérêts en points mobiles et en points statiques. Puis, selon la nature et le nombre des points d'intérêts détectés dans la *ROI* associée à un échantillon, un poids est affecté permettant la favorisation de l'échantillon avec la vraie étiquette.

Dans le cinquième chapitre, nous avons présenté les principales expérimentations effectuées pour l'évaluation de notre approche. Les expérimentations sont réparties sur cinq catégories. Une première catégorie est faite pour déterminer et valider certains paramètres et critères de notre approche. Une deuxième catégorie concerne l'étude de l'effet de la stratégie de collecte d'échantillons dans l'étape de prédiction. La troisième catégorie d'expérimentations a permis l'évaluation des deux stratégies d'observation et a montré la capacité de la méthode à intégrer de nouvelles stratégies. Dans la quatrième catégorie d'expérimentations nous avons comparé notre approche avec l'état de l'art. Dans la cinquième catégorie, nous avons étudié la généricité de notre approche à spécialiser différents types de classifieurs. Les différents résultats enregistrés montrent que notre approche converge dès les premières itérations de spécialisation. Ainsi, les performances de détecteurs obtenus par notre approche de spécialisation dépassent celles des détecteurs génériques sur différentes scènes de trafic routier et ceci aussi bien de la détection de piétons que de voitures. L'étude expérimentale menée dans nos travaux nous a permis de montrer la généricité de notre approche via l'intégration de différentes stratégies d'observation et via la spécialisation de deux types de détecteurs ; un détecteur HOG-SVM et un détecteur Faster R-CNN basé sur l'apprentissage profond.

Dans le dernier chapitre nous avons présenté le logiciel OD SOFT de Logiroad et son mode de configuration. Nous avons exposé également l'intégration de la sortie de notre approche dans ce logiciel sans être trop dépendant du type de détecteur. Ainsi, nous avons montré le gain en performance ajouté par les détecteurs spécialisés via une comparaison entre ces détecteurs et la méthode Vu-mètre de Logiroad.

Perspectives

D'après les résultats obtenus, la méthode de spécialisation proposée permet d'améliorer les performances de détection à une scène particulière sans intervention humaine et peut être appliquée pour tout type de détecteur. Cependant, plusieurs perspectives peuvent être envisagées.

Comme première perspective, nous proposons d'étudier de manière profonde les paramètres de chaque stratégie d'observation. Au cours de nos travaux, nous avons déterminé empiriquement les valeurs des paramètres de chacune des deux stratégies. Mais, il est possible de régler ces paramètres en utilisant des algorithmes du seuillage adaptatif et des a priori sur la scène. Les deux stratégies proposées se basent sur des indices visuels et des hypothèses simples alors qu'utiliser d'autres indices contextuels complexes et combiner ces derniers avec d'autres informations pré-acquises sur la scène permettent d'améliorer les performances et d'accélérer la convergence de la spécialisation.

Par ailleurs dans notre travail, nous nous sommes intéressés à des scènes cibles qui sont surveillées par des caméras statiques. Cette hypothèse nous a aidé à extraire nos indices contextuels et à justifier les stratégies d'observation utilisées. Comme deuxième perspective, l'étude d'un ensemble d'informations de contexte qui peuvent être extraites avec une caméra mobile mériterait d'être établie pour étendre l'utilisation de notre méthode avec des caméras mobiles.

Dans le chapitre 2, nous avons présenté que l'analyse automatique des vidéos de trafic routier est une chaine d'étapes en série. Or dans nos expérimentations, nous avons étudié uniquement l'apport

de spécialisation sur l'étape de détection. Évidement, améliorer la détection d'objets (première étape dans la chaine) introduit une augmentation dans les performance des étapes suivantes. Cependant, il est important d'étudier l'effet de la spécialisation sur le reste de la chaine d'analyse. Ceci permet d'étendre la spécialisation proposée pour améliorer le suivi des objets dans la scène.

La littérature et nos expérimentations montrent que les performances des classifieurs (ou des détecteurs) de type deep learning dépassent nettement les méthodes les plus populaires dans le domaine de vision par ordinateur. D'autres perspectives de nos travaux liées à ces classifieurs peuvent être étudiées : Une extension de notre approche pour spécialiser un détecteur en ligne, semble être une idée intéressante puisque notre approche de spécialisation est réalisée en mode hors ligne. Également, spécialiser un détecteur à base d'apprentissage profond pour l'importer et l'utiliser directement dans une caméra embarquée est une autre perspective.

Bibliographie

[Agarwal *et al.*, 2004] Agarwal, S., Awan, A. et Roth, D. (2004). Learning to detect objects in images via a sparse, part-based representation. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 26(11):1475–1490. 82, 84

[Agrawala, 1970] Agrawala, A. (1970). Learning with a probabilistic teacher. *Transactions on Information Theory (IT)*, 16(4):373–379. 20

[Alvarez *et al.*, 2009] Alvarez, S., Sotelo, M., Parra, I., Llorca, D. et Gavilán, M. (2009). Vehicle and pedestrian detection in esafety applications. *In World Congress on Engineering and Computer Science (WCECS)*, volume 2, pages 1–6. 18

[Angulo et Marcotegui, 2005] Angulo, J. et Marcotegui, B. (2005). Sur l'influence des conditions d'éclairage dans la segmentation morphologique couleur par LPE. *Actes de CORESA 2005 (Compression et Représentation des Signaux Audiovisuels)*, pages 313–318. XV, 3

[Arbeláez *et al.*, 2014] Arbeláez, P., Pont-Tuset, J., Barron, J. T., Marques, F. et Malik, J. (2014). Multiscale combinatorial grouping. *In Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 328–335. IEEE. 18

[Aytar, 2014] Aytar, Y. (2014). *Transfer learning for object category detection.* Thèse de doctorat, University of Oxford. 24, 25, 31, 32, 40

[Aytar et Zisserman, 2011] Aytar, Y. et Zisserman, A. (2011). Tabula rasa : Model transfer for object category detection. *In International Conference on Computer Vision (ICCV)*, pages 2252–2259. IEEE. 28, 32, 33, 40

[Bai *et al.*, 2006] Bai, H., Wu, J. et Liu, C. (2006). Motion and haar-like features based vehicle detection. *In 12th International Multi-Media Modelling Conference (IMMMC)*, pages 4–pp. IEEE. 38

[Bautista *et al.*, 2016] Bautista, C. M., Dy, C. A., Mañalac, M. I., Orbe, R. A. et Cordel, M. (2016). Convolutional neural network for vehicle detection in low resolution traffic videos. *In Region 10 Symposium (TENSYMP), 2016 IEEE*, pages 277–281. IEEE. 39

[Bay *et al.*, 2006] Bay, H., Tuytelaars, T. et Van Gool, L. (2006). Surf : Speeded up robust features. *In European Conference on Computer Vision (ECCV)*, pages 404–417. Springer. 36, 38

[Black, 1992] Black, M. J. (1992). *Robust incremental optical flow.* Thèse de doctorat, PhD thesis, Yale university. 16

[Black et Anandan, 1996] Black, M. J. et Anandan, P. (1996). The robust estimation of multiple motions : Parametric and piecewise-smooth flow fields. *Computer Vision and Image Understanding (CVIU)*, 63(1):75–104. 16

[Blitzer *et al.*, 2006] Blitzer, J., McDonald, R. et Pereira, F. (2006). Domain adaptation with structural correspondence learning. *In Empirical Methods in Natural Language Processing (EM in NLP)*, pages 120–128. Association for Computational Linguistics. 29, 32

[Blum et Chawla, 2001] Blum, A. et Chawla, S. (2001). Learning from labeled and unlabeled data using graph mincuts. *In Eighteenth International Conference on Machine Learning (ICML)*, pages 19–26. 21

[Blum et Mitchell, 1998] Blum, A. et Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. *In Proceedings of the eleventh annual conference on Computational Learning Theory (CLT)*, pages 92–100. ACM. 20

[Bodor *et al.*, 2003] Bodor, R., Jackson, B. et Papanikolopoulos, N. (2003). Vision-based human tracking and activity recognition. *In 11th Mediterranean Conference on Control and Automation (CA)*, volume 1. Citeseer. 15

[Boltz *et al.*, 2009] Boltz, S., Debreuve, E. et Barlaud, M. (2009). High-dimensional statistical measure for region-of-interest tracking. *Transactions on Image Processing (IP)*, 18(6):1266–1283. 88

[Carbonetto *et al.*, 2008] Carbonetto, P., Dorkó, G., Schmid, C., Kück, H. et De Freitas, N. (2008). Learning to recognize objects with little supervision. *International Journal of Computer Vision (IJCV)*, 77(1-3):219–237. 84

[Chapelle *et al.*, 2006] Chapelle, O., Scholkopf, B. et Zien, A. (2006). *Semi-Supervised Learning*. 1st ed. Cambridge, MA : MIT Press. 19, 21

[Chesnais, 2013] Chesnais, T. (2013). *Contextualisation d'un détecteur de piétons : Application à la surveillance d'espaces publics*. Thèse de doctorat, Université Blaise Pascal - Clermont II. 16, 17, 20, 21, 22, 40, 45

[Chesnais *et al.*, 2012] Chesnais, T., Allezard, N., Dhome, Y. et Chateau, T. (2012). Automatic process to build a contextualized detector. *In International Conference on Computer Vision Theory and Applications (VISAPP), Volume 1*, volume 1, pages 513–520. SciTePress. 20, 21

[Dai *et al.*, 2007] Dai, W., Yang, Q., Xue, G.-R. et Yu, Y. (2007). Boosting for transfer learning. *In International Conference on Machine learning (ICML)*, pages 193–200. ACM. 26, 32

[Dai *et al.*, 2008] Dai, W., Yang, Q., Xue, G.-R. et Yu, Y. (2008). Self-taught clustering. *In International Conference on Machine learning (ICML)*, pages 200–207. ACM. 26

[Dalal et Triggs, 2005] Dalal, N. et Triggs, B. (2005). Histograms of oriented gradients for human detection. *In Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 886–893. IEEE. 18, 36, 38, 40, 41, 46, 50, 84, 94, 97

[Dalal *et al.*, 2006] Dalal, N., Triggs, B. et Schmid, C. (2006). Human detection using oriented histograms of flow and appearance. *In European Conference on Computer Vision (ECCV)*, pages 428–441. Springer. 20

[Danescu *et al.*, 2011] Danescu, R., Oniga, F. et Nedevschi, S. (2011). Modeling and tracking the driving environment with a particle-based occupancy grid. *Transactions on Intelligent Transportation Systems (ITS)*, 12(4):1331–1342. 18

[Daumé III, 2009] Daumé III, H. (2009). Frustratingly easy domain adaptation. *CoRR*, abs/0907.1815. 31

[Daume III et Marcu, 2006] Daume III, H. et Marcu, D. (2006). Domain adaptation for statistical classifiers. *Journal of Artificial Intelligence Research (JAIR)*, 26:101–126. 26

[Dollár *et al.*, 2009] Dollár, P., Wojek, C., Schiele, B. et Perona, P. (2009). Pedestrian detection : A benchmark. *In Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 304–311. IEEE. 82

[Donahue *et al.*, 2013] Donahue, J., Hoffman, J., Rodner, E., Saenko, K. et Darrell, T. (2013). Semi-supervised domain adaptation with instance constraints. *In Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 668–675. IEEE Computer Society. 32

[Doucet *et al.*, 2001] DOUCET, A., DE FREITAS, N. et GORDON, N. (2001). *Sequential M]onte Carlo Methods in Practice*. Springer. 6, 56

[Douze *et al.*, 2011] DOUZE, M., RAMISA, A. et SCHMID, C. (2011). Combining attributes and fisher vectors for efficient image retrieval. *In Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 745–752. IEEE. 30

[Duan *et al.*, 2009] DUAN, L., TSANG, I. W., XU, D. et MAYBANK, S. J. (2009). Domain transfer svm for video concept detection. *In Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1375–1381. IEEE. 27, 40

[Elgammal *et al.*, 2000] ELGAMMAL, A., HARWOOD, D. et DAVIS, L. (2000). Non-parametric model for background subtraction. *In European Conference on computer Vision (ECCV)*, pages 751–767. Springer. 15

[Elkerdawi *et al.*, 2014] ELKERDAWI, S. M., SAYED, R. et ELHELW, M. (2014). *Real-Time Vehicle Detection and Tracking Using Haar-Like Features and Compressive Tracking*, pages 381–390. Springer International Publishing. 38

[Erich Buch *et al.*, 2009] ERICH BUCH, N., ORWELL, J. et VELASTIN, S. A. (2009). 3d extended histogram of oriented gradients (3dhog) for classification of road users in urban scenes. *In British Machine Vision Conference (BMVC)*, pages 1–11. British Machine Vision Association. 38

[Everingham *et al.*, 2010] EVERINGHAM, M., VAN GOOL, L., WILLIAMS, C. K., WINN, J. et ZISSERMAN, A. (2010). The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision (IJCV)*, 88(2):303–338. 49

[Farajidavar *et al.*, 2014] FARAJIDAVAR, N., de CAMPOS, T. et KITTLER, J. (2014). Transductive transfer machine. *In Asian Conference on Computer Vision (ACCV)*, pages 623–639. Springer. 25

[Fei-Fei *et al.*, 2006] FEI-FEI, L., FERGUS, R. et PERONA, P. (2006). One-shot learning of object categories. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 28(4):594–611. 28

[Fink, 2005] FINK, M. (2005). Object classification from a single example utilizing class relevance metrics. *Advances in Neural Information Processing Systems (NIPS)*, 17:449–456. 29

[Gao *et al.*, 2008a] GAO, J., FAN, W., JIANG, J. et HAN, J. (2008a). Knowledge transfer via multiple model local structure mapping. *In ACM International Conference on Knowledge Discovery and Data Mining (ACM SIGKDD)*, pages 283–291. ACM. 28

[Gao *et al.*, 2008b] GAO, T., LIU, Z.-g., GAO, W.-c. et ZHANG, J. (2008b). Moving vehicle tracking based on sift active particle choosing. *In International Conference on Neural Information Processing (ICNIP)*, pages 695–702. Springer. 37

[Gao *et al.*, 2012] GAO, T., STARK, M. et KOLLER, D. (2012). What makes a good detector ?– structured priors for learning from few examples. *In European Conference on Computer Vision (ECCV)*, pages 354–367. Springer. 28, 33

[Girshick, 2015] GIRSHICK, R. (2015). Fast R-CNN. *In International Conference on Computer Vision (ICCV)*, pages 1440–1448. IEEE. 98

[Girshick *et al.*, 2013] GIRSHICK, R. B., DONAHUE, J., DARRELL, T. et MALIK, J. (2013). Rich feature hierarchies for accurate object detection and semantic segmentation. *CoRR*, abs/1311.2524. 39, 98

[Goutte et Gaussier, 2005] GOUTTE, C. et GAUSSIER, E. (2005). A probabilistic interpretation of precision, recall and f-score, with implication for evaluation. *In European Conference on Information Retrieval (ECIR)*, pages 345–359. Springer. 49

[Guan *et al.*, 2016] GUAN, H., XINGANG, W., WENQI, W., HAN, Z. et YUANYUAN, W. (2016). Real-time lane-vehicle detection and tracking system. *In Chinese Control and Decision Conference (CCDC)*, pages 4438–4443. IEEE. 39

[Han *et al.*, 2006] Han, F., Shan, Y., Cekander, R., Sawhney, H. S. et Kumar, R. (2006). A two-stage approach to people and vehicle detection with hog-based svm. *In Performance Metrics for Intelligent Systems (PMIS) Workshop*, pages 133–140. Citeseer. 18, 38

[Han *et al.*, 2008] Han, Z., Ye, Q. et Jiao, J. (2008). Online feature evaluation for object tracking using kalman filter. *In International Conference on Pattern Recognition (ICPR)*, pages 1–4. IEEE. 16

[Haussecker et Fleet, 2001] Haussecker, H. W. et Fleet, D. J. (2001). Computing optical flow with physical models of brightness variation. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 23(6):661–673. 16

[Horn et Schunck, 1981] Horn, B. K. et Schunck, B. G. (1981). Determining optical flow. *Artificial Intelligence (AI)*, 17(1-3):185–203. 15, 16

[Hou *et al.*, 2007] Hou, C., Ai, H. et Lao, S. (2007). Multiview pedestrian detection based on vector boosting. *In Asian Conference on Computer Vision (ACCV)*, pages 210–219. Springer. XV, 3

[Hsieh *et al.*, 2014] Hsieh, J.-W., Chen, L.-C. et Chen, D.-Y. (2014). Symmetrical surf and its applications to vehicle detection and vehicle make and model recognition. *Transactions on Intelligent Transportation Systems (ITS)*, 15(1):6–20. 38

[Huang *et al.*, 2006] Huang, J., Smola, A. J., Gretton, A., Borgwardt, K. M. et Schölkopf, B. (2006). Correcting sample selection bias by unlabeled data. *In Advances in neural information processing systems (NIPS)*, pages 601–608. 27

[Isard et Blake, 1998] Isard, M. et Blake, A. (1998). Condensation — conditional density propagation for visual tracking. *International journal of computer vision (IJCV)*, 29(1):5–28. 55, 56

[Jain *et al.*, 2011] Jain, M., Jégou, H. et Gros, P. (2011). Asymmetric hamming embedding : taking the best of our bits for large scale image search. *In the 19th ACM International Conference on Multimedia (ICM)*, pages 1441–1444. ACM. 26

[Jiang et Zhai, 2007] Jiang, J. et Zhai, C. (2007). Instance weighting for domain adaptation in NLP. *In 45th Annual Meeting of the Association for Computational Linguistics (ACL)*, volume 7, pages 264–271. The Association for Computational Linguistics. 27, 32

[Krizhevsky *et al.*, 2012] Krizhevsky, A., Sutskever, I. et Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *In Advances in Neural Information Processing Systems (NIPS)*, pages 1097–1105. 39

[Kumar et Daume III, 2012] Kumar, A. et Daume III, H. (2012). Learning task grouping and overlap in multi-task learning. *In 29th International Conference on Machine Learning (ICML)*. icml.cc / Omnipress. 32

[Kuzborskij *et al.*, 2013] Kuzborskij, I., Orabona, F. et Caputo, B. (2013). From n to n+ 1 : Multiclass transfer incremental learning. *In Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3358–3365. IEEE Computer Society. 32

[Levin *et al.*, 2003] Levin, A., Viola, P. et Freund, Y. (2003). Unsupervised improvement of visual detectors using cotraining. *In International Conference on Computer Vision (ICCV)*, pages 626–633. IEEE. 20, 21

[Li *et al.*, 2015] Li, X., Ye, M., Fu, M., Xu, P. et Li, T. (2015). Domain adaption of vehicle detector based on convolutional neural networks. *International Journal of Control, Automation and Systems (IJCAS)*, 13(4):1020–1031. 98

[Lim *et al.*, 2011] Lim, J. J., Salakhutdinov, R. et Torralba, A. (2011). Transfer learning by borrowing examples for multiclass object detection. *In Advances in Neural Information Processing Systems (NIPS)*, pages 118–126. 27, 33, 40

[Lin *et al.*, 2012] Lin, B.-F., Chan, Y.-M., Fu, L.-C., Hsiao, P.-Y., Chuang, L.-A., Huang, S.-S. et Lo, M.-F. (2012). Integrating appearance and edge features for sedan vehicle detection in the blind-spot area. *Transactions on Intelligent Transportation Systems (ITS)*, 13(2):737–747. 18

[Liu *et al.*, 2011] Liu, X., Lin, L., Yan, S., Jin, H. et Jiang, W. (2011). Adaptive object tracking by learning hybrid template online. *Transactions on Circuits and Systems for Video Technology (CSVT)*, 21(11):1588–1599. 17

[Lo et Velastin, 2001] Lo, B. et Velastin, S. (2001). Automatic congestion detection system for underground platforms. *In International Symposium on Intelligent Multimedia, Video and Speech Processing (IMVSP)*, pages 158–161. IEEE. 14

[Lowe, 1999] Lowe, D. G. (1999). Object recognition from local scale-invariant features. *In International Conference on Computer vision (ICCV)*, volume 2, pages 1150–1157. IEEE. 36, 37

[Lucas *et al.*, 1981] Lucas, B. D., Kanade, T. *et al.* (1981). An iterative image registration technique with an application to stereo vision. *7th International Joint Conference on Artificial Intelligence (IJCAI)*, 81(1):674–679. 15, 16

[Ma et Grimson, 2005] Ma, X. et Grimson, W. E. L. (2005). Edge-based rich representation for vehicle classification. *In International Conference on Computer Vision (ICCV), Volume 1*, volume 2, pages 1185–1192. IEEE. 37

[Maamatou *et al.*, 2015] Maamatou, H., Chateau, T., Gazzah, S., Goyat, Y. et Essoukri Ben Amara, N. (2015). Transfert d'apprentissage par un filtre séquentiel de Monte Carlo : application à la spécialisation d'un détecteur de piétons. *In Journées francophones des jeunes chercheurs en vision par ordinateur ORASIS*. 8

[Maâmatou *et al.*, 2016a] Maâmatou, H., Chateau, T., Gazzah, S., Goyat, Y. et Essoukri Ben Amara, N. (2016a). Sequential Monte Carlo filter based on multiple strategies for a scene specialization classifier. *EURASIP Journal on Image and Video Processing (EURASIP - JIVP)*, 2016(1):40. 8

[Maâmatou *et al.*, 2016b] Maâmatou, H., Chateau, T., Gazzah, S., Goyat, Y. et Essoukri Ben Amara, N. (2016b). Transductive transfer learning to specialize a generic classifier towards a specific scene. *In International Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2016) - Volume 4 : Computer VISion Theory and APPlications (VISAPP)*, volume 4, pages 411–422. SciTePress. 8

[Mao et Yin, 2015] Mao, Y. et Yin, Z. (2015). Training a scene-specific pedestrian detector using tracklets. *In Winter Conference on Applications of Computer Vision (WACV)*, pages 170–176. IEEE. 95, 97

[Maskell et Gordon, 2001] Maskell, S. et Gordon, N. (2001). A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *In Target Tracking : Algorithms and Applications (Ref. No. 2001/174), IEE (TTAA)*, pages 2–1. IET. 56

[Mei et Ling, 2011] Mei, X. et Ling, H. (2011). Robust visual tracking and vehicle classification via sparse representation. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 33(11):2259–2272. 55

[Mhalla *et al.*, 2017] Mhalla, A., Chateau, T., Maâmatou, H., Gazzah, S. et Essoukri Ben Amara, N. (2017). SMC Faster R-CNN : Toward a scene specialized multi-object detector. *Computer Vision and Image Understanding (CVIU), en cours de révision*. 8

[Mhalla *et al.*, 2016] Mhalla, A., Maâmatou, H., Chateau, T., Gazzah, S. et Essoukri Ben Amara, N. (2016). Faster R-CNN scene specialization with a sequential Monte-Carlo framework. *In International Conference on Digital Image Computing : Techniques and Applications (DICTA)*, pages 1–7. IEEE. 8

[Mihalkova *et al.*, 2007] Mihalkova, L., Huynh, T. et Mooney, R. J. (2007). Mapping and revising markov logic networks for transfer learning. *In Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence (CAI)*, volume 7, pages 608–614. AAAI Press. 31

[MILGRAM, 2007] MILGRAM, J. (2007). *Contribution à l'intégration des machines à vecteurs de support au sein des systèmes de reconnaissance de formes : Application à la lecture automatique de l'écriture manuscrite.* Thèse de doctorat, École de Technologie Supérieure, Université du Québec. 41

[Miller *et al.*, 2015] Miller, N., Thomas, M. A., Eichel, J. A. et Mishra, A. (2015). A hidden markov model for vehicle detection and counting. *In 12th Conference on Computer and Robot Vision (CRV)*, pages 269–276. 38

[Momin et Mujawar, 2015] Momin, B. F. et Mujawar, T. M. (2015). Vehicle detection and attribute based search of vehicles in video surveillance system. *In International Conference on Circuit, Power and Computing Technologies (ICCPCT)*, pages 1–4. IEEE. 38

[Nair et Clark, 2004] Nair, V. et Clark, J. J. (2004). An unsupervised, online learning framework for moving object detection. *In Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages II–317. IEEE. 20, 21, 95, 97

[Negri *et al.*, 2008] Negri, P., Clady, X., Hanif, S. M. et Prevost, L. (2008). A cascade of boosted generative and discriminative classifiers for vehicle detection. *EURASIP Journal on Advances in Signal Processing*, 2008:136. 39

[Oliver *et al.*, 2000] Oliver, N. M., Rosario, B. et Pentland, A. P. (2000). A bayesian computer vision system for modeling human interactions. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 22(8):831–843. 15

[Oquab *et al.*, 2014] Oquab, M., Bottou, L., Laptev, I. et Sivic, J. (2014). Learning and transferring mid-level image representations using convolutional neural networks. *In Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1717–1724. IEEE. 98

[Pan *et al.*, 2008a] Pan, J., Hu, B. et Zhang, J. Q. (2008a). Robust and accurate object tracking under various types of occlusions. *Transactions on Circuits and Systems for Video Technology (CSVT)*, 18(2):223–236. 17

[Pan *et al.*, 2008b] Pan, S. J., Kwok, J. T. et Yang, Q. (2008b). Transfer learning via dimensionality reduction. *In Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence (CAI)*, volume 8, pages 677–682. AAAI Press. 30

[Pan *et al.*, 2011] Pan, S. J., Tsang, I. W., Kwok, J. T. et Yang, Q. (2011). Domain adaptation via transfer component analysis. *Transactions on Neural Networks (NN)*, 22(2):199–210. 30, 32

[Pan et Yang, 2010] Pan, S. J. et Yang, Q. (2010). A survey on transfer learning. *Transactions on Knowledge and Data Engineering (KDE)*, 22(10):1345–1359. 13, 22, 23, 25, 31

[Perkins *et al.*, 1992] Perkins, D. N., Salomon, G. *et al.* (1992). Transfer of learning. *International encyclopedia of education*, 2:6452–6457. 22

[Philip et Updike, 2001] Philip, B. et Updike, P. (2001). Caltech computational vision caltech cars 2001 (rear). 82, 84

[Piccardi, 2004] Piccardi, M. (2004). Background subtraction techniques : a review. *In International Conference on Systems, Man and Cybernetics (SMC)*, volume 4, pages 3099–3104. IEEE. 15

[Pitard et Goyat, 2015] Pitard, O. et Goyat, Y. (2015). *Manuel OD Soft V1.3.* Logiroad. 103

[Prisacariu et Reid, 2009] Prisacariu, V. et Reid, I. (2009). fasthog - a real-time gpu implementation of hog. Rapport technique 2310/09, Department of Engineering Science, Oxford University. 39, 40

[Qian *et al.*, 2013] Qian, Z., Yang, J. et Duan, L. (2013). Multiclass vehicle tracking based on local feature. *In Chinese Intelligent Automation Conference (CIAC)*, pages 137–144. Springer. 37

[Quattoni *et al.*, 2008] Quattoni, A., Collins, M. et Darrell, T. (2008). Transfer learning for image classification with sparse prototype representations. *In Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. IEEE. 29

[Raina *et al.*, 2007] Raina, R., Battle, A., Lee, H., Packer, B. et Ng, A. Y. (2007). Self-taught learning : transfer learning from unlabeled data. *In International conference on Machine learning (ICML)*, pages 759–766. ACM. 25, 29, 32

[Rattani, 2010] Rattani, A. (2010). *Adaptive Biometric System based on Template Update Procedures.* Thèse de doctorat, Dept. Of Electrical and Electronic Engineering University of Cagliari. 20, 21, 22

[Ren *et al.*, 2015] Ren, S., He, K., Girshick, R. et Sun, J. (2015). Faster R-CNN : Towards real-time object detection with region proposal networks. *In Advances in Neural Information Processing Systems (NIPS)*, pages 91–99. 98

[Richardson et Domingos, 2006] Richardson, M. et Domingos, P. (2006). Markov logic networks. *Machine Learning (ML)*, 62(1-2):107–136. 31

[Rosenberg *et al.*, 2005] Rosenberg, C., Hebert, M. et Schneiderman, H. (2005). Semi-supervised self-training of object detection models. *In Seventh Workshop on Applications of Computer Vision (WACV)*. IEEE Press. 20

[Saenko *et al.*, 2010] Saenko, K., Kulis, B., Fritz, M. et Darrell, T. (2010). Adapting visual category models to new domains. *In European conference on computer vision (ECCV)*, pages 213–226. Springer. 29

[Saravanakumar *et al.*, 2011] Saravanakumar, S., Vadivel, A. et Saneem Ahmed, C. (2011). Human object tracking in video sequences. *IJIVP, ICTACT*, 2(1):264–273. 16

[Scudder, 1965] Scudder, H. (1965). Probability of error of some adaptive pattern-recognition machines. *Transactions on Information Theory (IT)*, 11(3):363–371. 19, 20

[Sermanet *et al.*, 2013] Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R. et LeCun, Y. (2013). Overfeat : Integrated recognition, localization and detection using convolutional networks. *CoRR*, abs/1312.6229. 39

[Shah, 2013] Shah, P. M. (2013). Notes de cours : Lecture 5.5-histograms of oriented gradients (pdf). 41

[Shantaiya *et al.*, 2013] Shantaiya, S., Verma, K. et Mehta, K. (2013). A survey on approaches of object detection. *International Journal of Computer Applications (IJCA)*, 65(18). 13

[Shi et Tomasi, 1994] Shi, J. et Tomasi, C. (1994). Good features to track. *In Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 593–600. IEEE. 73

[Shujuan *et al.*, 2015] Shujuan, S., Zhize, X., Xingang, W., Guan, H., Wenqi, W. et De, X. (2015). Real-time vehicle detection using haar-surf mixed features and gentle adaboost classifier. *In The 27th Chinese Control and Decision Conference (CCDC)*, pages 1888–1894. IEEE. 38

[Sivaraman et Trivedi, 2013] Sivaraman, S. et Trivedi, M. M. (2013). Vehicle detection by independent parts for urban driver assistance. *transactions on Intelligent Transportation Systems (ITS)*, 14(4):1597–1608. 18

[Smal *et al.*, 2007] Smal, I., Niessen, W. et Meijering, E. (2007). Advanced particle filtering for multiple object tracking in dynamic fluorescence microscopy images. *In 4th International Symposium on Biomedical Imaging : From Nano to Macro (BIFNM)*, pages 1048–1051. IEEE. 55

[Smith, 2007] Smith, K. C. (2007). *Bayesian methods for visual multi-object tracking with applications to human activity recognition.* Thèse de doctorat, Ecole Polytechnique Fédérale de Lausanne. 56

[Song *et al.*, 2011] Song, Z., Chen, Q., Huang, Z., Hua, Y. et Yan, S. (2011). Contextualizing object detection and classification. *In Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1585–1592. IEEE. 30

[Stark *et al.*, 2009] Stark, M., Goesele, M. et Schiele, B. (2009). A shape-based object class model for knowledge transfer. *In International Conference on Computer Vision (ICCV)*, pages 373–380. IEEE. 28, 29

[Stauffer et Grimson, 1999] Stauffer, C. et Grimson, W. E. L. (1999). Adaptive background mixture models for real-time tracking. *In Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2. IEEE. 15

[Sugiyama *et al.*, 2008] Sugiyama, M., Nakajima, S., Kashima, H., Buenau, P. V. et Kawanabe, M. (2008). Direct importance estimation with model selection and its application to covariate shift adaptation. *In Advances in neural information processing systems (NIPS)*, pages 1433–1440. 28

[Sun *et al.*, 2010] Sun, D., Roth, S. et Black, M. J. (2010). Secrets of optical flow estimation and their principles. *In Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2432–2439. IEEE. 15

[Sun *et al.*, 2008] Sun, D., Roth, S., Lewis, J. et Black, M. J. (2008). Learning optical flow. *In European Conference on Computer Vision (ECCV)*, volume 5304 de *LNCS*, pages 83–97. Springer-Verlag. 16

[Sun et Watada, 2015] Sun, D. et Watada, J. (2015). Detecting pedestrians and vehicles in traffic scene based on boosted hog features and svm. *In 9th International Symposium on Intelligent Signal Processing (WISP)*, pages 1–4. IEEE. 18, 39, 40

[Tang *et al.*, 2012] Tang, K., Ramanathan, V., Fei-Fei, L. et Koller, D. (2012). Shifting weights : Adapting object detectors from image to video. *In Advances in Neural Information Processing Systems (NIPS)*, pages 638–646. 33

[Tomasi et Kanade, 1991] Tomasi, C. et Kanade, T. (1991). *Detection and tracking of point features.* School of Computer Science, Carnegie Mellon Univ. Pittsburgh. 73

[Tommasi, 2013] Tommasi, T. (2013). *Learning to learn by exploiting prior knowledge.* Thèse de doctorat, École Polytechnique Fédérale de Lausanne. 24, 25, 32

[Tommasi et Caputo, 2009] Tommasi, T. et Caputo, B. (2009). The more you know, the less you learn : from knowledge transfer to one-shot learning of object categories. *In British Machine Vision Conference (BMVC)*, pages 1–11. British Machine Vision Association. 32

[Tommasi *et al.*, 2010] Tommasi, T., Orabona, F. et Caputo, B. (2010). Safety in numbers : Learning categories from few examples with multi model knowledge transfer. *In Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3081–3088. IEEE. 32

[Torrey et Shavlik, 2009] Torrey, L. et Shavlik, J. (2009). Transfer learning. *Handbook of Research on Machine Learning Applications and Trends : Algorithms, Methods, and Techniques*, 1:242. 25

[Toth *et al.*, 2000] Toth, D., Aach, T. et Metzler, V. (2000). Illumination-invariant change detection. *In 4th Southwest Symposium on Image Analysis and Interpretation (IAI)*, pages 3–7. IEEE. 16

[Toyama *et al.*, 1999] Toyama, K., Krumm, J., Brumitt, B. et Meyers, B. (1999). Wallflower : Principles and practice of background maintenance. *In International Conference on Computer Vision (ICCV)*, volume 1, pages 255–261. IEEE. 15

[Uijlings *et al.*, 2013] Uijlings, J. R., van de Sande, K. E., Gevers, T. et Smeulders, A. W. (2013). Selective search for object recognition. *International journal of computer vision (IJCV)*, 104(2):154–171. 18

[Viola et Jones, 2001a] Viola, P. et Jones, M. (2001a). Rapid object detection using a boosted cascade of simple features. *In Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages I–511. IEEE. 18, 36, 38

[Viola et Jones, 2001b] Viola, P. et Jones, M. (2001b). Robust real-time object detection. *International Journal of Computer Vision (IJCV)*, 4:51–52. 36, 38

[Wang et Mahadevan, 2008] Wang, C. et Mahadevan, S. (2008). Manifold alignment using procrustes analysis. *In International Conference on Machine Learning (ICML)*, pages 1120–1127. ACM. 29

[Wang *et al.*, 2010] Wang, G., Forsyth, D. et Hoiem, D. (2010). Comparative object similarity for improved recognition with few or no examples. *In Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3525–3532. IEEE. 27

[Wang *et al.*, 2014a] Wang, H., Cai, Y. et Chen, L. (2014a). A vehicle detection algorithm based on deep belief network. *The Scientific World Journal (SWJ)*, 2014. 39

[Wang *et al.*, 2012a] Wang, M., Li, W. et Wang, X. (2012a). Transferring a generic pedestrian detector towards specific scenes. *In Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3274–3281. IEEE. 46, 47, 50, 85

[Wang et Wang, 2011] Wang, M. et Wang, X. (2011). Automatic adaptation of a generic pedestrian detector to a specific traffic scene. *In Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3401–3408. IEEE. 26, 33, 40, 46, 85

[Wang *et al.*, 2012b] Wang, X., Hua, G. et Han, T. X. (2012b). Detection by detections : Non-parametric detector adaptation for a video. *In Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 350–357. IEEE. 33

[Wang *et al.*, 2009] Wang, X., Ma, X. et Grimson, W. E. L. (2009). Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 31(3):539–555. 50

[Wang *et al.*, 2014b] Wang, X., Wang, M. et Li, W. (2014b). Scene-specific pedestrian detection for static video surveillance. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 36(2):361–374. 40, 46, 95, 97, 98

[Wang *et al.*, 2008] Wang, Z., Song, Y. et Zhang, C. (2008). *Transferred Dimensionality Reduction*, pages 550–565. Springer Berlin Heidelberg, Berlin, Heidelberg. 30

[Wu, 2008] Wu, L. (2008). *Multi-view hockey tracking with trajectory smoothing and camera selection.* Thèse de doctorat, University of British Columbia. 14

[Xie *et al.*, 2015] Xie, H., Wu, Q., Chen, B., Chen, Y. et Hong, S. (2015). Vehicle detection in open parks using a convolutional neural network. *In Sixth International Conference on Intelligent Systems Design and Engineering Applications (ISDEA)*, pages 927–930. IEEE. 39

[Xue *et al.*, 2008] Xue, G.-R., Dai, W., Yang, Q. et Yu, Y. (2008). Topic-bridged plsa for cross-domain text classification. *In the 31st annual international ACM SIGIR conference on Research and Development in Information Retrieval (RDIR)*, pages 627–634. ACM. 30

[Yang *et al.*, 2007] Yang, J., Yan, R. et Hauptmann, A. G. (2007). Adapting svm classifiers to data with shifted distributions. *In Seventh International Conference on Data Mining Workshops (ICDMW)*, pages 69–76. IEEE. 28, 40

[Yao *et al.*, 2011] Yao, B., Jiang, X., Khosla, A., Lin, A. L., Guibas, L. et Fei-Fei, L. (2011). Human action recognition by learning bases of action attributes and parts. *In International Conference on Computer Vision (ICCV)*, pages 1331–1338. IEEE. 29, 30

[Yuan *et al.*, 2011] Yuan, Q., Thangali, A., Ablavsky, V. et Sclaroff, S. (2011). Learning a family of detectors via multiplicative kernels. *IEEE transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 33(3):514–530. 18

[Zeng *et al.*, 2014] Zeng, X., Ouyang, W., Wang, M. et Wang, X. (2014). Deep learning of scene-specific classifier for pedestrian detection. *In European Conference on Computer Vision (ECCV)*, pages 472–487. Springer. 98

[Zhang *et al.*, 2008] Zhang, C., Hamid, R. et Zhang, Z. (2008). Taylor expansion based classifier adaptation : Application to person detection. *In Computer Vision and Pattern Recognition, Conference on*, pages 1–8. IEEE. 32

[Zweig et Weinshall, 2007] Zweig, A. et Weinshall, D. (2007). Exploiting object hierarchy : Combining models from different category levels. *In International Conference on Computer Vision (ICCV)*, pages 1–8. IEEE. 28