Thèse de Doctorat

Spécialité : Vision Robotique

présentée à **l'École Doctorale en Sciences Technologie et Santé (ED585)**

de **l'Université de Picardie Jules Verne**

par

PAUL BLONDEL

Détection de personnes à partir d'un banc stéréoscopique multimodal embarqué sur un drone

pour obtenir le grade de Docteur de l'Université de Picardie Jules Verne

Thèse dirigée par
Alex POTELLE, Claude PÉGARD et Rogelio LOZANO

Jury proposé :

Rapporteur(s) :	Mohamed DAOUDI	-	LIFL - CNRS
	Vincent FRÉMONT	-	Heudiasyc - CNRS
Examinateur(s) :	Pascal VASSEUR	-	LITIS - UR
Directeur(s) :	Alex POTELLE	-	MIS - UPJV
	Claude PÉGARD	-	MIS - UPJV
	Rogelio LOZANO	-	Heudiasyc - CNRS

"C’est lorsque nous croyons savoir quelque chose qu’il faut justement réfléchir un peu plus profondément."
Frank Herbert

Résumé : Le drone est un excellent outil pour assister les équipes de secours dans la recherche de personnes disparues ou alors pour la surveillance de zones à risque. Pour l'adapter mieux encore à ces missions nous avons souhaité, dans cette étude, mettre au point un système embarqué capable de détecter la présence de personnes au sol en toute autonomie.

Dans cette thèse, nous présentons différentes approches de détection supervisée permettant de détecter les personnes au sol. Nous avons d'abord cherché à exploiter les informations présentes dans le spectre visible ce qui nous a amené à proposer une première approche de détection. Dans un second temps nous avons combiné les spectres visible et infrarouge pour proposer deux autres approches de détection. Nous avons également proposé une nouvelle approche d'entraînement basée sur l'utilisation conjointe du spectre visible et du spectre infrarouge.

La première approche proposée est en mesure de détecter les personnes quelque soit la distance et quelque soient les angles de roulis et de tangage combinés du système de vision par rapport au sol dans le spectre visible. De plus, l'algorithme de détection ne nécessite pas d'avoir de connaissance a priori de la scène.

Parmi les deux approches de détection combinant le spectre visible et le spectre infrarouge proposées : une est conçue pour les vols à moyenne altitude (entre 10 et 80 m de hauteur) et l'autre est conçue pour les vols à basse altitude (inférieure à 10 m de hauteur) mais où la luminosité peut variée fortement (comme en forêt par exemple). La première approche utilise l'information infrarouge pour réduire l'espace de recherche et ainsi réduire les temps de calcul. La deuxième approche explore l'espace de recherche des solutions de manière optimisée en faisant collaborer détecteurs visible et infrarouge ; la détection est rapide et s'adapte dynamiquement à la réponse des capteurs et des détecteurs associés.

Nous présentons également dans cette thèse une approche d'apprentissage multimodale visible / infrarouge semi-supervisée. Nous avons conçu une chaîne de traitement particulière permettant de renforcer itérativement les détecteurs de personnes dans le spectre visible et dans le spectre infrarouge au vu des résultats de ceux-ci dans leur spectre respectif. L'avantage est que l'on peut générer autant de données d'entraînement que voulu : cela permet de régler le problème du manque de donnée d'entraînement disponible.

Mots clés : Drone, stéréovision, infrarouge, détection de personnes, apprentissage semi-supervisé, apprentissage supervisé, invariance en rotation.

Human detection using a multimodal stereoscopic system embedded on a UAV

Abstract : UAVs are perfect tools to assist rescue teams in the search for lost people or to watch dangerous areas to prevent incidents. For this purpose, they have to be capable of detecting people from the air.

In this thesis, we present several supervised detection approaches permitting to detect people from the air. We first tried to use the information available from the visible spectrum and it leads us to a first approach. In a second time, we combined the visible and the far-infrared spectrums and we proposed two other supervised detection approaches. We also proposed a new training approach which conjointly uses the visible and the infrared spectrums.

The first proposed approach can detect people regardless of the distance and regardless of the roll and the pitch angles of the acquisition system combined. Moreover, the detection algorithm does not require any a priori knowledge of the scene.

Amongst the two proposed detection approaches combining the visible and the infrared spectrums : one is designed for average altitude flights (between 10 and 80 m high) and the other is designed for low altitude flights (inferior to 10 m high) but where the luminosity is likely to change quite rapidly (such as in forest, for instance). The first approach uses the infrared information to reduce the search space and thus reduce the computation time. The second approach explores the search space in an optimized way by making collaborate visible and infrared detectors ; the detection is fast and adapt itself dynamically to one malfunctioning sensor or detector.

A semi-supervised visible / infrared multimodal learning approach is also presented is this thesis. We designed a specific processing pipeline to iteratively reinforce the detectors based on infrared and visible people detections. The main advantage of this is that we can generate as much training data as we want : this permits to fix the problem of the lack of available infrared data.

Keywords : UAV, stereovision, infrared, human detection, semi-supervised learning, supervised learning, rotation invariance.

Remerciements

Table des matières

Glossaire

ACF	*Aggregate Channel Features*, Caractéristiques de Canaux Agrégés.
ADAS	*Advanced Driver Assistance Systems*, Système d'aide à la conduite.
AerialTest1	*AerialTest1*, Base de données de test contenant des images aériennes.
AerialTest2	*AerialTest2*, Base de données de test contenant des images aériennes infrarouge et visible.
ATI	*Alpha Training Infrared sataset*, Base de données d'entraînement alpha contenant des images infrarouge.
ATV	*Alpha Training Visible Dataset*, Base de données d'entraînement alpha contenant des images visible.
AVIS	*Visible Infrarouge Synchronized Dataset*, Base de données de test contenant des images visible et infrarouge synchronisées.
CAO	*Centered and Anti-Overflow objectness*, Mesure du caractère objet centrée et sans débordement.
CBT	*Cluster Boosting Tree*, Groupes de boosting d'arbre.
CNN	*Convolutional Neural Network*, Réseau de neurones convolutionnels.
CTAVIS-1	*CoTraining Alpha Visible Infrarouge Synchronized dataset*, Base de données alpha contenant des images visible et infrarouge synchronisés pour le co-entraînement.
CTAVIS-2	*CoTraining Alpha Visible Infrarouge Synchronized Dataset*, Base de données alpha contenant des images visible et infrarouge synchronisés pour le co-entraînement numéro 2.
CTAVIS-3	*CoTraining Alpha Visible Infrarouge Synchronized Dataset*, Base de données alpha contenant des images visible et infrarouge synchronisés pour le co-entraînement numéro 3.

EB	*Edge-Boxes score*, Mesure "Edge-Boxes".
FPPI	*Faux-Positifs Par Image*, Taux de faux-positifs par images.
GMVRT1	*Generalized Multi-View Realisitic Training dataset 1*, Base de données d'entraînement contenant des images réalistes multi-élévations.
GMVRT2	*Generalized Multi-View Realistic Training dataset 2*, Base de données d'entraînement contenant des images réalistes multi-élévations très diverse.
GMVST	*Generalized Multi-View Synthetic Training*, Base de données pour l'entraînement multi-vues en image de synthèse.
GMVST2	*Generalized Multi-View Synthetic Training 2*, Base de données pour l'entraînement multi-vues en image de synthèse.
GPU	*Graphic Processor Unit*, Unité de calcul graphique.
HOG	*Histogram of Oriented Gradients*, Histogramme de gradients orientés.
HOPE	*Histogram of Oriented Phase Energy*, Histogramme d'orientation des énergies de phase.
HSC	*Histogram of Sparse Code*, Histogramme de Code Éparse.
ICF	*Integral Channel Features*, Caractéristiques de Canaux Intégraux.
INRIA	*Institut National de Recherche en Informatique et en Automatique*, Institut de recherche en informatique.
IR-ACF	*Infrared Aggregate Channel Features*, Caractéristiques de canaux agrégés pour l'infrarouge.
K-SVD	*K-Singular Value Decomposition*, k-Décomposition en Valeur Singulière.
LSVM	*Latent Support Vector Machine*, Séparateur à Vaste Marge Latent.

M2D	*Multiple Modalites Detection*, Approche de détection multi-modalités.
MIL	*Multiple Instance Learning*, Instance d'apprentissage multiple.
MOPSO	*Multiple Objective Particle Swarm Optimization*, Essaim de particules multi-objectifs.
NMS	*Non-Maximal Suppression*, Suppression des non-maximums.
PASCAL	*Pattern Analysis, Statistical Modelling and Computational Learning.*, Analyse de formes, modélisation statistiques et apprentissage.
PD	*Pitch Detector*, Détecteur de personnes entraîné pour l'angle de tangage.
PRD	*Pitch and Roll-trained Detector*, Détecteur de personnes entraîné pour les angles de roulis et de tangage.
PSO	*Particule Swarm Optimization*, Essaim de particules.
RD	*Roll Detector*, Détecteur de personnes entraîné pour l'angle de roulis.
ROI	*Region Of Interest*, Région d'intérêt.
SEARCH	*Système d'Exploration Aérien pour la Recherche et la Cartographie en milieu Hostiles*, Recherche de personnes disparues. Projet financé par la région de Picardie.
SIFT	*Scale Invariant Transform Feature*, Caractéristique invariante aux transformations d'échelles.
SNPSO	*Sequential Nitching Particule Swarm Optimization*, Essaim de particules avec "niching" séquentiel.
SVM	*Support Vector Machine*, Séparateur à Vaste Marge.
SyntheticAerialTest1	*SyntheticAerialTest1*, Base de données de test contenant des images aériennes de synthèse.

UTC	*Université Technologique de Compiègne*, École d'Ingénieur généraliste située à Compiègne, Picardie.

Liste des figures

Liste des tableaux

Introduction

Contexte

Motivation

L'évolution rapide des sciences et des technologies a permis à l'homme de réaliser un de ses plus grands rêves : celui de se déplacer dans les airs et ainsi accéder à la liberté de mouvement des oiseaux. L'émergence de l'aviation a permis de raccourcir les temps de déplacement entre villes et d'envisager un grand nombre d'applications, militaires, pendant la Grande Guerre au début du 20éme siècle, et civiles, avec le transport de courriers, de marchandises ou de personnes. Plus de cent ans après les débuts de l'aviation, la robotisation des moyens aériens ouvre de nouvelles perspectives d'utilisations. Ce qui aura un impact important sur la société. Le "robot volant" ou drone a en effet certains avantages que l'avion et l'hélicoptère n'ont pas, notamment : sa taille qui peut être très réduite (car il n'y a pas de pilote), il peut être contrôlé à distance ou être totalement autonome, il peut être utilisé dans des environnements très hostiles et il peut potentiellement manutentionner des objets dans l'espace plus aisément qu'un robot classique grâce à sa liberté de mouvements. Un drone de taille réduite peut être utilisé en société pour assister l'homme dans de nombreuses tâches. D'un point de vue robotique : le drone supplante le robot mobile terrestre par sa mobilité et sa capacité à explorer l'espace environnant.

L'équipement du drone devient de plus en plus accessible et de moins en moins énergivore et lourd ; la miniaturisation ainsi que la qualité d'image des caméras ne cessent de s'améliorer ; le prix des caméras thermiques devient plus abordable pour les petites entreprises et les laboratoires ; les capacités de calculs s'accroissent d'années en années, suivant la loi de Moore, accroissant ainsi le spectre des possibilités et permettant une meilleure autonomisation des drones. De nombreuses avancées dans le contrôle-commande des drones permettent : une stabilisation du vol des drones [Santos 2013], le déplacement d'une flottille de drones sans collision [Saif 2014], l'utilisation d'un bras robotique sur un drone [Caccavale 2014], etc. Parallèlement à cela, de nouveaux types de propulsion sont étudiés tels que les drones hybrides avion-hélicoptère permettant de jouir des avantages du vol stationnaire ainsi que du vol avion [Rudakevych 2007].

Les récentes avancées de cette dernière décennie permettent d'envisager une utilisation des drones pour un grand nombre d'applications potentielles, et dans des

domaines très divers : pour la livraison rapide de petits colis en ville, pour la surveillance de bâtiments ou de manifestations, pour l'étude thermique des bâtiments, pour la mise en place de relais de communications, pour la traque de personnes hostiles, pour la construction de bâtiments, pour la recherche de personnes disparues, etc. Les obstacles technologiques qui séparent les professionnels du drone de ces applications sont en passe d'être éliminés grâce à l'intensification des recherches. Le drone assistera l'homme au quotidien pour l'aider à accomplir des tâches complexes et/ou dangereuses.

Dans le cadre de cette thèse nous nous intéressons à l'utilisation des drones équipés de système de vision pour la recherche automatique de personnes disparues. Et plus particulièrement au développement d'algorithmes de détection automatique de personnes à partir de drones. Cette thèse s'est déroulée dans le cadre du projet régional SEARCH [1] (Système d'Exploration Aérien pour la Recherche et la Cartographie en Milieux Hostiles), financé par la région Picardie.

Projet SEARCH

La recherche de personnes disparues est une tache fastidieuse qui requiert un effort humain, et parfois aussi financier, très important. Dans la même thématique la surveillance des zones hostiles pour la prévention d'accidents est aussi une tâche fastidieuse ; la surveillance doit être continuelle dans le temps et aussi exhaustive que possible. Bien que ces taches ont été considérablement facilitées et améliorées par l'utilisation de moyens aériens et l'émergence sur le marché de caméras thermiques toujours plus performantes et toujours plus abordables, il reste malgré tout une marge importante de progression pour rendre ces opérations plus efficaces, moins fastidieuses et moins onéreuses.

Dans la région Picardie, la sécurité civile utilise des hélicoptères pour trouver et rapatrier les éventuelles personnes égarées dans les baies. En effet, il est recensé chaque année un nombre important de noyades en Baie de Somme. En cause : la montée rapide de la marée qui rend subitement inaccessible le rivage aux personnes se trouvant dans la baie. Le fonctionnement actuel de la surveillance a été jugé coûteux et pas assez efficace par la région. Toujours en Picardie, la topographie ainsi que la densité des forêts peuvent provoquer la perte de repère des promeneurs. Chaque année les services de secours sont sollicités pour rechercher des enfants ou même des adultes égarés.

Le projet SEARCH est un projet financé par la région Picardie. SEARCH signifie : Système d'Exploration Aérien pour la Recherche et la Cartographie en milieux Hostiles. Le but de ce projet est de développer une solution pour la recherche de

1. *Système d'Exploration Aérien pour la Recherche et la Cartographie en milieu Hostiles*, Recherche de personnes disparues. Projet financé par la région de Picardie

personnes en sites ouverts ou d'accès plus difficile. L'idée est de proposer une solution peu onéreuse, rapide à déployer et couvrant la zone de recherche rapidement grâce à l'utilisation d'une flottille de drones semi-automatisés. Les opérateurs pourraient indiquer à l'aide d'une station au sol la zone de recherche géographique à explorer, la flottille décollerait et s'organiserait de manière à balayer rapidement la zone de recherche avec un maximum d'efficacité et chaque drone enverrait à la station au sol les éventuelles détections automatiques de personnes. Pour chaque détection, les coordonnées GPS associées à une vue aérienne issue du drone seraient renvoyées afin que les autorités puissent dépêcher sur place des moyens lourds si le caractère d'urgence est avéré.

Deux équipes de recherche collaborent à ce projet : une équipe spécialisée en vision pour la robotique (du laboratoire MIS) et une équipe spécialisée en contrôle-commande de véhicules aériens (du laboratoire HEUDIASYC). Le laboratoire MIS est rattaché à l'Université Picardie Jules Verne d'Amiens (UPJV) et le laboratoire HEUDIASYC est rattaché à l'Université Technologique de Compiègne (UTC [2]).

Objectifs

Les travaux présentés dans cette thèse concernent la partie vision du projet SEARCH et plus particulièrement la détection automatique de personnes à partir de drones. Tel qu'il est décrit dans le projet, chaque drone de la flottille doit être en mesure de détecter des personnes. Dans le cas du projet SEARCH, la détection automatique de personnes à partir de drones est un challenge à de multiples niveaux ; en effet, plusieurs contraintes doivent être respectées pour résoudre ce challenge, notamment :

(a) La détection doit être possible à partir d'une plate-forme en mouvement dans tout l'espace, le système doit donc être réactif et robuste aux changements d'orientation du système de vision..

(b) Le système doit être aussi capable de détecter les personnes de l'environnemt qui ne sont pas en mouvement.

(c) Les lieux parcourus par le drone ne sont pas connus à l'avance.

(d) La détection doit être robuste aux variations de luminosité en extérieur (allant du plein ensoleillement à la nuit totale).

(e) Il doit être possible de détecter aussi bien des personnes éloignées que proches.

2. *Université Technologique de Compiègne*, École d'Ingénieur généraliste située à Compiègne, Picardie

Les objectifs de cette thèse sont 1) d'étudier les techniques existantes de détections de personnes dans le contexte drone, 2) de proposer des solutions aux contraintes en utilisant seulement la modalité visible et 3) de proposer des solutions en utilisant deux modalités différentes.

Organisation du document

Ce document est organisé de la manière suivante :

Introduction

Dans cette partie, les motivations de cette thèse ainsi que le projet de recherche SEARCH dans lequel s'inscrit cette thèse sont décrits. Les contraintes fondamentales de la détection de personnes à partir de drones, dans le contexte du projet, sont présentées.

- Chapitre 1. Détection de piétons dans le spectre visible

Un état de l'art des techniques de détection de personnes en vue piéton utilisant la modalité visible est présenté. Les différentes étapes de la détection supervisée de personnes en vue piéton sont décrites : le calcul de caractéristiques visuelles, l'apprentissage de motifs humains en se basant sur les caractéristiques visuelles et la recherche de motifs appris. Plusieurs approches récentes de calcul de caractéristiques et d'apprentissage basées sur des modèles monolithiques sont donnés à titre d'exemples. Les approches récentes d'apprentissage de motifs multi-parties sont également présentées.

- Chapitre 2. Détection de personnes en vue aérienne

Un état de l'art des techniques de détection de personnes en vue aérienne "drone" est présenté. Les limites des approches piétons et aériennes décrites précédemment sont étudiées et confrontées aux contraintes du projet SEARCH. Dans ce chapitre nous proposons, dans un premier temps, une adaptation simple des détecteurs de piétons pour permettre une détection de personnes au sol à partir d'un drone qui est alternative aux approches existantes. Nous proposons également des optimisations de fonctionnement pour réduire les temps de calcul à la détection. Dans un second temps, nous proposons un détecteur de personnes au sol qui permet une détection robuste pour un plus grand nombre d'utilisations.

- Chapitre 3. Détection de personnes utilisant deux modalités

Les propriétés de la modalité infrarouge sont étudiées. Les avantages et inconvénients du spectre visible et du spectre infrarouge sont discutés, et l'analyse simultanée des deux modalités est étudiée. Dans ce chapitre nous parlons également des différentes solutions matérielles existantes pour superposer la modalité visible et la modalité infrarouge, nous exposons les avantages et

les inconvénients de chaque approche et nous présentons notre choix matériel. Un état de l'art des approches de détection de piétons dans l'infrarouge seul et différentes approches collaboratives de détections est également présenté. Les limites des approches précédentes sont étudiées et confrontées aux besoins du projet. Une méthode permettant l'apprentissage conjoint d'un détecteur de personnes dans l'infrarouge et d'un détecteur de personnes dans le visible est décrite. Notre détection collaborative, qui est à la fois adaptative et légère, est enfin proposée. Deux déclinaisons de cette approche sont présentées en vue piéton, pour un scénario recherche de personnes en forêts, et en vue aérienne, pour un scénario de surveillance en zone dégagée.

Conclusion et perspectives

Nous faisons une synthèse des contributions importantes de notre travail et nous proposons des perspectives à la fois scientifiques et industrielles.

CHAPITRE 1

Détection de piétons dans le spectre visible

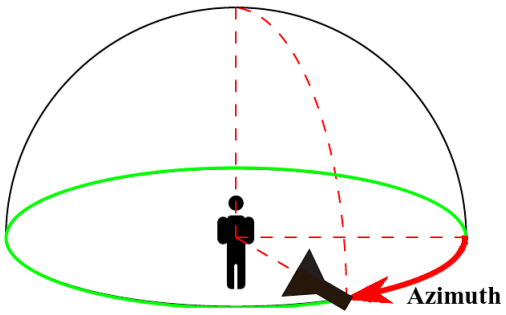


FIGURE 1.1 – La vue piéton correrspond à la vue obtenue lorsque la caméra est positionnée sur la base de la demi-sphère, c'est-à-dire, lorsque l'angle d'azimuth est compris entre 0 et 360 degrés, et que la caméra pointe en direction et à mi-hauteur des personnes.

FIGURE 1.2 – Exemples d'images de personnes prises dans la rue et en vue piéton (images extraites de la base de données INRIA).

La détection automatique de piétons est un sujet très étudié dans le domaine de la vision par ordinateur. Le but est de détecter automatiquement à partir d'images ou de flux vidéos des personnes en vue piéton (Fig.1.1 et Fig.1.2). Un travail continu de recherche a été mené pour réaliser des détecteurs de piétons qui soient à la fois robustes, rapides et performants depuis les dix dernières années. En effet, le détecteur de piétons sera une composante indispensable des prochains systèmes

d'aide à la conduite (ou en anglais les systèmes "ADAS [1]", pour "Advanced Driver Assistance Systems"). Ces systèmes devraient être disponibles dans quelques années pour permettre de réduire drastiquement le nombre de tués et de blessés par l'anticipation des accidents de la route. Les piétons sont particulièrement vulnérables dans les villes. Le détecteur de piétons pourrait permettre, par exemple, de réduire la vitesse du véhicule à proximité de piétons, ou tout simplement, d'arrêter le véhicule si un ou des piétons se trouvent sur sa trajectoire.

Bien que les cas d'utilisation soient différents, il existe quelques similitudes entre le contexte de la détection de personnes à partir de drones et le contexte de la détection de piétons. Ces similitudes rendent intéressants l'étude des détecteurs de piétons dans notre contexte. En effet, dans les deux cas le détecteur a pour contraintes d'utilisation : 1) d'être réactif sur un système embarqué (par définition moins puissant), 2) de détecter des personnes à des distances très différentes, 3) d'être capable de détecter quelque soit le lieu, 4) d'être robuste aux changements d'illumination qui peuvent survenir durant la journée et 5) d'être capable de détecter aussi bien des personnes mobiles qu'immobiles. Presque toutes les contraintes sont retrouvées dans les deux cas d'utilisation.

1.1 La détection de piétons

1.1.1 Approches de détection

Dans la littérature, on peut distinguer deux approches de détection différentes : 1) l'approche de détection guidée par les régions d'intérêt et 2) l'approche de détection exhaustive, guidée par une fenêtre de détection balayée sur toute l'image et pour plusieurs niveaux.

(1) Dans le premier cas, on considère que l'extraction de régions d'intérêt permet d'identifier la grande majorité des êtres humains de la scène. Un critère simple peut être utilisé ensuite pour confirmer ou infirmer la présence d'un être humain pour chaque zone d'intérêt précédemment extraite. Cela peut être un critère sur la forme (utilisant des contours, tel que proposé par Toth et al. [Toth 2003]) ou encore un critère sur l'apparence visuelle (utilisant un "codebook" visuel, tel que proposé par Zhou et al [Zhou 2005]). La soustraction de fond est basée sur le mouvement, ce qui limite l'approche à la détection de personnes en mouvements. Nous verrons dans ce manuscrit qu'il est possible de procéder à une extraction de régions d'intérêt qui ne dépend pas du mouvement.

L'approche de détection 2) considère tout l'espace de recherche. Aucune réduction de l'espace de recherche n'est effectuée en amont, car toute zone de l'image est potentiellement une personne. Ici, généralement, un critère de détection plus

1. *Advanced Driver Assistance Systems*, Système d'aide à la conduite

élaboré (mais aussi plus coûteux) est utilisé pour infirmer ou confirmer la présence d'un être humain pour chaque zone de l'image. Pour la plupart, les détecteurs utilisés avec cette approche sont des détecteurs supervisés.

Par la suite, nous allons plus particulièrement nous intéresser aux détecteurs supervisés qui permettent une analyse en profondeur des images pour trouver des personnes. Nous allons présenter les chaînes de traitement des détecteurs supervisés de piétons, pour une bonne compréhension du fonctionnement de ceux-ci.

1.1.2 Chaînes de traitement des détecteurs supervisés

Une grande majorité des détecteurs les plus performants de l'état de l'art sont des détecteurs supervisés [Dollár 2009a]. La détection supervisée a été introduite dans les travaux de Papagorgiou et al [Papageorgiou 2000] puis popularisée par les travaux de Viola et Jones en 2001 [Viola 2001]. Un détecteur supervisé est un détecteur utilisant un classifieur qui a été entraîné en utilisant un large ensemble d'images de référence (ou images d'entraînement). L'entraînement peut être vu comme la phase d'apprentissage d'un modèle humain générique. Plus la base de données d'entraînement est riche, plus le modèle final sera générique et donc plus son utilisabilité sera étendue. Cependant les performances de détection du détecteur supervisé ne dépendent pas entièrement de la richesse des images d'entraînement ; les caractéristiques visuelles utilisées pour construire le modèle ont aussi une grande importance.

Entraînement du classifieur

L'entraînement du classifieur d'un détecteur supervisé (Fig.1.3) est effectué en trois étapes successives : 1) le chargement d'images d'entraînement positives (images de personnes) et négatives (images de fond quelconques), 2) l'extraction des caractéristiques visuelles de chaque image (négative et positive) et 3) l'entraînement du classifieur en se basant sur toutes les caractéristiques visuelles précédemment extraites. Certains détecteurs nécessitent une quatrième phase d'optimisation du classifieur, pour permettre une utilisation plus rapide [Bourdev 2005].

Phase de détection

La phase de détection du détecteur supervisé (Fig.1.4) est effectuée en trois étapes successives : 1) la zone de l'image à analyser est récupérée, 2) les caractéristiques visuelles sont extraites de la zone de l'image et 3) elles sont envoyées au classifieur binaire qui retourne une réponse positive (personne détectée) ou négative.

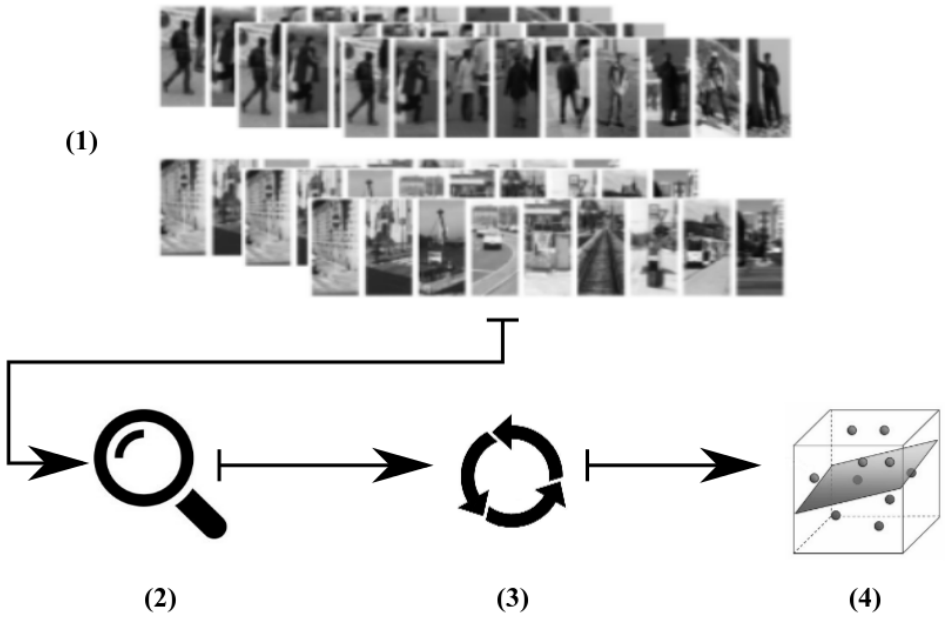


FIGURE 1.3 – Entraînement d'un détecteur supervisé.

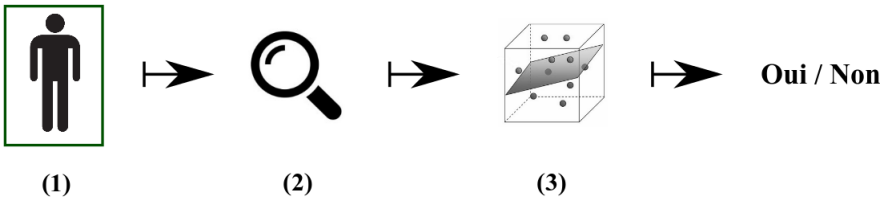


FIGURE 1.4 – Détection supervisée.

Nous allons détailler chaque élément de la chaîne de traitement sans la suite du document.

1.2 Caractéristiques visuelles

Pour trouver un objet particulier dans une scène, le cerveau recherche les caractéristiques visuelles spécifiques de l'objet. Ces caractéristiques ne sont pas ou peu changeantes et permettent à elles seules de reconnaître l'objet parmi les autres objets de la scène ; cela peut être une combinaison de caractéristiques telles que : la texture, la couleur, la taille, et la forme générale de l'objet. En vision par ordinateur, l'approche de recherche de caractéristiques est comparable.

On peut dissocier deux types de recherche d'objets : 1) la reconnaissance d'ob-

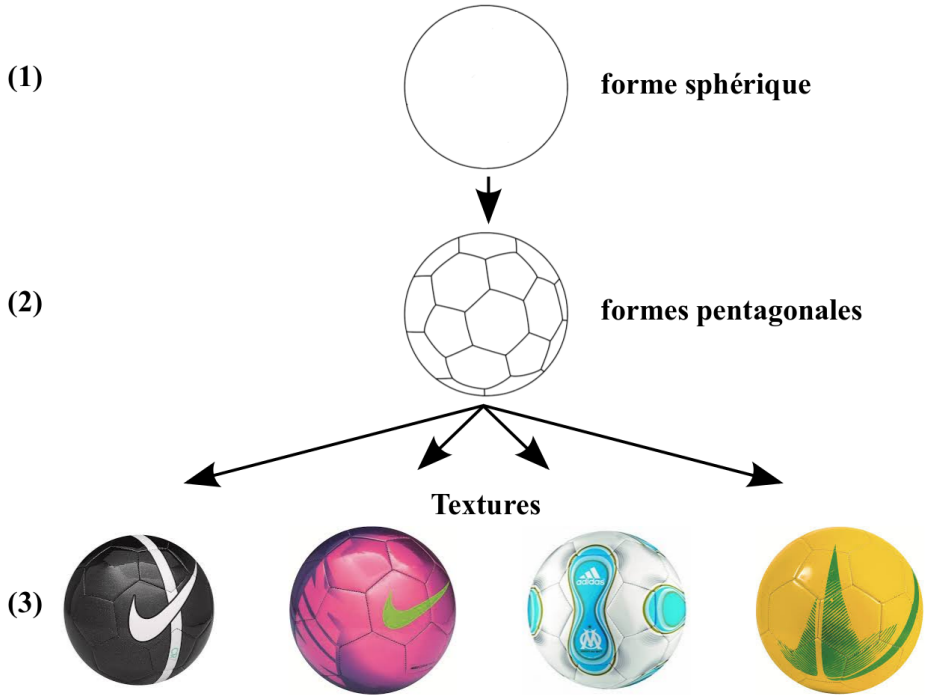


FIGURE 1.5 – Hiérarchie des caractéristiques visuelles pour des objets de la classe "ballon", allant des caractéristiques géométriques aux caractéristiques texturales.

jets (recherche d'instances d'objets) et 2) la détection d'objets (recherche d'objets d'une même classe). La reconnaissance d'objets s'attache à retrouver un objet en particulier (par exemple, spécifiquement le ballon noir de la Fig.1.5. Dans le cas de la détection une combinaison de caractéristiques visuelles plus globales permet de trouver des objets similaires tel que des objets de la même classe Par exemple, en choisissant des caractéristiques visuelles jusqu'au niveau hiérarchique 2, pour trouver tous les ballons de la Fig.1.5.

Les caractéristiques visuelles peuvent également être vues comme une façon de réduire la quantité d'information nécessaire pour décrire l'apparence d'un objet. Ainsi, l'apparence du ballon noir de la Fig.1.5 (faisant 100 par 100 pixels) peut être décrite grâce à seulement quatre caractéristiques visuelles : sa forme sphérique, la présence de formes pentagonales à sa surface, sa couleur noir ainsi que son logo. La quantité d'information est donc divisée par 2500.

En détection d'objets, les caractéristiques visuelles idéales doivent avoir un caractère global, et aussi, ne pas changer de nature suivant l'orientation de l'objet, la distance de celui-ci avec l'observateur et les changements d'illumination. À noter, qu'une grande importance est donnée au temps de calcul nécessaire à l'extraction des caractéristiques. En effet, celles-ci sont souvent extraites un très grand nombre de fois lors de l'analyse d'une image.

Dans la suite de cette section quatre types de caractéristiques visuelles vont être étudiées, les très populaires : caractéristiques Pseudo-Haar et les Histogrammes de Gradients Orientés (HOG [2]) ainsi que les plus récentes et rapides : caractéristiques de canaux intégraux (ICF [3]) et caractéristiques de canaux agrégés (ACF [4]).

1.2.1 Histogrammes de Gradients Orientés

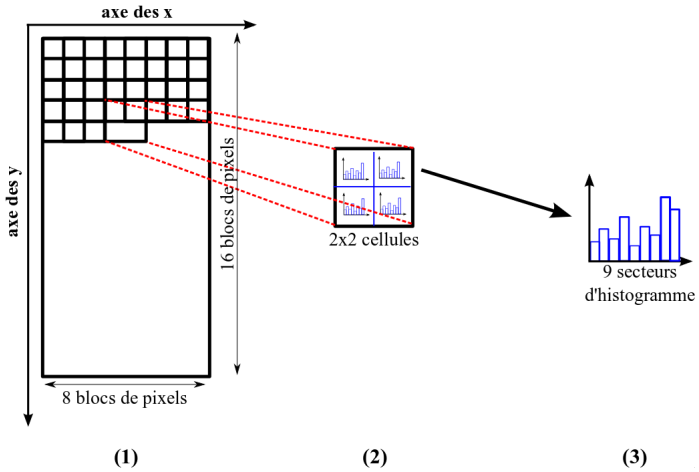


FIGURE 1.6 – Processus d'extraction des histogrammes de gradients orientés : séparation par blocs de la fenêtre et calcul de quatre histogrammes pour chaque bloc.

Les Histogrammes de Gradients Orientés sont des caractéristiques visuelles très populaires dans le domaine de la détection de personnes. Un grand nombre de détecteurs de personnes utilisent ces caractéristiques ou une version modifiée de ces caractéristiques [Dollár 2009a]. Le détecteur du même nom, proposé par Dalal et Triggs, est cité en référence dans un grand nombre de travaux [Dalal 2005]. Les Histogrammes de Gradients Orientés ont été inspirés des travaux de David Lowe sur le détecteur de points d'intérêts SIFT [5] (pour Scale Invariant Feature Transform) [Lowe 1999]. À la différence du SIFT, les histogrammes sont calculés par blocs et pour une zone élargie qui correspond à la fenêtre de détection (Fig.1.6).

2. *Histogram of Oriented Gradients*, Histogramme de gradients orientés
3. *Integral Channel Features*, Caractéristiques de Canaux Intégraux
4. *Aggregate Channel Features*, Caractéristiques de Canaux Agrégés
5. *Scale Invariant Transform Feature*, Caractéristique invariante aux transformations d'échelles

Organisation des blocs

L'organisation des blocs d'histogrammes à l'intérieur de la fenêtre de détection est telle que les blocs se chevauchent de moitié les uns avec les autres. Chacun des blocs est normalisé (normalisation L2-Hys) pour permettre une plus grande robustesse aux changements locaux d'illumination (Equ.1.7). Dalal et Triggs proposèrent plusieurs configurations différentes de cellules et de pixels par cellule [Dalal 2005]. La configuration présentée dans ce manuscrit utilise des blocs de 2x2 cellules qui elles-mêmes contiennent 8x8 pixels. Avec cette configuration : 128 blocs se chevauchent dans une fenêtre de détection de 64 par 128 pixels. Cette configuration présente l'avantage d'être efficace et de produire un nombre raisonnable d'histogrammes [Dalal 2005].

Calcul des histogrammes

Le calcul des Histogrammes de Gradients Orientés se fait en plusieurs étapes :

(1) On calcule les gradients suivant la direction de l'axe des x et suivant la direction de l'axe des y pour les trois canaux RGB. Les gradients sont calculés en utilisant les filtres présentés ci-dessous :

$$G_x = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} \tag{1.1}$$

$$G_y = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} \tag{1.2}$$

(2) L'amplitude et l'orientation du gradient de chaque pixel sont calculées en prenant le gradient maximal suivant la direction de l'axe x (pour les trois canaux), et le gradient maximal suivant la direction de l'axe y (pour les trois canaux) :

$$G_x^{max}(x,y) = max_{c\in\{R,G,B\}} G_x^c(x,y) \tag{1.3}$$

$$G_y^{max}(x,y) = max_{c\in\{R,G,B\}} G_y^c(x,y) \tag{1.4}$$

$$G(x,y) = \sqrt{G_x^{max}(x,y)^2 + G_y^{max}(x,y)^2} \tag{1.5}$$

$$\theta(x,y) = atan((\widehat{G_x^{max}(x,y), G_y^{max}}(x,y))) \tag{1.6}$$

L'orientation du gradient du pixel est normalisé entre 0 et 180 degrés.

(3) Pour chaque cellule de chaque bloc, on calcule un histogramme d'orientation des gradients allant de 0 à 180 degrés et contenant 9 secteurs de 20 degrés.

L'histogramme est construit de la manière suivante : pour chaque pixel de la cellule, la valeur de l'amplitude du gradient du pixel est distribuée de manière pondérée pour chaque paire de secteurs d'histogrammes. La pondération est calculée en se basant sur la différence de degrés qu'il y a entre l'orientation du pixel et les centres des secteurs.

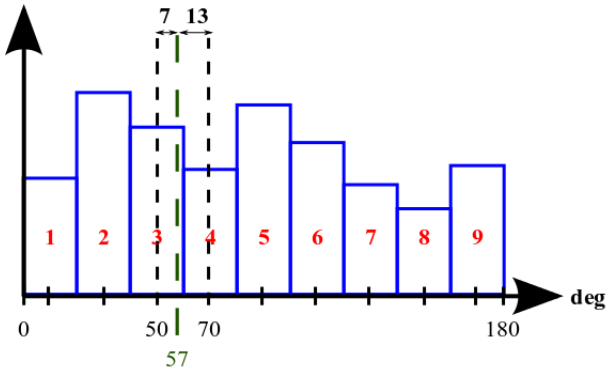


FIGURE 1.7 – Pondération du remplissage de l'histogramme d'orientation de gradients

Pour un pixel ayant une valeur d'orientation de 57 degrés (Fig.1.7) : on affectera au secteur 3 l'amplitude de gradient du pixel multipliée par $(1 - \frac{(57-50)}{20})$ et on affectera au secteur 4 l'amplitude de gradient du pixel cette fois ci multipliée par $(1 - \frac{(70-57)}{20})$.

(4) Pour finir, les histogrammes $\vec{v}$ sont normalisés par blocs en utilisant la norme L2-Hys :

$$\vec{n} = \frac{\vec{v}}{\sqrt{\|v\| + e}} \tag{1.7}$$

Où $\vec{n}$ est le vecteur normalisé. Les composantes du vecteur $\vec{v}$ supérieures à 0.2 sont fixées à 0.2, tel qu'il est conseillé par Lowe [Lowe 1999]. Dans Equ.1.7, e est une valeur epsilon quelconque.

1.2.2 Caractéristiques Pseudo-Haar

Les caractéristiques Pseudo-Haar ont été décrites pour la première fois par Viola et Jones dans leurs travaux concernant la détection automatique de visages [Viola 2001]. Les auteurs se sont inspirés des travaux de Papageorgiou et al qui proposèrent des caractéristiques visuelles basées sur les ondelettes de Haar [Papageorgiou 1998]. Les caractéristiques Pseudo-Haar ont l'avantage d'être très simples et très rapides à calculer. Elles permettent de capturer simplement les changements d'intensités dans une image en niveau de gris. Il est possible de capturer

des formes complexes en combinant plusieurs de ces caractéristiques Pseudo-Haar entre elles, ce qui a pour effet d'augmenter le pouvoir discriminant.

Les types de caractéristiques Pseudo-Haar

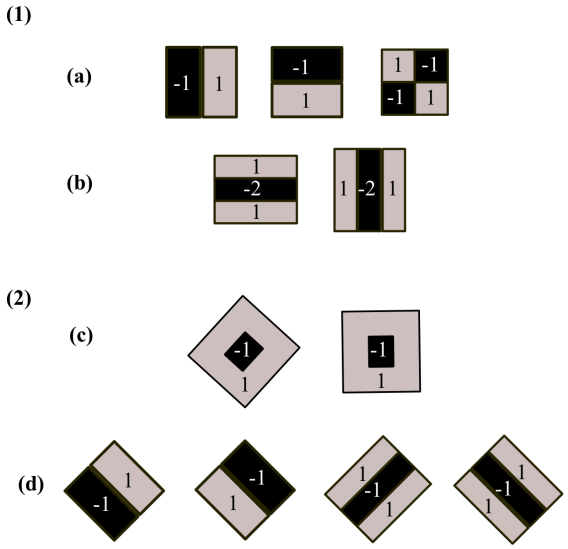


FIGURE 1.8 – Ensemble de caractéristiques Pseudo-Haar proposé pour la détection d'objets

Viola et Jones ont décrit plusieurs types de caractéristiques Pseudo-Haar (Fig.1.8). Certaines permettent de capturer des changements d'intensités comparables à ceux observés lors d'une traversée horizontale, verticale ou oblique des bords d'un objet (Fig.1.8 1.a) et certaines permettent de capturer des changements d'intensités comparables à ceux observés lors d'une traversée horizontale ou verticale d'une bordure (Fig.1.8 1.b).

À la suite des travaux de Viola et Jones, Lienhart et al proposèrent une extension des caractéristiques Pseudo-Haar [Lienhart 2002]. Ils proposèrent un ensemble de caractéristiques orientées à 45 degrés (Fig.1.8.2) qui permettent d'étendre le domaine d'apprentissage à des caractéristiques plus pertinentes pour la détection de personnes. Des caractéristiques centre-pourtour ont également été proposées pour capturer les changements d'intensités qui sont centrés (Fig.1.8.2). Les auteurs prétendent obtenir une amélioration des performances de près de 10% en utilisant cette extension de caractéristiques par rapport à l'usage seul des caractéristiques proposées par Viola et Jones [Lienhart 2002].

Calcul d'une caractéristique Pseudo-Haar

La valeur d'une caractéristique Pseudo-Haar (V) est calculée comme définie ci-dessous :

$$V = \sum_{r \in R} (signe(r) \times \sum_{\substack{x_1^r < x \leq x_2^r \\ y_1^r < y \leq y_2^r}} i(x, y)) \tag{1.8}$$

La somme des pixels contenus dans la (ou les) zone(s) grisée(s) est soustraite à la somme des pixels contenus dans la (ou les) zone(s) noire(s) (Fig.1.8). R est l'ensemble des rectangles de la caractéristique, $signe(r)$ est le signe associé au rectangle r et $i(x, y)$ est l'intensité au point (x, y) du rectangle r. Le rectangle r est définit tel que (x_1^r, y_1^r) correspond à son coin haut gauche et tel que (x_2^r, y_2^r) correspond à son coin bas droit.

Calcul rapide d'une caractéristique Pseudo-Haar

Le calcul de la somme des pixels d'un rectangle est relativement rapide. Cependant, si l'opération est répétée un trop grand nombre de fois, les temps de calcul du détecteur peuvent être impactés lourdement. Viola et Jones proposèrent une technique permettant de calculer la somme des intensités de pixel d'un rectangle à partir des quatre points d'extrémité du rectangle (Fig.1.9) :

$$\sum_{\substack{x_1 < x \leq x_2 \\ y_1 < y \leq y_2}} i(x, y) = I(x_2, y_2) + I(x_1, y_1) - I(x_2, y_1) - I(x_1, y_2) \tag{1.9}$$

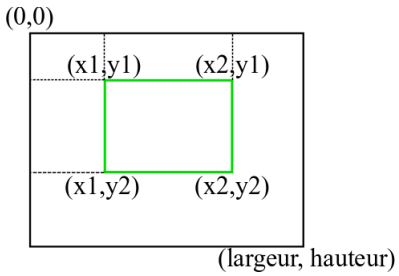


FIGURE 1.9 – Les quatre points d'extrémité d'un rectangle suffisant à calculer la somme des intensités de pixel de celui-ci.

Calculer la somme des pixels de rectangles de tailles très différentes demande le même effort de calcul. D'une manière générale, le temps pour calculer la somme d'un rectangle est constant. Cette optimisation est basée sur l'utilisation d'une

image intégrale. L'image intégrale d'une image est définie telle que la valeur de ses pixels est calculée suivant :

$$I(x,y) = \sum_{\substack{x' \leq x \\ y' \leq y}} i(x',y') \tag{1.10}$$

Où $i(x',y')$ est l'intensité de l'image (en niveau de gris) à la position (x',y'). La valeur $I(x,y)$ de l'image intégrale correspond simplement à la somme de tous les intensités de l'image qui sont contenues dans le rectangle de coin supérieur gauche $(0,0)$ et de coin inférieur droit (x,y).

Le calcul de l'image intégrale peut être effectué rapidement et en une seule étape en utilisant la relation récursive suivante :

$$I(x,y) = i(x,y) + I(x-1,y) + I(x,y-1) - I(x-1,y-1) \tag{1.11}$$

Les valeurs de la première colonne et de la première ligne de l'image intégrale I sont nulles.

Une image intégrale calculée à 45 degrés doit être construite pour calculer les caractéristiques Pseudo-Haar additionnelles proposées par Lienhart et al.

1.2.3 Caractéristiques de Canaux Intégraux (ICF)

Les caractéristiques de canaux intégraux (ou «Integral Channel Features» en anglais, ou encore simplement «ICF») ont été proposées par Dollár et al en 2009 [Dollár 2009b]. Ces caractéristiques partagent un certain nombre de similitudes avec les caractéristiques Pseudo-Haar : 1) la primitive de base est également le calcul de la somme des pixels contenus dans un rectangle et 2) elles permettent aussi de calculer des différences (mais sur des canaux de natures très différentes et d'une manière décorrélée dans l'image).

Approche

Un très grand nombre de caractéristiques de forme rectangulaire, ayant des positions et des tailles différentes, sont générées aléatoirement sur plusieurs canaux. Dollár et al recommandent de générer 30 000 caractéristiques candidates. Seules les caractéristiques les plus pertinentes (trouvées à l'apprentissage durant l'étape de sélection de caractéristiques) sont utilisées (Fig.1.10). Le principe est de combiner ces caractéristiques de base entre elles, quels que soient leurs positions, leurs tailles et leurs canaux associés dans le but d'accroître le pouvoir discriminant. Les caractéristiques de canaux intégraux sont calculées sur dix canaux différents. Pour

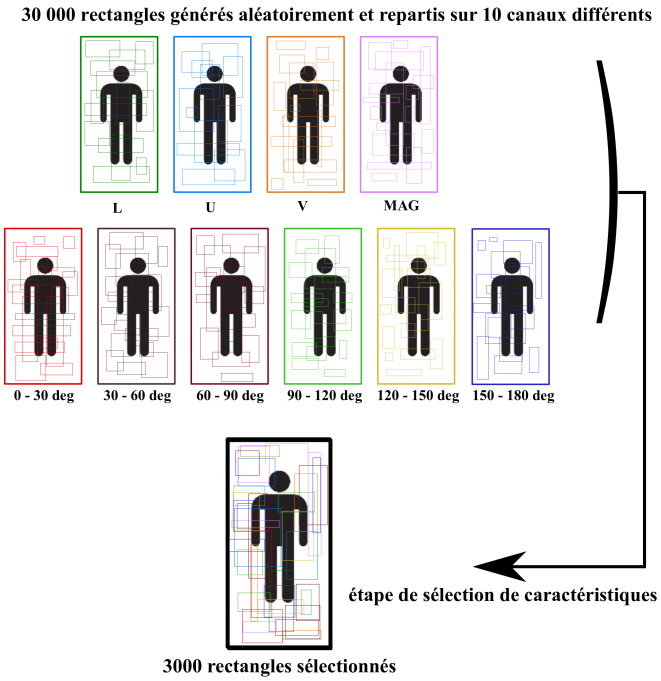


FIGURE 1.10 – Génération aléatoire des caractéristiques de canaux intégraux sur les dix canaux, et sélection des caractéristiques les plus pertinentes

chaque canal, une image intégrale est calculée dans le but d'accélérer les temps de calcul.

Calcul des canaux

Les dix canaux utilisés sont : les trois canaux de couleurs LUV, le canal "amplitude de gradient" et six canaux des orientations des gradients. Dollár et al testèrent plusieurs combinaisons différentes de canaux, et ils obtinrent les meilleurs résultats avec cette combinaison de canaux [Dollár 2009b]. Plusieurs étapes sont nécessaires à l'obtention de ces dix canaux : 1) deux changements d'espaces de couleurs doivent être réalisés pour obtenir les nouveaux canaux de couleurs LUV et 2) chaque canal d'orientation des gradients est calculé pour une plage de 30 degrés ; la totalité couvrant ainsi la plage de 0 à 180 degrés (Fig.1.10).

(1) Les canaux RGB de l'image doivent d'abord être convertis dans l'espace de couleur CIEXYZ tel que définit dans [Cie 1931] et décrit ci dessous :

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \frac{1}{0.17697} \begin{bmatrix} 0.49 & 0.31 & 0.20 \\ 0.17697 & 0.81240 & 0.01063 \\ 0.00 & 0.01 & 0.99 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{1.12}$$

Les canaux CIEXYZ sont finalement convertis en canaux CIELUV tel que dé-

crit ci dessous :

$$L = \begin{cases} 116\sqrt[3]{y_r} - 16, & \text{si } y_r > \varepsilon, \\ \kappa y_r, & \text{si } y_r \leq \varepsilon \end{cases} \tag{1.13}$$

$$U = 13L(u' - u'_r) \tag{1.14}$$

$$V = 13L(v' - v'_r) \tag{1.15}$$

$$y_r = \frac{Y}{Y_r} \tag{1.16}$$

$$u' = \frac{4X}{X + 15Y + 3Z} \tag{1.17}$$

$$v' = \frac{9X}{X + 15Y + 3Z} \tag{1.18}$$

$$u'_r = \frac{4X_r}{X_r + 15Y_r + 3Z_r} \tag{1.19}$$

$$v'_r = \frac{9Y_r}{X_r + 15Y_r + 3Z_r} \tag{1.20}$$

Où X_r, Y_r et Z_r définissent le blanc de référence, $\varepsilon = 0.008856$ et $\kappa = 903.3$.

Le passage de l'espace de couleur CIEXYZ à l'espace de couleur CIELUV nécessite le calcul d'une racine cubique, ce qui requiert l'exécution d'un grand nombre d'instructions par le processeur. Dollár et al préconisent d'utiliser, à la place, une approximation grossière de la racine cubique. Cela affecte peu les performances du détecteur et permet de réduire considérablement le nombre d'instructions nécessaires au calcul. Une approximation suffisante peut être obtenue après quelques itérations de la méthode de Newton, par exemple.

(2) Les six canaux des orientations des gradients partagent quelques similarités avec les Histogrammes de Gradients Orientés présentés Sec.1.2.1 : on calcule l'amplitude et l'orientation des gradients tel qu'il est décrit dans Equ.1.5 et Equ.1.6. Les six canaux des orientations couvrent la plage d'angle 0 - 180 degrés ; chaque canal correspond à une plage de 30 degrés. La valeur des pixels des canaux est trouvée en utilisant la même approche de pondération que celle présentée Fig.1.7.

1.2.4 Caractéristiques de Canaux Agrégés (ACF)

Pour calculer les caractéristiques de canaux agrégés on utilise les dix mêmes canaux que ceux utilisés pour calculer les caractéristiques de canaux intégraux [Dollár 2014]. À la différence des caractéristiques de canaux intégraux, les canaux ici subissent une transformation particulière : une agrégation. Ceci n'est pas la seule différence avec la méthode précédente. Dans le cas présent, la primitive utilisée est différente : il ne s'agit plus du calcul de la somme des pixels contenus à l'intérieur d'un rectangle ou du calcul d'un histogramme local, mais tout simplement de l'extraction de la valeur d'un pixel dans un des dix canaux agrégés.

Calcul des canaux

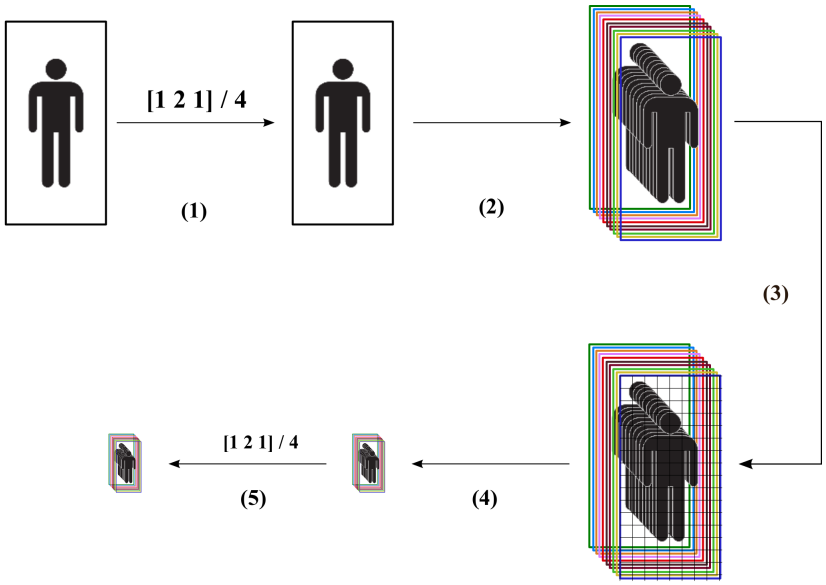


FIGURE 1.11 – Étapes successives permettant l'obtention des canaux agrégés d'une image.

La chaîne de traitement permettant d'obtenir les canaux agrégés contient des étapes de filtrage ainsi qu'une étape d'agrégation des canaux. Les canaux sont obtenus comme il suit :

1. Un filtre triangulaire (Equ.1.21) est appliqué sur les trois canaux RGB de l'image d'entrée (Fig.1.11.1).
2. Les dix canaux (L, U, V, amplitude de gradients et les six canaux d'orientations des gradients) sont calculés de la même manière que pour les canaux de l'ICF (Fig.1.11.2).

3. Chaque canal est découpé de manière uniforme en blocs de 4x4 pixels (Fig.1.11.3).
4. Chaque canal est agrégé : cela signifie que, pour chaque bloc, on additionne les pixels contenus dans le bloc et que seule cette dernière valeur est gardée permettant ainsi de réduire la résolution du canal (Fig.1.11.4).
5. Chaque canal agrégé une nouvelle étape de filtrage est réalisée (Fig.1.11.5) grâce au filtre présenté Equ.1.21.

$$C_f = C * \frac{\begin{bmatrix}1 & 2 & 1\end{bmatrix}}{4} \tag{1.21}$$

Où C_f symbolise le canal filtré, C le canal avant filtrage, $*$ l'opération de convolution.

1.3 Approximation des caractéristiques visuelles

Le calcul des caractéristiques visuelles pour tous les niveaux de la pyramide d'image prend beaucoup de temps de calcul. Dollàr et al observèrent que les caractéristiques visuelles calculées pour un niveau de l'image peuvent être approximées pour les niveaux voisins en utilisant une loi exponentielle [Dollár 2010][Dollár 2014]. D'après Dollàr et al, le ratio d'énergie E des caractéristiques calculées par f d'une image I à une échelle e_1 et les caractéristiques calculées par f de la même image I sous-échantillonnée à une échelle e_0 est indépendante de l'échelle de l'image originale e_1, et elle dépend uniquement d'un rapport d'échelle $\frac{e_1}{e_0}$ multiplié par une valeur λ [Dollár 2010] :

$$E(f(I, e_1)/f(I, e_0)) = e^{-\lambda \times \frac{e_1}{e_0}} \tag{1.22}$$

Grâce à cette propriété, il n'est plus nécessaire de calculer explicitement les caractéristiques visuelles pour chaque niveau de la pyramide mais uniquement pour un sous-ensemble d'entre eux. Il suffit ensuite de procéder à une approximation des caractéristiques visuelles pour obtenir les caractéristiques des niveaux intermédiaires.

L'approximation des caractéristiques est formulée comme il suit :

$$f(I, e_0) \approx f(I, 0)e^{-\lambda \times e_0} \tag{1.23}$$

Le paramètre d'approximation λ doit être adapté suivant que le niveau est sous-échantillonné ou est sur-échantillonné. Ce paramètre est également différent suivant la nature des caractéristiques visuelles approximées. Dollàr et al portèrent une attention particulière au choix de la valeur λ pour les caractéristiques visuelles basées gradients [Dollár 2010][Dollár 2014].

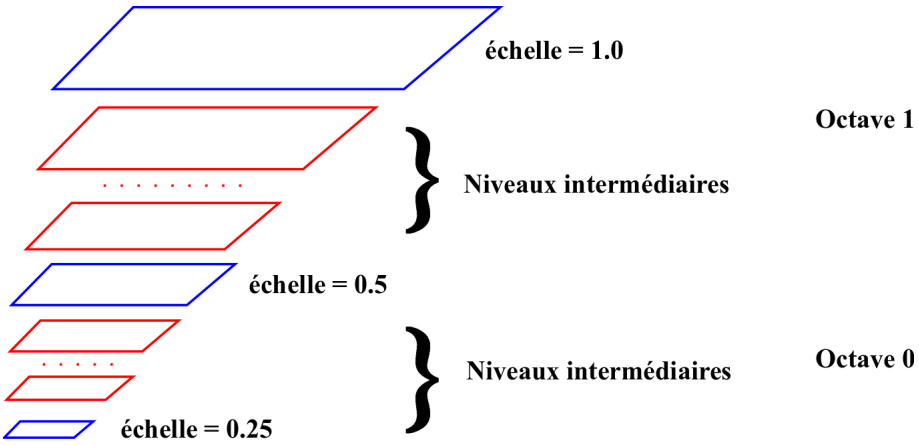


FIGURE 1.12 – Pyramide d'images composée de deux octaves.

Plus la différence entre l'échelle d'origine et l'échelle d'approximation est grande, plus la qualité d'approximation des caractéristiques diminue. Afin de limiter la dégradation de la qualité des caractéristiques, la pyramide d'image est subdivisée en octaves. Après chaque octave, la surface du niveau d'entrée est divisée par 4. Les caractéristiques visuelles du premier niveau de chaque octave sont extraites de manière conventionnelle. Pour les autres niveaux de l'octave, les caractéristiques sont approximées en utilisant la loi exponentielle et les paramètres λ de chaque type de caractéristiques. Dans Fig.1.12 les niveaux en rouges sont les niveaux intermédiaires et les niveaux en bleus les niveaux d'origine, à partir desquels sont approximés les niveaux rouges.

Les gains en temps de calcul obtenus avec cette approche sont importants [Dollár 2010]. Cette optimisation permet d'atteindre quasiment le temps réel pour des cas simples (image d'entrée de faible résolution et pyramide peu profonde).

1.4 Apprentissage

L'étape d'apprentissage permet l'obtention d'un classifieur ; celui ci permet de déterminer la classe de nouvelles données d'entrée dont on ne connaît pas la nature. Dans le contexte de la détection de personnes nous avons deux classes d'objets différentes : la classe objet "humain" et la classe objet "fond". Lors de l'analyse de l'image nous souhaitons être en mesure de dissocier ces deux classes. L'utilisation de techniques d'apprentissage pour la détection d'objets a été popularisée par les travaux de Papageorgiou et al [Papageorgiou 2000]. De nos jours, beaucoup de détecteurs de personnes se basent directement ou indirectement sur le principe proposé par ces auteurs [Dalal 2005][Viola 2001][Dollár 2009a].

1.4.1 Principe général

Plusieurs approches d'apprentissage sont envisageables : 1) l'approche supervisée utilise des données d'entraînement dont les classes sont explicitement connues, 2) l'approche non-supervisée utilise des données d'entraînement dont on ne connaît pas la nature (les classes sont découvertes durant l'apprentissage) et 3) l'approche semi-supervisée consistant à mélanger les deux approches précédentes : pour une partie des données d'entraînement les classes sont connues et on ne connaît pas les classes des données d'entraînement pour une autre partie des données.

(1) L'approche supervisée est plus rapide à mettre en place et génère des classifieurs relativement performants. Cependant, elle nécessite l'utilisation d'une base de données d'entraînement labélisées manuellement. La création d'une base de données d'entraînement peut être fastidieuse. De plus, les performances du classifieur généré avec cette approche dépendent de la richesse de la base et de sa pertinence par rapport au cas d'utilisation.

(2) L'approche non-supervisée présente un avantage majeur : aucune labélisation manuelle n'est nécessaire. Pour l'apprentissage de classes d'objets complexes, on préférera une approche supervisée ou semi-supervisée.

(3) L'approche semi-supervisée nécessite moins de données d'entraînement que l'approche supervisée. Les performances d'un classifieur entraîné avec cette approche peuvent surpassées celles d'un classifieur entraîné avec une approche supervisée [Krishnapuram 2004]. Cependant, cette approche est plus complexe à mettre en place.

1.4.2 L'apprentissage d'un classifieur

Comme énoncé plus haut, dans le cas de la détection de personnes nous avons deux classes d'objets différentes : la classe objet "humain" (appelée aussi classe positive) et la classe objet "fond" (appelée aussi classe négative). Dans la base de données d'entraînement, la classe objet "humain" est constituée d'images de personnes, et la classe "fond" est constituée d'images de fond quelconque, c'est-à-dire tout ce qui n'est pas une personne dans une image (Fig.1.13). Il existe de nombreuses bases de données disponibles pour apprendre la classe objet "humain" et la classe objet "fond", telles que : la base de données INRIA [6] ou la base de données CalTech [Dalal 2005][Dollár 2009a].

Concrètement, l'apprentissage consiste en l'obtention d'une relation mathématique permettant de séparer au mieux les données d'entraînement négatives des données d'entraînement positives dans l'espace des caractéristiques visuelles. Cette

6. *Institut National de Recherche en Informatique et en Automatique*, Institut de recherche en informatique

FIGURE 1.13 – Exemple d'images d'entraînement positives (1) et négatives (2) de la base de données INRIA.

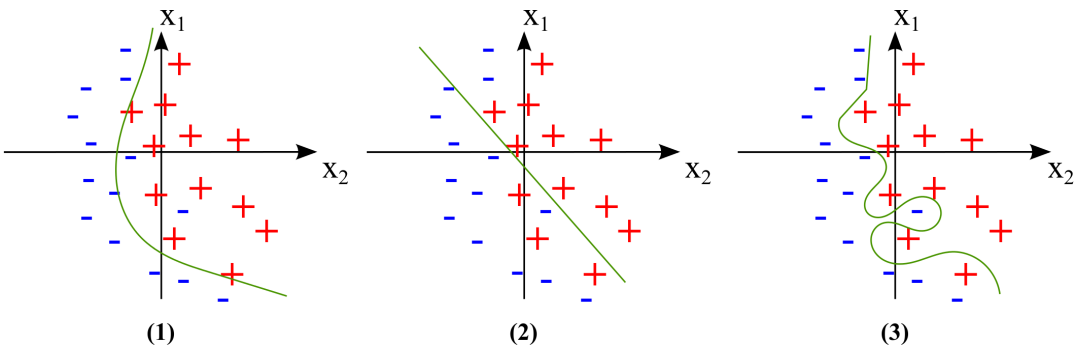


FIGURE 1.14 – Exemple de trois séparations de classes différentes obtenues après trois apprentissages différents (cas normal, cas "overfitting" et cas "underfitting")

séparation des classes doit être à la fois suffisamment précise et suffisamment généralisable pour obtenir de bonnes performances de classification (Fig.1.14.1) :

(1) L'erreur de classification sur les images d'entraînement doit être faible. Si la séparation n'est pas assez précise (Fig.1.14.2), cela revient à dire que le classifieur n'a pas une bonne connaissance des classes : la séparation obtenue correspond faiblement aux classes (on dit que l'on sous-ajuste, en anglais : "underfitting").

(2) L'erreur de généralisation doit être faible. Si la séparation est trop précise sur les données d'entraînement (Fig.1.14.3), l'erreur de généralisation sera très importante (on dit que l'on sur-ajuste, en anglais : "overfitting"). Dans ce cas, les performances de classification seront très bonnes pour classifier les images d'entraînement et très médiocres pour classifier de nouveaux cas. Le classifieur généré est donc d'aucune utilité. Ce problème peut apparaître si : la quantité d'images

d'entraînement est insuffisante ou alors si les paramètres d'apprentissage ne sont pas adéquats.

De nombreux algorithmes d'apprentissage ont été proposés dans la littérature. Chaque algorithme a ses avantages et ses inconvénients. Certains algorithmes d'apprentissage génèrent : des classifieurs rapides à exécuter, des classifieurs dont la structure présente des avantages lors de leurs évaluations, des classifieurs tenant peu de place en mémoire, etc. Le choix de l'algorithme d'apprentissage doit être fait en fonction du problème à résoudre. Dans cette partie nous allons essentiellement nous intéresser aux approches supervisées d'apprentissage. Nous verrons plus tard dans cette thèse comment l'approche semi-supervisée peut améliorer les performances du classifieur.

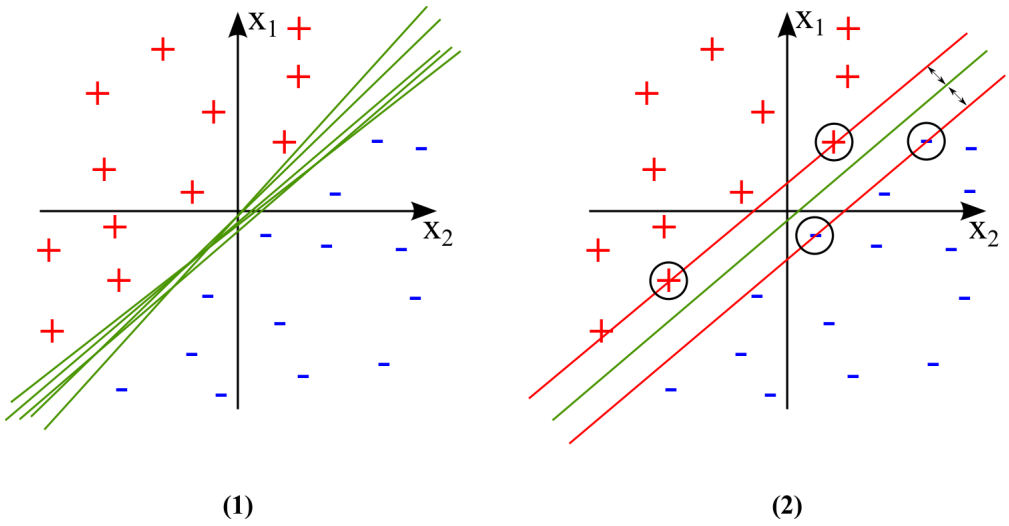
Nous nous intéressons ci-dessous à trois algorithmes d'apprentissage très populaires dans les travaux de recherche sur la détection de personnes : l'algorithme des séparateurs à vaste marge, le Boosting et le l'algorithme des séparateurs à vaste marge latent ("Latent SVM" ou "LSVM [7]", en anglais). Un quatrième algorithme sera également présenté : le réseau de neurones convolutionels. Plus de détails de fonctionnement seront donnés pour les deux premiers algorithmes car ceux-ci sont utilisés à de nombreuses reprises dans cette thèse.

1.4.3 Séparateurs à vaste marge

L'algorithme des Séparateurs à Vaste Marge (SVM [8]) est une méthode de classification récente nécessitant un apprentissage supervisé. Cette méthode a été introduite pour la première fois dans les travaux de Cortes et Vapnik en 1995 [Cortes 1995]. Elle est basée sur l'utilisation de noyaux mathématiques (dont la forme la plus usitée est le noyau linéaire). Les noyaux sont utilisés pour trouver une séparation entre les données d'entraînement positives et les données d'entraînement négatives. Le but du SVM est de trouver un classifieur qui va séparer les données négatives des données positives en maximisant la distance entre les deux classes.

Les données d'entrée sont des points à N dimensions pouvant appartenir à deux classes différentes (classe positive ou classe négative). Trouver l'hyperplan (dimension $N-1$) qui sépare au mieux les deux classes permet de modéliser un modèle mathématique de séparation qui peut être réutilisé par la suite pour prédire l'appartenance de nouvelles données à une des deux classes. Cependant, il existe une infinité d'hyperplans séparateurs entre les deux classes (en vert, Fig.1.15.1). Parmi tous ces hyperplans, un hyperplan est plus approprié que les autres, on dit qu'il est l'hyperplan optimal (en vert, Fig.1.15.2). À l'apprentissage, le SVM se charge de trouver cet hyperplan optimal, c'est-à-dire l'hyperplan séparateur qui maximise la

7. *Latent Support Vector Machine*, Séparateur à Vaste Marge Latent
8. *Support Vector Machine*, Séparateur à Vaste Marge



(1) (2)

FIGURE 1.15 – Exemples de séparations de deux classes en dimension 2, et séparation optimale obtenue grâce au Séparateur à Vaste Marge.

distance minimale entre les données d'entraînement négatives et les données d'entraînement positives. Ainsi, la séparation est moins sensible aux "effets de bords" pouvant survenir si la séparation est trop proche d'une classe plutôt que d'une autre (séparations montrées Fig.1.15.1) ; ces effets de bord peuvent conduire à la mauvaise classification de nouvelles données qui se trouveraient proches de l'hyperplan. En d'autres termes, cela permet d'obtenir une erreur de généralisation plus petite, ainsi, l'aptitude du classifieur à classifier correctement des données nouvelles est améliorée. De plus, une séparation à vaste marge peut permettre d'obtenir une classification des données quasi-optimale avec moins de données d'entraînement qu'avec d'autres méthodes d'apprentissage. Le modèle mathématique du classifieur généré durant la phase d'apprentissage du SVM présente un gros avantage en terme de taille. En effet, seuls les vecteurs de support sont nécessaires à la prédiction (points entourés de noir Fig.1.15.2).

Cas linéairement séparable

Les cas linéairement séparabales peuvent être résolus en utilisant une représentation linéaire du SVM tel que décrit dans ce paragraphe.

Soit Φ une fonction noyau et $h(x) = 0$ un hyperplan qui est défini dans un espace vectoriel à N dimensions :

$$h(x) = w^T \Phi(x) + b = 0 \tag{1.24}$$

Où $\Phi(x)$ est directement remplaçable par x dans ce cas la fonction noyau Φ est un noyau linéaire, car le problème est linéairement séparable. Cet espace vectoriel

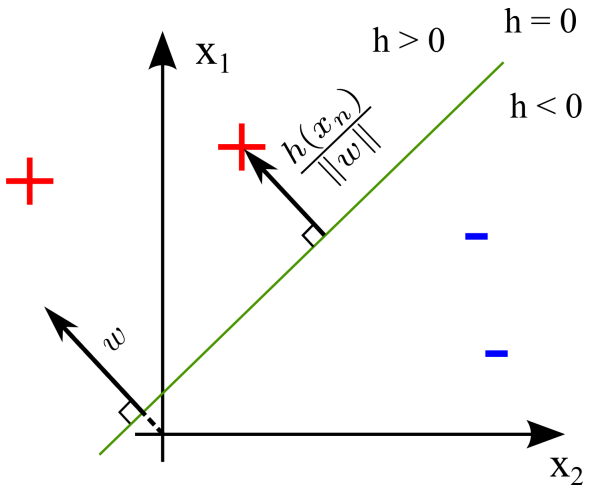


FIGURE 1.16 – Hyperplan séparateur de points.

est peuplé de points définis sur N dimensions. Un élément d'entraînement est défini comme un couple (x_n, t_n), où x_n est une position dans l'espace et t_n est une labélisation ($t_n = 1$ si la classe est positive et $t_n = -1$ si la classe est négative).

L'hyperplan est défini par les paramètres w et b : w donne l'orientation de l'hyperplan (il est perpendiculaire à celui-ci), b est le biais, il positionne l'hyperplan par rapport à l'origine. On considère que les éléments se situant au dessus de l'hyperplan font partie de la classe positive et que les éléments se situant en dessous de celui-ci font partie de la classe négative (Fig.1.16). Chaque élément d'entraînement (x_n, t_n) est à une distance $\frac{h(x_n)}{\|w\|}$ de l'hyperplan (Fig.1.16).

Par définition, la marge est la distance minimale entre les éléments d'entraînement et un hyperplan quelconque. Apprendre le classifieur consiste à trouver la marge maximale pour l'hyperplan qui classifie le mieux les éléments d'entraînement. Cela est équivalent à l'optimisation du problème quadratique ayant pour contrainte Equ.1.25 et pour équation Equ.1.26 [Bishop 2006] :

$$t_n(w^T \Phi(x_n) + b) = 1 \tag{1.25}$$

$$\arg\min_{w,b} (\frac{1}{2}\|w\|^2) \tag{1.26}$$

Il existe de nombreuses librairies informatiques pour l'optimisation de ce problème quadratique, telles que : la librairie QuadProg++ [QuadProg++ 2015] et OOQP [OOQP 2015].

Les paramètres appris w et b définissent le classifieur que l'on utilisera pour prédire la classe d'un nouvel élément.

Cas non-linéairement séparable

Les cas non-linéairement séparables nécessitent l'utilisation d'un noyau non-linaire, dans ce cas on doit utiliser la représentation duale du SVM. Pour obtenir la représentation duale du SVM, il faut reprendre le problème d'optimisation de Equ.1.26 avec les contraintes de Equ.1.25. Cette fois-ci, le problème d'optimisation est résolu en utilisant un Lagrangien avec N contraintes de résolution :

$$\mathcal{L}(w,b,a) = \frac{1}{2}\|w\|^2 - \sum_{n=1}^{N}(a_n(t_n(w^T\Phi(x_n)+b)-1)) \text{ tel que } a_n \geq 0 \quad (1.27)$$

On peut démontrer que la solution à ce problème quadratique satisfait les propriétés données dans Equ.1.28 et Equ.1.29 grâce aux conditions Karush-Kuhn-Tucker (KKT) [Bishop 2006] (appendice E).

$$a_n \geq 0 \text{ et } t_n h(x_n) - 1 \geq 0 \quad (1.28)$$

$$a_n(t_n h(x_n) - 1) = 0 \Leftrightarrow \forall n \; a_n = 0 \text{ ou } t_n h(x_n) = 1 \quad (1.29)$$

Dans le cas de la représentation duale, nous pouvons réécrire Equ.1.24 en substituant w, nous obtenons donc l'équation ci-dessous :

$$h(x) = \sum_{n=1}^{N} a_n t_n k(x, x_n) + b \quad (1.30)$$

Les x_n tels que $a_n = 0$ n'apparaissent pas dans la somme de Equ.1.30 et donc ils ne jouent aucun rôle dans la prédiction. Ce sont les éléments d'entraînement autres que les vecteurs de support, qui sont au delà de la marge maximale de l'hyperplan ($t_n h(x_n) > 1$, voir propriété 1.29). Les autres points x_n tels que $a_n > 0$ sont les vecteurs de support, en effet ils vérifient la propriété $t_n h(x_n) = 1$ car, bien sûr, ils sont positionnés sur la marge maximale de l'hyperplan. Grâce à la représentation duale, seuls les vecteurs de support sont nécessaires pour la prédiction. Ainsi, un faible nombre de vecteurs de support peut être gardé en mémoire, ce qui permet d'économiser de la mémoire virtuelle pour la phase de prédiction. Tout comme pour le cas linéairement séparable, la prédiction est effectuée en observant le signe de l'évaluation des nouvelles données x par h (Equ.1.30).

La représentation duale combinée à l'utilisation de noyaux non-linéaires permet de résoudre un plus grand nombre de problèmes, tel que rendre séparable des classes qui ne sont pas linéairement séparables comme montré dans Fig.1.17. Dans ce cas précis, on peut utiliser un noyau gaussien :

$$k(x_n, x_m) = exp(\frac{-\|x_n - x_m\|^2}{2\sigma^2}) \tag{1.31}$$

Où σ est l'écart-type.

Les cas de non-séparabilité linéaires peuvent être rencontrés notamment quand on a moins d'éléments d'entraînement que de caractéristiques pour les analyser ; cependant cela arrive rarement dans le cas de la détection de piétons. La Fig.1.17 illustre bien le problème et la manière de le résoudre : bien que le cas ne soit pas linéairement séparable dans l'espace des caractéristiques de dimension 2, le cas est linéairement séparable dans l'espace des caractéristiques implicitement définit par le noyau non-linéaire gaussien utilisé (délimitations vertes).

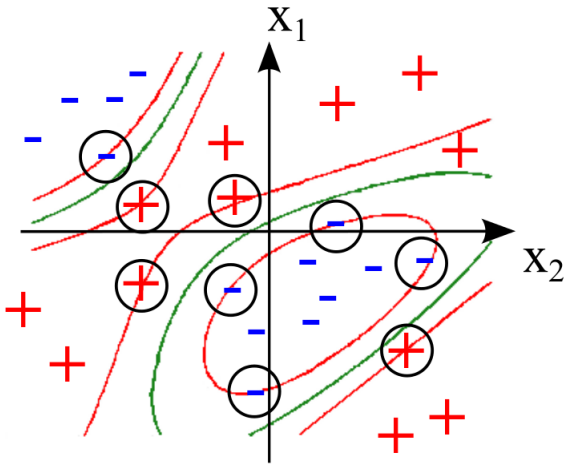


FIGURE 1.17 – Utilisation d'un noyau non-linéaire gaussien pour la séparabilité des deux classes dans un cas non linéairement séparable avec une machine à vecteurs de support.

1.4.4 Boosting

Les algorithmes d'apprentissage de type "Boosting" combinent de manière séquentielle plusieurs classifieurs à faible pouvoir discriminant entre eux pour produire un classifieur à fort pouvoir discriminant. Les classifieurs faibles sont entraînés séquentiellement en utilisant un système de pondération des éléments d'entraînement. Ainsi, à chaque fois qu'un nouveau classifieur faible est entraîné, l'entraînement est guidé pour corriger la classification des éléments d'entraînement qui sont mal classifiés par les précédents classifieurs faibles appris dans la séquence (Fig.1.18). Concrètement, le poids des éléments bien classés est abaissé (ce qui a pour effet de réduire leur influence pour le choix du prochain classifieur faible) et le poids des éléments mal classés est augmenté (ce qui a pour effet de forcer leur prise

en compte au choix du prochain classifieur faible). C'est le principe du "Boosting". Une fois que tous les classifieurs faibles ont été entraînés, leurs prédictions sont combinées dans une somme pondérée (Equ.1.36).

Parmi les algorithmes existants de Boosting, l'algorithme AdaBoost est probablement le plus connu et le plus utilisé d'entre tous, y compris dans le domaine de la détection d'objets et à fortiori, de la détection de personnes. AdaBoost a été introduit en 1996 par Freund et Shapire [Freund 1996].

La Fig.1.18.1 montre une population initiale d'éléments d'entraînement positifs et négatifs, Les Fig.1.18.2, 1.18.3 et 1.18.4 montrent la création séquentielle de trois classifieurs faibles (ainsi que la re-pondération des éléments d'entraînement avant chaque nouvelle création de classifieur faible). La Fig.1.18.4 montre le classifieur fort qui est une combinaison des trois classifieurs faibles.

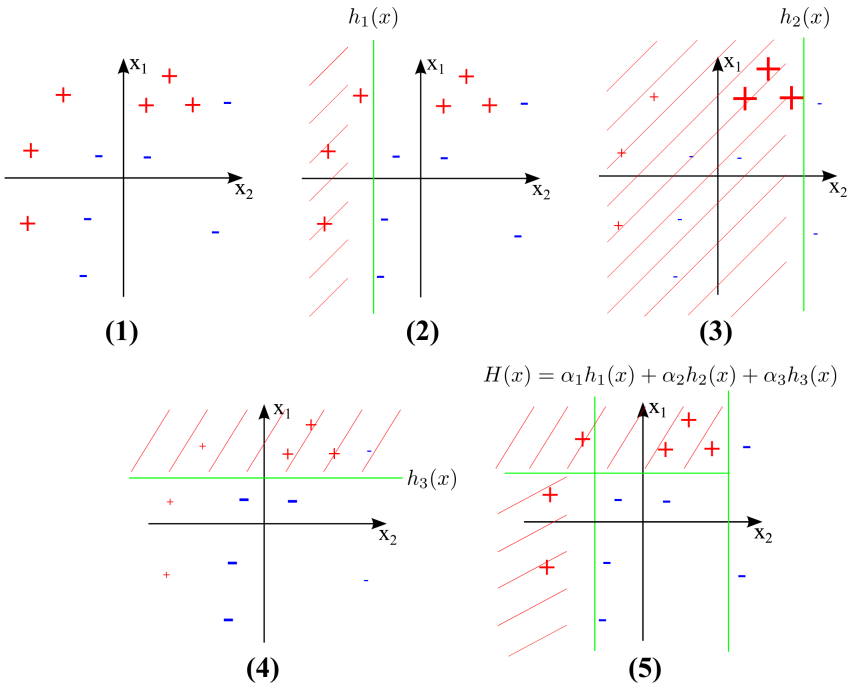


FIGURE 1.18 – Pondération des éléments d'entraînement pour la création séquentielle de classifieurs faibles avec AdaBoost.

Apprentissage avec AdaBoost

Les étapes de Alg.1 décrivent l'apprentissage d'un classifieur par l'algorithme AdaBoost. Pour chaque classifieur faible $h_m(x)$, 1 est retourné dans le cas où le classifieur faible conclut que x est un élément de la classe positive, et -1 est retourné dans le cas où le classifieur faible conclut que x est un élément de la classe négative.

Algorithm 1: Apprentissage AdaBoost

Data: Données d'entraînement

Result: Classifieur H

1 Initialiser les poids w_n des éléments d'entraînement à $\frac{1}{N}$

2 **for** $m = 0$ *to* M **do**

3 Trouver un classifieur $h_m(x)$ minimisant, pour tous les éléments d'entraînement, l'erreur pondérée suivante :

$$\varepsilon_m = \sum_{n=1}^{N} w_n exp(-\frac{1}{2} t_n \alpha_m h_m(x_n)) \tag{1.32}$$

4 Calculer α_m :

$$\alpha_m = ln(\frac{1-\varepsilon_m}{\varepsilon_m}) \tag{1.33}$$

5 Accumuler le classifieur faible à la séquence :

$$H(x) = H(x) + \alpha_m h_m(x) \tag{1.34}$$

6 Mettre à jour les poids des éléments d'entraînement en utilisant :

$$\forall n \in N, w_n = w_n exp(-\frac{1}{2} t_n \alpha_m h_m(x_n)) \tag{1.35}$$

7 **end**

Prédiction utilisant le modèle construit avec AdaBoost

L'approche générale pour prédire la classe de nouvelles données consiste à observer le signe de la séquence pondérée de classifieurs faibles, tel que décrit ci-dessous :

$$h(x) = sign(H(x)) = sign(\sum_{m=1}^{M} (\alpha_m h_m(x)) \tag{1.36}$$

La prédiction peut cependant être faite différemment. En effet, les classifieurs entraînés par des algorithmes de type "Boosting" ont un gros avantage par rapport aux autres classifieurs ; leurs structures séquentielles permettent l'utilisation d'astuces d'optimisation pour considérablement réduire les temps de calcul. On peut observer qu'il n'est pas nécessaire de continuer l'évaluation de la séquence de classifieurs faibles si, on a déjà accumulé assez d'indices pour déduire la classe des données avec quasi-certitude. Ce principe s'appelle l'évaluation en cascade. Ex-

primé d'une manière différente, la construction séquentielle des classifieurs boostés permet une analyse des données allant du plus grossier au plus fin. Ainsi, si l'analyse grossière n'est pas concluante, il n'est pas nécessaire de perdre du temps de calcul en poursuivant l'analyse. Il est à noter que l'approche en cascade est particulièrement bien adaptée dans le cas de la détection de personnes.

Viola et Jones furent les premiers à proposer une approche de classification en cascade en 2001 [Viola 2001]. Ils proposèrent de découper la classification en plusieurs étapes différentes entraînées en utilisant des sous-ensembles différents de caractéristiques visuelles (Fig.1.19).

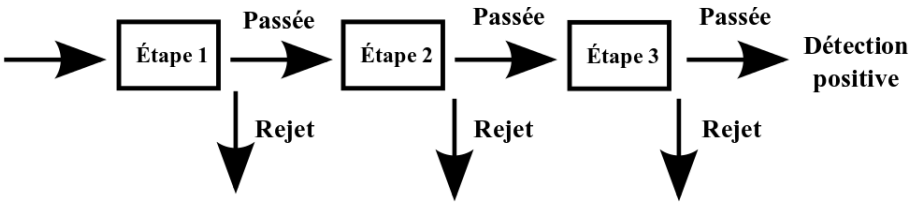


FIGURE 1.19 – Approche de classification en cascade proposée par Viola et Jones.

Avec la méthode proposée, une grande importance est donnée aux seuils de rejet à la fin de chaque étape, pour permettre un rejet progressif des éléments cas négatifs. Les auteurs ont montré que l'utilisation de la cascade permettait de considérablement réduire les temps de calculs nécessaire à la détection de visage [Viola 2001]. Avec cette approche, nous avons une détection si tous les seuils de rejet sont passés, c'est à dire, si chaque étape est passée sans que le cas évalué par le classifieur soit rejeté (Fig.1.19).

Quelques années après, une approche plus souple a été proposée pour permettre une optimisation de la phase de prédiction : il s'agit de l'approche dite "soft-cascade", proposée en 2005 par Bourdev et Brandt pour la détection de visage [Bourdev 2005]. Cette fois-ci une seule étape de classification est considérée et l'évaluation du classifieur peut être interrompu à n'importe quel moment. Cette approche se base sur l'utilisation de la somme partielle des classifieurs faibles :

$$H_m(x) = \sum_{n=1}^{m} h_n(x) \tag{1.37}$$

Celle-ci est comparée, après chaque évaluation d'un classifieur faible de la séquence, à un seuil de rejet, si la somme est en dessous, l'évaluation du classifieur est arrêtée et la classe déterminée pour le cas x est la classe négative (Alg.2).

Les M seuils de rejet forment une "trace de rejet". La sensibilité du classifieur (et donc du détecteur) est changée si la trace est translatée suivant l'axe des accumulations (Fig.1.20). La trace de rejet idéale doit rejeter le maximum possible de faux-positifs tout en laissant passer le plus possible de vrai-positifs. Le choix

Algorithm 2: Test de rejet

$H_0(x) = 0$
for $m = 0$ *jusqu'à* M **do**
    $H_m(x) = H_{m-1}(x) + \alpha_m h_m(x)$
    if $H_m(x) < seuil_m$ **then**
        On retourne -1
    end
end

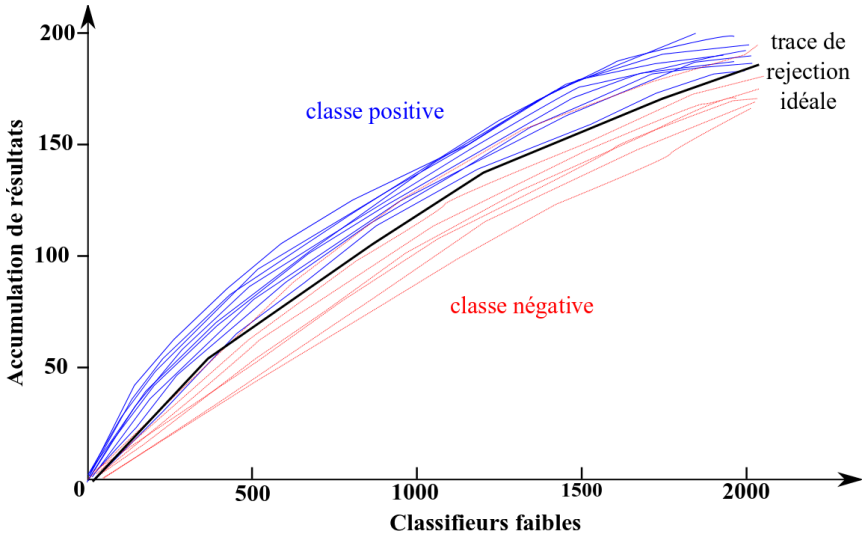


FIGURE 1.20 – Accumulation de résultats pour les éléments d'entraînement positifs (en bleu) et négatifs (rouge) et trace de rejet idéale (noire).

de la bonne trace de rejet est souvent un compromis. Cette trace est construite de telle manière qu'elle sépare au mieux les accumulations de résultats des évaluations des classifieurs faibles pour les éléments d'entraînement positifs avec celles des éléments d'entraînement négatifs.

La trace de rejet nécessite d'être construite après la phase d'apprentissage du classifieur (Alg.3). Zhang et Viola proposèrent un mécanisme de construction de la trace de rejet qui est plus facile à aborder et permet d'obtenir de meilleures traces de rejets qu'avec l'approche originale proposée par Bourdev et Brandt dans leur papier de référence sur l'approche "soft-cascade" [Bourdev 2005][Zhang 2007a]. La méthode employée est basée sur l'Élagage d'Instances Multiples ("Multiple-Instance Pruning" en anglais).

Algorithm 3: Calcul de la trace de rejet

Data: Classifieur entraîné (avec AdaBoost) contenant M classifieurs faibles, N données d'entraînement et seuil de rejet final θ_M
Result: la trace de rejet ($seuil_m$ avec $m = 1$ à M).

for $n = 1$ *to* N **do**
  if *le cas* n *est un élément d'entraînement positif* **then**
    on évalue l'élément n avec le classifieur
    On sauvegarde l'accumulation finale de résultats $S(n) = H_M(n))$
    if $S(n) \geq \theta_M$ **then**
      l'élément n est marqué comme valide
    end
  end
end
for $m = 1$ *to* M **do**
  for $n = 1$ *to* N **do**
    if *l'élément* n *est valide et est positif* **then**
      $accumulation_n = accumulation_n + \alpha_m h_m(n)$
      $seuil_m = min(seuil_m, accumulation_n)$
    end
  end
end

Choix du classifieur faible

Le classifieur faible est la brique de base du classifieur entraîné par AdaBoost. Par définition, un classifieur faible a une variance faible (peu sensible à la diversité des données d'entraînement) et donc un biais élevé (modélise avec peu de précision la séparation des classes). Le classifieur faible est couramment défini comme un classifieur ayant une erreur de classification tout juste inférieure à 50%. C'est-à-dire qui permet d'avoir une erreur de classification tout juste meilleure qu'en jouant à "pile ou face". En théorie donc, il existe un grand choix de classifieurs faibles. En pratique, on choisit souvent un classifieur faible qui est rapide à évaluer. En effet, un grand nombre de ces classifieurs de base doit être évalué pour prédire la classe finale des données d'entrée.

Parmi les classifieurs faibles les plus utilisés on peut citer : les arbres binaires de décision, les "souches" de décision ou encore les classifieurs naïfs bayésiens. Des classifieurs de conception plus complexe peuvent également être utilisés tel que le SVM (mais avec un faible pouvoir de classification), par exemple.

1. **Classifieur naïf bayésien**
   Le classifieur naïf bayésien se base sur le théorème de Bayes. Dans ce cas ci la probabilité qu'un ensemble de caractéristiques $x_i \ \forall i \in [\![1, n]\!]$ correspond à une classe C particulière ($P(C|x_1, ..., x_n)$) est simplifiée en faisant l'hypothèse qu'il y a indépendance conditionnelle entre les probabilités ($P(C|x_1)$ jusqu'à $P(C|x_n)$). C'est le caractère "naïf" du classifieur. En d'autres termes, on considère qu'il n'y a pas de liens entre les différentes caractéristiques lorsque l'on classifie. Cette hypothèse présente certains inconvénients comme certains avantages : cela simplifie l'obtention des probabilités (on ne calcule pas de co-variance) mais cela résulte en un modèle de classification trop simpliste dans certains cas. Cela en fait donc un bon candidat de classifieur faible. La classification des nouvelles données x est donnée par :

$$h(x) = \arg\max_c p(C = c) \prod_{i=1}^{n} p(X_i = x_i | C = c) \tag{1.38}$$

   L'estimation des paramètres du classifieur naïf bayésien peut être faite en utilisant une table de fréquence extraite en observant les caractéristiques d'un jeu de données d'entraînement (par exemple : le nombre d'occurrences de classes en fonction de la caractéristique x_1, etc.).

2. **Souche de décision ("decision stump" en anglais)**
   Dans le cas présent une seule caractéristique est évaluée, à l'inverse du classifieur naïf bayésien qui peut évaluer plus d'une caractéristique à la fois. Une souche de décision est en fait un simple seuil associé à une caractéristique particulière. La valeur de la caractéristique de la souche de décision est comparée à la valeur du seuil de la souche de décision (Fig1.21). La prédiction de la classe est faite suivant le signe de la comparaison ainsi que suivant la valeur du seuil. Le signe de la comparaison et le seuil sont déterminés à l'apprentissage du classifieur. Bien entendu, les souches de décision ne peuvent prédire que deux classes différentes.
   L'apprentissage d'une souche de décision nécessite plusieurs étapes : 1) trier les éléments d'entraînement (négatifs et positifs) en fonction de la valeur de la caractéristique obtenue pour chacun d'entre eux, et suivant un ordre croissant (ou décroissant) et 2) trouver la valeur seuil de la caractéristique qui sépare au mieux la population des éléments d'entraînement négatifs de la population des éléments positifs (c'est-à-dire, là où l'erreur de classification est la plus petite). L'erreur de classification est aussi fonction du signe de la comparaison.

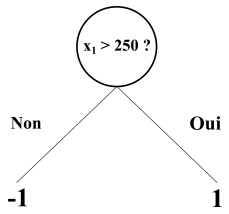


FIGURE 1.21 – Un exemple de souche de décision.

3. **Arbre binaire de décision**

   Les arbres binaires de décision sont la généralisation des souches de décision à plusieurs noeuds. L'utilisation de plusieurs noeuds permet d'augmenter le pouvoir discriminant du classifieur (Fig.1.22). Cela est effectué en modifiant la profondeur de l'arbre. Chaque noeud de l'arbre analyse une caractéristique différente, ainsi, des modèles plus complexes peuvent être appris. Les arbres binaires de décision permettent de prédire deux classes. En théorie, plus la profondeur est grande, plus l'arbre sur-ajuste les données d'entraînement (risque de sur-apprentissage), moins la profondeur est grande, meilleure est la généralisation du classifieur, mais le risque de sous-ajustement augmente en même temps. La profondeur est un paramètre important qu'il faut choisir avec attention.

   L'évaluation d'un arbre binaire de décision se fait suivant l'algorithme de parcours en profondeur : le noeud racine est évalué, puis le parcours est guidé au fur et à mesure suivant les résultats des différents noeuds, jusqu'à atteindre les feuilles de l'arbre, ce qui donne la prédiction finale :

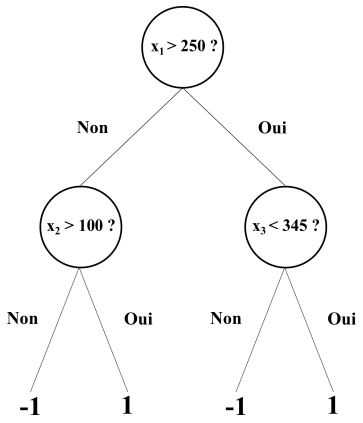


FIGURE 1.22 – Exemple d'arbre de décision binaire de profondeur 1.

L'apprentissage d'un arbre binaire de décision se fait noeud après noeud.

Une fois que le noeud racine est entraîné, les éléments d'entraînement sont scindés en deux groupes distincts : les éléments d'entraînement inférieurs au seuil, et ceux supérieures au seuil. Les éléments d'entraînement inférieurs au seuil sont utilisés pour entraîner le noeud fils gauche, et les éléments d'entraînement supérieurs au seuil sont utilisés pour entraîner le noeud fils droit. Le noeud fils gauche et le noeud fils droit sont entraînés exactement de la même manière que le noeud racine, mais sur leurs données respectives. Et ainsi de suite, jusqu'à atteindre la profondeur souhaitée.

1.4.5 Apprentissage d'un modèle déformable (LSVM)

Certaines méthodes d'apprentissage permettent de prendre en compte le caractère déformable de la classe d'objet que l'on souhaite apprendre. Ainsi, apprendre un modèle déformable LSVM permet de détecter un objet pour une multitude d'aspects, et donc pour plusieurs poses. Cette approche est intéressante, par exemple, pour la détection de personnes dans des poses variées.

Une des méthodes les plus populaires pour permettre l'apprentissage de classes d'objets à partir d'un modèle déformable est le Latent-SVM (ou LSVM). Le LSVM fut introduit en 2008 par Felzenszwalb et al [Felzenszwalb 2008]. L'approche proposée par Felzenszwalb et al a le gros avantage d'apprendre de lui-même le modèle déformable le plus adapté en fonction des données d'apprentissage qui lui sont fournies.

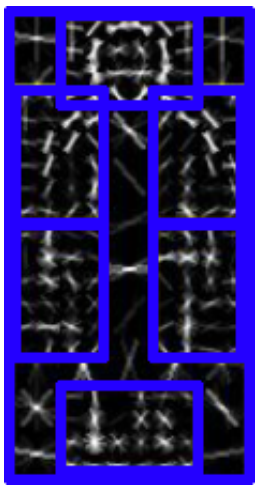


FIGURE 1.23 – Exemple de configuration de filtres obtenue à l'apprentissage d'un modèle déformable avec le LSVM.

La méthode consiste à analyser les éléments d'entraînement positifs en utilisant un filtre principal et plusieurs autres filtres rectangulaires, disposés suivant un

modèle en étoile (Fig.1.23). Le but du filtre principal est de capturer l'information grossière de l'élément d'entraînement positif. Le rôle des filtres en étoile est de capturer la diversité des placements des parties de la classe d'objet (parties de corps, par exemple).

Les données d'entraînement sont des triplets (x_i, y_i, z_i). Avec x_i : le vecteur de caractéristiques visuelles de l'élément d'entraînement i, y_i : la labélisation de l'élément (positif ou négatif) et z_i la configuration du placement de ses parties. À noter qu'ici, seules les données d'entraînement positives sont définies par une boîte englobante.

$$f_w(x) = \max_{z \in Z(x)} (w.\Phi(x, z)) \tag{1.39}$$

$$w^* = \arg\max_w \lambda \|w\|^2 + \sum_{i=1}^{n} max(0, 1 - y_i f_w(x_i)) \tag{1.40}$$

Dans Equ.1.39 : Φ est un des filtres triangulaires de l'image, w est le vecteur de paramètres, z est le vecteur de configuration du placement des filtres dans l'espace et $Z(x)$ est l'ensemble des configurations spatiales possibles des filtres pour l'élément d'entraînement x. Le score obtenu avec le modèle pour le placement des filtres z est donné par $w.\Phi(x, z)$. Apprendre le modèle déformable avec LSVM consiste à optimiser le problème donné dans Equ.1.40.

L'entraînement du LSVM tel que proposé par Felzenszwalb se fait de manière itérative, en traitant les données d'entraînement au fur et à mesure et en récoltant des données d'entraînement négatives en utilisant un système de cache [Felzenszwalb 2008]. Le LSVM produit des modèles qui sont lents à évaluer. De plus cette approche ne semble pas adaptée à la détection de personnes définies sur un faible nombre de pixels : le modèle appris est trop complexe pour être pertinent sur un grand intervalle d'échelles.

1.4.6 Réseau de neurones convolutionnels

Le réseau de neurones convolutionnels (ou "Convolutional Neural Network" en anglais, CNN [9]) est une approche hiérarchique de classification qui est essentiellement utilisée pour la détection et la reconnaissance d'objets [LeCun 1989]. Un des gros avantages de cette approche est qu'il n'est pas nécessaire de définir l'espace des caractéristiques visuelles ; les caractéristiques sont trouvées automatiquement lors du processus d'apprentissage du CNN grâce à un apprentissage hiérarchique. Les travaux récents sur l'apprentissage profond ("deep learning") ont repopularisé le CNN pour la détection d'objets. Récemment, de nombreux travaux

9. *Convolutional Neural Network*, Réseau de neurones convolutionnels

démontrèrent l'efficacité de cette approche combinée à un apprentissage profond [Krizhevsky 2012][Sermanet 2012].

Le CNN diffère du réseau de neurones classique par sa structure : les couches du réseau sont alternativement des couches de caractéristiques et des couches d'agrégation. La couche finale est obtenue grâce à une regression "soft-max" de l'avant dernière couche du réseau. Le CNN se base sur trois principes : 1) la connectivité locale des pixels (les images ont une topologie 2D, les pixels voisins partagent une information commune), 2) le partage de paramètres pour chaque connectivité locale de pixels possible (permettant la création de caractéristiques visuelles) et 3) l'agrégation des caractéristiques visuelles (par un calcul du maximum ou de la moyenne sur les couches de caractéristiques).

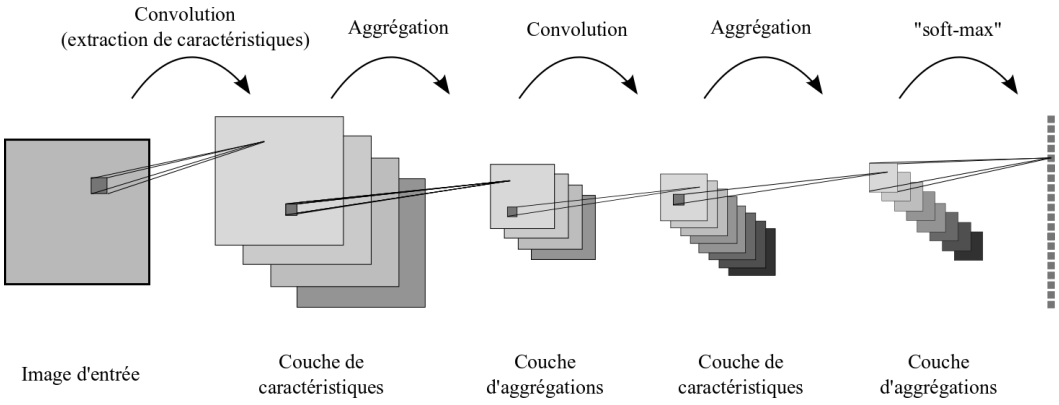


FIGURE 1.24 – Un réseau de neurones convolutionnels avec quatre couches (deux couches de caractéristiques et deux couches d'agrégations).

Pour chaque caractéristique visuelle différente il y a un noyau de convolution différent. Le noyau de convolution est la transposée d'une matrice de paramètres (de poids) qui correspond à une configuration 2D spatiale particulière permettant d'extraire une caractéristique visuelle. Ces matrices de paramètres sont obtenues soit : par génération aléatoire, soit par un processus non-supervisé (utilisation d'autoencoders, etc). Les couches de caractéristiques sont construites par opération de convolution (Fig.1.24). Les couches de caractéristiques sont ensuite sous-échantillonnées par une étape d'agrégation. Le but de cette étape est d'extraire les éléments significatifs de la couche précédente.

La structure hiérarchique du CNN (où chaque pixel d'entrée est traité séparément) introduit une lenteur d'analyse. La multiplication des couches (nécessaire au CNN profond) n'arrange pas le problème. Pour améliorer les temps de calcul, une implémentation sur GPU [10] est souvent envisagée [Krizhevsky 2012]. De plus, un CNN profond requiert un grand nombre d'images d'entraînement au risque de

10. *Graphic Processor Unit*, Unité de calcul graphique

sur-apprendre les classes. À noter que le CNN non-profond n'est pas compétitif par rapport aux autres approches d'apprentissage présentées dans cette thèse.

1.5 Recherche dans l'espace des solutions

Les méthodes présentées ici concernent l'étape de recherche de personnes dans l'espace. Il s'agit de la dernière étape de la chaîne de traitement dont le but est de rechercher le modèle humain appris précédemment à la phase d'apprentissage. La recherche s'effectue pour plusieurs positions et pour plusieurs échelles. Dans la suite de cette section nous allons présenter l'approche traditionnelle de recherche de personnes dans les images : la recherche exhaustive. Dans un second temps, nous allons également présenter la recherche métaheuristique de personnes.

1.5.1 Recherche exhaustive

La méthode de recherche exhaustive est la méthode la plus directe et la moins optimisée. Elle consiste à chercher le modèle humain en tous lieux de l'image et pour toutes les échelles. Cette recherche exhaustive est rendue possible grâce à la construction et à l'utilisation d'une pyramide d'images : la pyramide d'images est construite en générant plusieurs niveaux sous-échantillonnés et sur-échantillonnés de l'image d'entrée (Fig.1.25.1) :

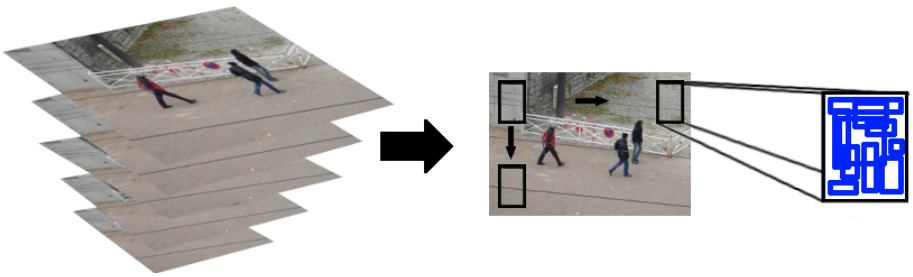


FIGURE 1.25 – Balayage exhaustif d'une image en tous lieux et pour plusieurs profondeurs à l'aide d'une pyramide d'image.

Les différentes échelles d'images permettent de chercher le modèle humain pour différentes profondeurs : la taille de la fenêtre de détection devant rester fixe, c'est la taille de l'image qui doit varier pour permettre l'analyse en profondeur. En effet, à l'apprentissage le classifieur est entraîné avec des éléments d'entraînement dont la taille est fixe. La même quantité de données doit donc être fournie au classifieur pour la phase de prédiction.

La recherche exhaustive génère très souvent un grand nombre de détections autour des scores maximums. Il est donc nécessaire de procéder à une étape de post-traitement pour ne garder qu'une seule détection par maximum de score. Cette étape est plus connue sous son appellation anglaise : "Non-Maximum Suppression" (ou NMS [11]).

Post-traitement des détections ("Non-Maximum Suppression")

Deux approches sont couramment envisagées pour procéder à la NMS : l'approche basée "Mean-Shift" [Dalal 2006] et l'approche par recouvrement [Felzenszwalb 2008]. L'approche basée "Mean-shift" se base sur une analyse de la densité des détections. Avec cette approche on considère que le lieu de densité maximum contient un bon candidat de détection. L'approche par recouvrement se base quant à elle essentiellement sur l'analyse du score des détections.

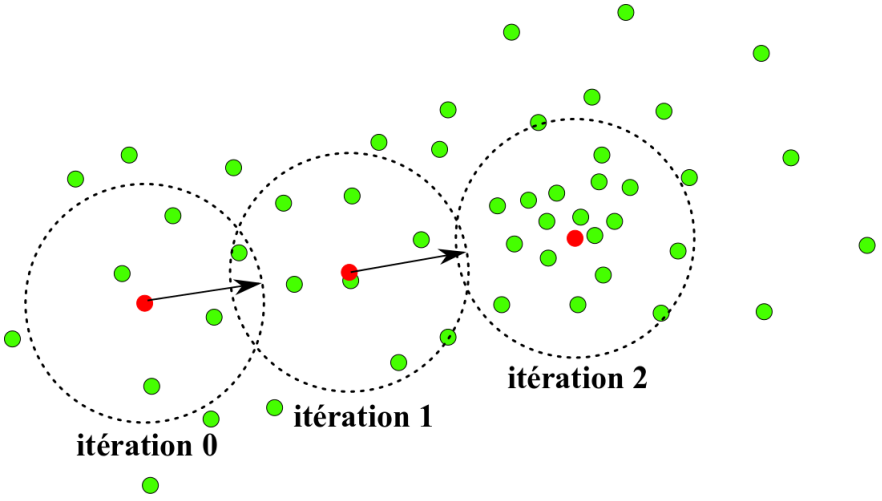


FIGURE 1.26 – Recherche itérative d'un mode dans des données discrétisées avec l'approche "Mean-Shift"

1. **Approche basée "Mean-Shift" :** L'algorithme "Mean-shift" permet de trouver les maximaux de densité (ou modes de densité) de données discrétisées. Cet algorithme est donc parfaitement adapté à la recherche du mode d'un ensemble de détections locales.
   L'algorithme procède à une analyse locale grâce à l'utilisation d'un noyau de calcul, où x est le centre du noyau et x' est un point quelconque :

$$Noyau(x', x) = exp^{-c\|x'-x\|^2} \tag{1.41}$$

11. *Non-Maximal Suppression*, Suppression des non-maximums

Le centre du noyau de calcul x est initialisé dans le voisinage du mode (noyau itération 0 dans Fig.1.26). Le centre du noyau de calcul est progressivement déplacé vers le mode après plusieurs itérations, en remplaçant x par $m(x)$:

$$m(x) = \frac{\sum_{x' \in V} Noyau(x', x)x'}{\sum_{x' \in V} Noyau(x', x)} \tag{1.42}$$

Le mode est trouvé lorsque l'algorithme arrête de converger. Le paramètre c permet de configurer le périmètre du noyau gaussien. V est le voisinage du centre x compris dans le noyau gaussien.

Le mode final correspond à la détection finale que l'on souhaite garder (point rouge, itération 2 Fig.1.26). À noter que, la taille de la détection finale est ré-ajustée, car le mode vers lequel l'algorithme a convergé correspond très rarement à une détection mais plus à une interpolation de détections (le point rouge se superpose rarement sur un point vert dans Fig.1.26).

2. **Approche par recouvrement :** L'approche par recouvrement consiste à trouver la détection qui a le meilleur score parmi les détections qui se recouvrent suffisamment. Cette approche est plus rapide que l'approche basée "Mean-Shift" mais est néanmoins moins précise. L'algorithme procède comme suit : pour chaque détection on trouve les autres détections qui la recouvrent avec un pourcentage de recouvrement suffisamment important (en pratique on pourra choisir un pourcentage de 50%). Parmi les détections qui se recouvrent on choisit celle qui a le score le plus élevée. Les autres sont supprimées.

1.5.2 Recherche métaheuristique

La recherche de personnes peut être vue comme un problème d'optimisation du score du détecteur de personnes dans l'espace de recherche (à trois dimensions). Résoudre ce problème d'optimisation permettrait de trouver les maximaux de score, et donc, les zones de l'image susceptibles de contenir des motifs humains. Il existe de nombreuses méthodes pour résoudre un problème d'optimisation, seules certaines d'entre elles peuvent être appliquées au cas de la détection de personnes.

Les méthodes d'optimisation analytiques classiques ne peuvent pas être employées dans le cas de la détection d'objet, car le détecteur ne peut pas être défini par une fonction analytique. La résolution du problème d'optimisation peut être effectuée en utilisant les méthodes métaheuristiques. Ces méthodes sont idéales pour résoudre les problèmes qui sont mathématiquement difficiles (ou impossible) à définir, et sont quelquefois plus efficace que les méthodes analytiques pour résoudre des problèmes d'optimisation analytiques. L'avantage des méthodes heuristiques

est qu'elles peuvent être adaptées à un grand nombre de cas d'optimisation différents sans changement majeur de l'algorithme. Le principe général des méthodes heuristiques est de faire progresser le score de la fonction objectif par échantillonnages successifs de celle-ci afin de se rapprocher progressivement des maximums. Plusieurs stratégies sont employées pour permettre cette progression vers les maximums. Ces stratégies sont inspirées du fonctionnement de la nature : certaines sont inspirées du comportement animal, d'autres de la biologie ou encore de la physique.

Parmi les approches les plus connues : on peut citer l'algorithme génétique (inspiré de la biologie), le recuit simulé (inspiré d'un processus physique utilisé en metallurgie) ainsi que les colonies de fourmis et les essaims de particules (tous deux inspirés du comportement animal).

Plusieurs travaux en détection d'objets utilisent les essaims de particules pour accélérer la recherche de personnes dans les images : Owechko et al proposèrent l'algorithme SNPSO [12] et Saisan et al utilisèrent les essaims de particules pour une détection de personnes utilisant plusieurs vues de la scène [Owechko 2004][Saisan 2005]. Il existe également quelques travaux se basant sur l'algorithme génétique pour la détection d'objets [Bebis 2000][Swets 1995]. L'approche génétique est cependant plus lourde à mettre en place pour le contexte de la recherche d'objets dans l'espace.

Optimisation par Essaim de Particules (PSO)

L'optimisation par essaim de particules (ou "Particle Swarm Optimization" en anglais) consiste à faire converger un essaim de particules itérativement vers un maximum global (ou minimum global). L'approche est illustrée dans Fig.1.27 :

L'idée est de faire se mouvoir à chaque itération les particules de l'essaim en fonction de la meilleure particule globale de l'essaim $\vec{g}$ et suivant la précédente meilleure position de la particule i : $\vec{b_i}$ (Equ.1.43 et Equ.1.44) [Kennedy 1995]. La particule $\vec{g}$ est tout simplement la particule de l'espace de recherche qui maximise au mieux la fonction objectif dans tout l'espace pour l'itération t. La précédente meilleure position de la particule i notée $\vec{b_i}$ est la position de la particule ayant obtenue le score le plus élevé sur le parcours de la particule.

Alg.4 décrit l'algorithme PSO [13] générique.

Les paramètres de l'algorithme qui permettent d'influer sur la vitesse de convergence sont : ω, s et c. L'inertie de la particule est contrôlée par ω, le "caractère social" de la particule est contrôlé par le coefficient s (c'est-à-dire, le degré d'influence de $\vec{g}$ dans la prédiction du futur mouvement), et le "caractère cognitif" de

12. *Sequential Nitching Particule Swarm Optimization*, Essaim de particules avec "niching" séquentiel
13. *Particule Swarm Optimization*, Essaim de particules

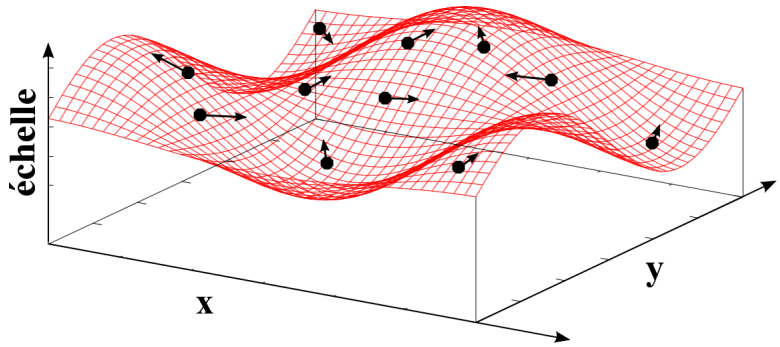


FIGURE 1.27 – Illustration de l'optimisation d'un problème par le déplacement d'un essaim de particules dans l'espace des solutions. En noir : les particules, en rouge : un maillage représentant l'espace des solutions. Les vecteurs accolés aux particules représentent le sens du déplacement des particules à la prochaine itération.

la particule est contrôlé par le coefficient c (c'est-à-dire, le degré l'influence de sa meilleure position dans la prédiction de son futur mouvement). Un caractère aléatoire est également ajouté à la prédiction du futur mouvement grâce à r, qui est une valeur aléatoire prise entre 0 et 1.

L'algorithme PSO converge théoriquement vers le maximum global de l'espace de recherche à partir d'un certain nombre d'itérations. Une convergence est observée lorsque les particules ont tendance à se regrouper à un même endroit de l'espace de recherche. Lorsque la convergence est stabilisée, une solution au problème d'optimisation est trouvée. La stabilisation d'une convergence nécessite un certain nombre d'itérations t. Il est important de choisir un nombre d'itérations maximal suffisamment important pour permettre la convergence vers une solution.

À noter que l'algorithme PSO originel n'est pas en mesure de trouver plusieurs maximums locaux de solutions dans l'espace de recherche mais seulement un. Il existe cependant un certain nombre de travaux permettant de corriger ce problème de manière séquentielle : avec le "niching" [Li 2010][Brits 2002], ou de manière dynamique, avec un processus de groupement ("clusterisation" en anglais) [Yang 2010].

L'optimisation par essaim de particules est une approche qui a déjà été utilisée avec succès dans le domaine de la détection d'objets : Owechko et al proposèrent un framework de détection basé sur cette technique d'optimisation [Owechko 2004]. Saisan et al proposèrent une approche de détection et de localisation des personnes utilisant plusieurs caméras et se basant également sur le PSO [Saisan 2005].

1. **PSO pour la détection de personnes**

Owecho et al proposèrent un dérivé de l'algorithme PSO : le SNPSO (Opti-

Algorithm 4: Optimisation par essaim de particules

Data: Espace de recherche des solutions
Result: Solution optimisant le problème

Initialiser aléatoirement les particules $\vec{p_i}$ dans l'espace de recherche
Initialiser les meilleures positions $\vec{b_i}$ tels que $\vec{b_i} = \vec{p_i}$
Initialiser à 0 les vitesses des particules $\vec{v_i}$
Évaluer la fonction objectif pour chaque position $\vec{p_i}$
for $t = 0$ *to* T **do**
    Trouver $\vec{g}$
    for $i = 0$ *to* $nombre\_maximum\_de\_particules$ **do**
        Mise à jour de la vitesse et de la position de la particule i :

$$\vec{v_i} = \omega.\vec{v_i} + c.r(\vec{b_i} - \vec{p_i}) + s.r(\vec{g} - \vec{p_i}) \tag{1.43}$$

$$\vec{p_i} = \vec{p_i} + \vec{v_i} \tag{1.44}$$

        Évaluation de la fonction objectif pour la position $\vec{p_i}$
        Mise à jour de $\vec{b_i}$ si la position courante de la particule i est meilleure
    end
end

misation par Essaim de Particules en utilisant le Niching Séquentiel, en anglais : "Sequential Niching Particle Swarm Optimization") [Owechko 2004]. Ce dérivé permet, contrairement au PSO, de trouver plusieurs maximaux locaux dans l'espace de recherche. Owecho et al appliquèrent leur algorithme à la détection de personnes dans l'imagerie infrarouge. Cependant l'approche est également utilisable dans le spectre visible en modifiant la nature du classifieur employé.

Le principe de l'algorithme se base sur l'évaluation, pour chaque emplacement de particule, des caractéristiques visuelles contenues dans une fenêtre de détection par un classifieur. Le reste de l'algorithme est semblable à l'algorithme PSO exposé ci-dessus. Pour détecter plusieurs personnes, les auteurs utilisent le principe du "niching" séquentiel. Cela consiste à supprimer de l'image une solution qui a déjà été trouvée dans le but que le maximum de score n'influt plus pour la recherche de nouvelles solutions. Le score des particules est scrutée : si celui-ci dépasse un seuil de score G_{best} le processus de "niching" est engagé.

La détection de personnes utilisant le "niching" séquentiel est effectuée comme montré dans Alg.5.

Les auteurs montrèrent que leur approche est beaucoup plus efficace que la

Algorithm 5: "Niching" séquentiel

Data: Essaim de particules, Image modifié
Result: Détections, Image

```
for i = 0 to nombre_maximum_particules do
    if Score de particule i est > score de G_best après T' itérations then
        On effectue un test de voisinage autour de la particule
        if Aucun score voisin dépasse le score G_best then
            On est face à un faux-positif
        end
        else
            On a un détection, elle est enregistrée.
        end
        On supprime localement l'image par une gaussienne.
    end
end
```

méthode de recherche exhaustive : les personnes sont détectées dans l'espace de recherche avec beaucoup moins d'appels au classifieur qu'avec l'approche exhaustive [Owechko 2004]. Un facteur d'accélération de 1000 par rapport à l'approche classique a été atteint sur leurs données de test. À noter qu'avec l'approche particulaire, il n'est pas nécessaire de procéder à une étape de fusion des doublons de détections (ou "Non-Maximum Suppression" en anglais), puisque les maximaux de résultats sont directement trouvés.

2. **PSO pour la détection de personnes sur plusieurs vues**

Saisan et al. proposèrent une approche basée sur l'algorithme PSO pour la détection de personnes et aussi pour le suivi de celles-ci [Saisan 2005]. Deux caméras sont utilisées : les particules de l'essaim sont évaluées pour les deux caméras. Les particules sont évaluées dans chacune des vues (le passage d'une vue à l'autre est fait grâce à une transformation géométrique). Le problème est vu comme étant la maximisation du résultat combiné des deux vues (fusion des résultats sur les deux vues). Le but est de réduire l'apparition de faux-positifs et de permettre la localisation de la personne dans la scène par géométrie épipolaire afin de faciliter son suivi.

1.6 Réduction de l'espace de recherche des solutions

Réduire l'espace de recherche permet de limiter le nombre de fois où la fenêtre de détection est évaluée dans l'espace des solutions. Suivant le pourcentage de

réduction de l'espace, les gains en temps de calcul peuvent être très significatifs. Réduire l'espace de recherche revient à identifier des régions d'intérêt particulières dans l'image. Ces régions sont susceptible de contenir des informations visuelles importantes.

L'extraction de régions d'intérêt nécessite une analyse bas-niveau de l'image. Dans la modalité visible, plusieurs alternatives sont envisageables pour permettre cela : la soustraction du fond (ou détection du premier plan), l'analyse du flux optique, l'analyse de la saillance visuelle de l'image ou encore l'analyse du "caractère objet".

1.6.1 Extraction de régions d'intérêt par soustraction de fond

La soustraction de fond permet d'extraire de l'image les régions de la scène qui sont en mouvements, par conséquent, il n'est pas possible de détecter des personnes statiques avec ces méthodes. L'extraction de fond est rendue possible par une analyse temporelle de la scène sur un flux vidéo. La méthode nécessite que la caméra soit statique et que le fond ne change pas (ou très peu) avec le temps. Cette méthode est mal adaptée pour les environnements extérieurs ou tout type d'environnement où il y a beaucoup de changements.

Parmi les approches de soustraction de fond nous pouvons citer : 1) les travaux de Wren et al en 1997 qui proposèrent de modéliser les pixels du fond par des distributions gausiennes pour être plus robuste aux changements d'illumination [Wren 1997], 2) les travaux de Stauffer et Grimsson qui proposèrent un modèle de mélange gaussien pour modéliser des changements plus complexes tels que l'eau qui s'écoule dans une rivière où le mouvement des feuilles dans les arbres [Stauffer 1999] et 3) Kim et al en 2005 qui proposèrent de modéliser le fond pas l'utilisation d'un "codebook" visuel [Kim 2005].

Parmi les auteurs ayant utilisé la soustraction de fond pour la détection d'objets nous pouvons citer Toth et al en 2003 et Zhou et al en 2005 qui proposèrent d'extraire des régions d'intéret par soustraction de fond avant une analyse plus fine de zones pour confirmer ou infirmer la présence des objets [Zhou 2005][Toth 2003].

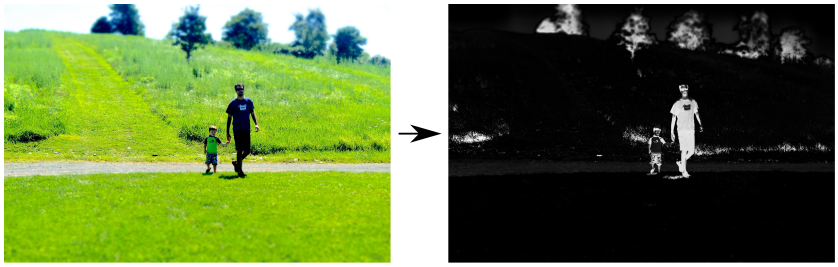
Dans le cas où la caméra est en mouvement le fond dynamique peut être soustrait en analysant le flot optique de la scène [Derome 2014].

1.6.2 Extraction de régions d'intérêt utilisant la saillance visuelle

L'étude de la saillance visuelle d'une image permet de prédire les points de fixations de l'oeil humain sur cette image. Les points de fixation de l'oeil humain sont les zones de l'image qui ont la propriété de visuellement "sortir du lot" de l'image

[Treisman 1980]. La saillance visuelle permet de reproduire les mécanismes bas-niveaux de l'attention visuelle humaine [Treisman 1980].

Par exemple, la saillance visuelle d'une personne marchant dans l'herbe sera élevée, car la personne "sort du lot" par rapport à l'environnement avoisinant (Fig.1.28).La saillance visuelle s'avère particulièrement utile pour réduire l'espace de recherche en concentrant l'effort de calcul sur les zones de l'image les plus saillantes, par exemple.



(1) (2)

FIGURE 1.28 – Carte de saillance calculée pour une image contenant des personnes.

Il existe deux types de saillances visuelles : 1) la saillance visuelle dite bas-haut ("bottom-up" en anglais), on part ici de l'information pixélique de la scène pour ensuite construire la carte de saillance, 2) la saillance visuelle dite haut-bas ("top-down" en anglais), où la saillance visuelle est guidée par une attente de haut niveau (une contextualisation de haut-niveau). La saillance visuelle bas-haut est plus simple à calculer car, généralement, seule l'information pixélique est nécessaire à son calcul. La saillance visuelle haut-bas nécessite quant à elle la prise en compte de mécanismes haut niveaux plus coûteux en temps de calcul. À noter que ces deux types de mécanismes d'attention visuelles existent chez l'être humain.

Deux approches différentes permettent de calculer la carte de saillance : l'approche biologiquement inspirée et l'approche computationnelle. Comme son nom l'indique, l'approche biologique s'inspire directement du mécanisme de l'attention visuelle humaine [Treisman 1980][Itti 1998]. L'approche computationnelle permet l'obtention de cartes de saillance en utilisant des moyens calculatoires non-biologiquement inspirés.

Ci-après sont présentés une approche de référence du calcul de la saillance biologiquement inspirée, ainsi que plusieurs approches computationnelles récentes et rapides permettant d'obtenir des cartes de saillance de bonne qualité.

Saillance biologiquement inspirée

Les travaux de Itti et al font partie des travaux fondateurs du calcul de la saillance visuelle biologiquement inspirée [Itti 1998]. Itti et al proposèrent une

méthode d'extraction des cartes de saillance se basant sur une architecture biologiquement plausible du fonctionnement bas-niveau de l'attention visuelle humaine [Itti 1998][Itti 2001]. Cette architecture a été proposée par Koch et Ullman en 1985 [Koch 1985].

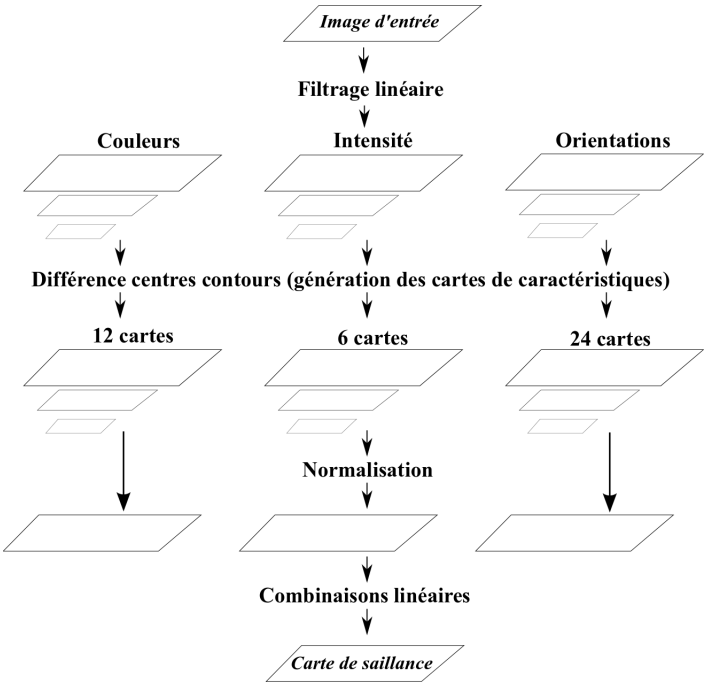


FIGURE 1.29 – Construction de la carte de saillance en utilisant le modèle biologiquement inspiré de Itti et al.

La méthode se base sur l'analyse de trois informations différentes : l'information d'intensité, de couleur et d'orientation (Fig.1.29). Mais d'autres informations peuvent également être incorporées à la méthode, comme l'information de mouvement. L'information d'intensité est simplement l'addition des trois canaux de couleurs ($I = \frac{(R+G+B)}{3}$). L'information de couleur regroupe les quatre canaux : $R'G'B'$ et Y ($R' = R - \frac{(G+B)}{2}$, $G' = G - \frac{R+B}{2}$, $B' = B - \frac{R+G}{2}$ et $Y = \frac{(R+G)}{2} - \frac{|R-G|}{2} - b$). L'information d'orientation regroupe plusieurs canaux des orientations (pour 0, 45, 90 et 135 degrés). Ces canaux d'orientations sont générés en utilisant des filtres de Gabor.

Une pyramide gaussienne avec 8 échelles est créée pour chacun des canaux (I,R',G',B',Y et les canaux des orientations O). Des cartes de caractéristiques sont ensuite générées en faisant la différence "centre moins contour" entre les échelles de niveaux paires et les échelles de niveaux impaires. Au total : 6 cartes de caractéristiques sont générées à partir de I, 12 cartes de caractéristiques sont générées à partir de R',G',B' et Y, et 24 cartes de caractéristiques sont générées à partir des canaux des orientations. Les cartes de caractéristiques sont ensuite normalisées,

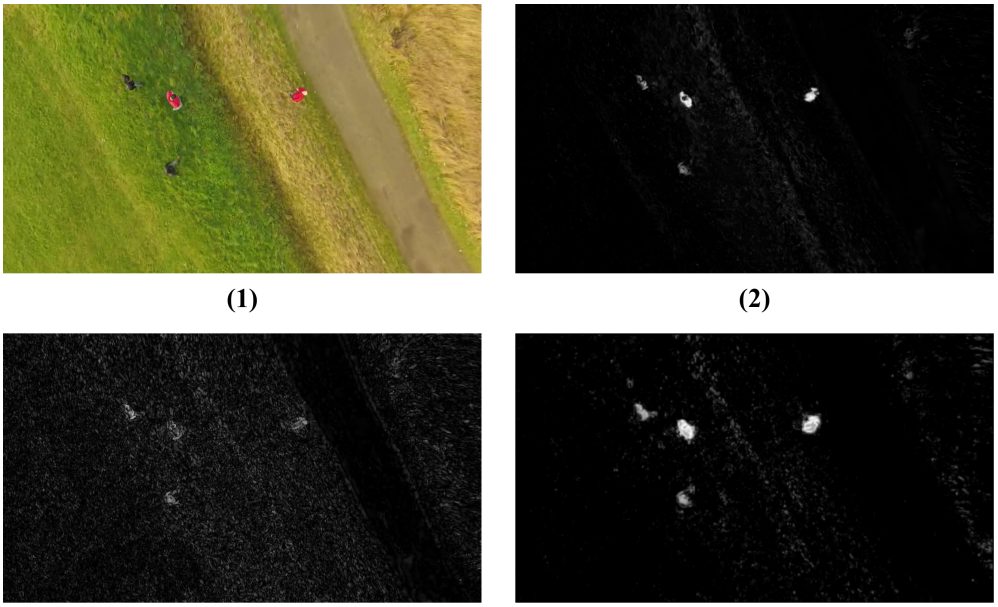


FIGURE 1.30 – Comparaison du calcul de carte de saillance de l'image (1) avec les approches computationnelles d'Achanta (2), de Katramados (3) et de Liu (4).

et elles sont combinées linéairement entre elles pour générer la carte de saillance finale [Itti 1998].

Calculer la saillance visuelle avec une approche biologiquement inspirée ne semble pas adéquat dans un contexte de détection de personnes car : les cartes de saillance obtenues avec cette approche sont généralement inférieures en taille à l'image d'entrée et la mimétisation du fonctionnement biologique est coûteux en temps de calcul, à comparer des temps de calcul obtenus avec la saillance computationnelle.

Saillance computationnelle

1. **Algorithme fréquentiel de détection des régions saillantes**

   Achanta et al. proposèrent un algorithme de détection des régions saillantes basé sur une étude fréquentielle de la couleur dans l'espace CIELAB. Les avantages majeurs de cette approche sont : la rapidité de calcul et la possibilité d'extraire des cartes de saillance en résolution entière. L'approche est basée sur l'application d'un filtre passe-bande.

   Selon les auteurs, l'algorithme d'extraction de carte de saillance idéal devrait : mettre en valeur les objets saillants les plus larges et permettre d'extraire des cartes de résolutions entières. Les objets les plus saillants sont mis

en valeur de manière uniforme en laissant passer les basses fréquences, les bordures des objets saillants sont également mis en valeur en gardant les hautes fréquences, le bruit et la texture sont rejetés en filtrant les très hautes fréquences. Un filtre passe-bande "Différence de Gaussiennes" (DoG) est utilisé pour le filtrage des fréquences :

$$G(x,y,\sigma_1) - G(x,y,\sigma_2) = \frac{1}{2\pi}(\frac{1}{\sigma_1^2}e^{\frac{-(x^2+y^2)}{2\sigma_1^2}} - \frac{1}{\sigma_2^2}e^{\frac{-(x^2+y^2)}{2\sigma_2^2}}) \tag{1.45}$$

Le filtre DoG est très satisfaisant pour détecter les changements d'intensité. La largeur de la bande passante est contrôlée par le ratio σ_1/σ_2 [Achanta 2009]. Afin d'avoir un large ratio entre σ_1 et σ_2, σ_1 est mis à l'infini (ce qui annule le premier terme). Pour enlever le bruit et la texture à très haute fréquences, les auteurs utilisent un petit noyau gaussien qui approxime plutôt bien le filtre DoG avec les paramètres σ_1 et σ_2 donnés. Ce qui est rapide à calculer.

La carte de saillance est calculée comme décrit ci-dessous :

$$S(x,y) = \|I_\mu - I_{w_{hc}(x,y)}\| \tag{1.46}$$

I_μ est le vecteur $CIELAB$ moyen de l'image, $I_{w_{hc}}$ est l'image d'entrée filtrée par un noyau gaussien.

2. **Détection de la saillance par encadrement symétrique maximum**

Cette approche est une amélioration de l'approche précédente. Elle bénéficie des mêmes avantages que l'approche précédente (temps de calcul réduit et génération de cartes de saillance de résolution entière). Les auteurs ont fait le constat que l'on n'avait pas de connaissance à priori de la taille des objets saillants de la scène, mais, que l'on pouvait faire l'hypothèse de la taille des objets saillants suivant leurs positionnements dans l'espace (en considérant que, on ne s'intéresse pas aux objets de la scène qui sont coupés aux bords de l'image). Concrètement, cela revient à monter la valeur de la fréquence de coupure basse du filtre passe-bande à proximité des bords de l'image, et à abaisser la valeur de la fréquence de coupure basse au centre de le l'image. Cela est fait en utilisant le principe de l'encadrement symétrique maximum : une fenêtre de taille variable $(h \times w)$ est utilisée pour calculer I_μ, la taille de la fenêtre est réduite aux bords de l'image, et augmentée au centre de l'image (Fig.1.31). La valeur du vecteur de couleur moyen par rapport à sa position dans l'image est changée suivant Equ.1.48. La saillance est calculée comme décrit dans Equ.1.47.

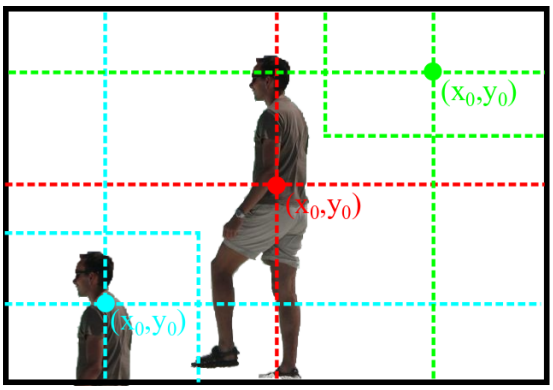


FIGURE 1.31 – Ajustement de la taille de la fenêtre utilisée pour calculer I_μ avec l'approche de calcul de saillance utilisant le principe de l'encadrement symétrique maximum.

$$S(x,y) = \|I_\mu(x,y) - I_{w_h c}(x,y)\| \tag{1.47}$$

$$I_\mu(x,y) = \frac{1}{A} \sum_{i=x-x_o}^{x+x_0} \sum_{j=y-y_o}^{y+y_o} I(i,j) \tag{1.48}$$

$$x_o = min(x, w-x) \tag{1.49}$$

$$y_o = min(y, h-y) \tag{1.50}$$

$$A = (2x_o+1)(2y_o+1) \tag{1.51}$$

3. **Calcul temps-réel de la saillance par division de gaussiennes**

   L'approche proposée par Katramados et al permet l'extraction temps-réel de cartes de saillance [Katramados 2011]. On peut extraire des cartes de saillance à partir d'un flux vidéo de résolution 640x480 à une fréquence de 27,7 Hz.

   La saillance est calculée en procédant à une division de gaussiennes (Fig.1.32.1) L'approche consiste à calculer deux pyramides gaussiennes distinctes à partir de l'image d'entrée (1). Chacune des pyramides contient 6 niveaux. La première pyramide de gaussiennes est créée en procédant à 5 étapes successives de sous-échantillonnages de construction de pyramide de gaussienne. La seconde pyramide de gaussiennes est créée en procédant à 5 étapes successives de sur-échantillonnages de construction de pyramide de gaussienne en partant de niveau le plus bas de la précédente pyramide. 2)

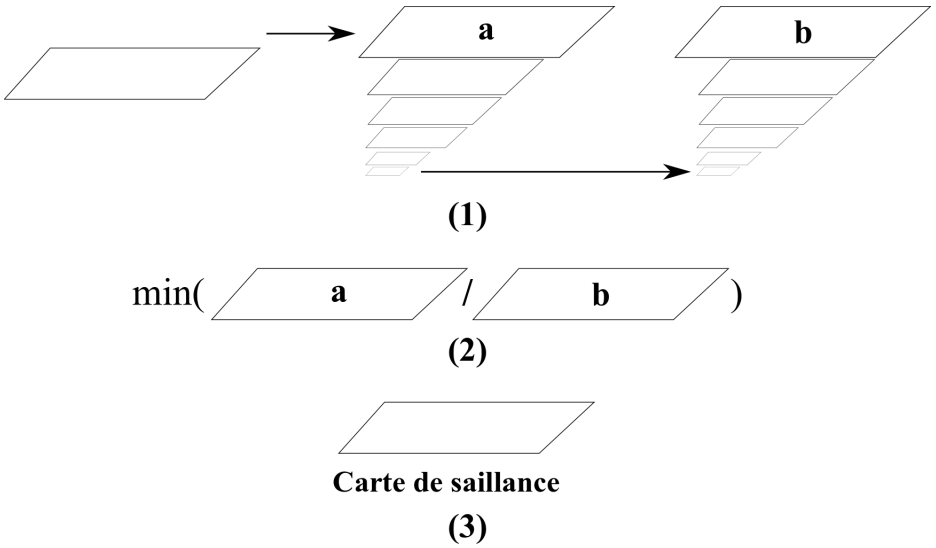


FIGURE 1.32 – Approche de calcul de la saillance par division de gaussiennes.

Les niveaux les plus haut des deux pyramides de gaussiennes sont normalisés, et les intensités de pixel sont ensuite divisées entre elles et en gardant le minimum de la division des deux intensités de pixel. 3) La carte de saillance est obtenue en normalisant le résultat de la division entre 0 et 255. 27,7 Hz pour du 640x480 couleur

4. **Modélisation de la saillance avec les histogrammes de co-occurrence**

   Cette approche est basée sur l'occurrence et la co-occurrence des valeurs d'intensité de l'image pour calculer la carte de saillance [Lu 2013]. Les avantages de cette approche sont multiples : il y a très peu de paramètres, c'est une approche robuste, et qui est tolérante aux changements d'échelle. Elle permet également de générer des cartes de saillance de taille complète (plein écran).

   Le calcul de la carte de saillance se base sur l'utilisation de plusieurs histogrammes de co-occurrence. En effet, les histogrammes à une seule dimension ne permettent que de capturer l'occurrence des valeurs de l'image. Aucune information sur la composition spatiale locale n'est disponible avec ce type d'histogramme. Les histogrammes de co-occurrences permettent quant à eux de capturer à la fois l'occurrence des valeurs de l'image ainsi que la co-occurrence locale des valeurs de l'image. Un voisinage z doit être paramétré pour jouer sur la taille de la zone d'analyse de la co-occurrence.

   L'histogramme de co-occurrence est une matrice symétrique. Les éléments de la diagonale de la matrice correspondent pour l'essentiel à l'histogramme d'occurrence classique à une dimension. Les éléments éloignés de la diagonale correspondent à la co-occurrence capturé localement. À noter que,

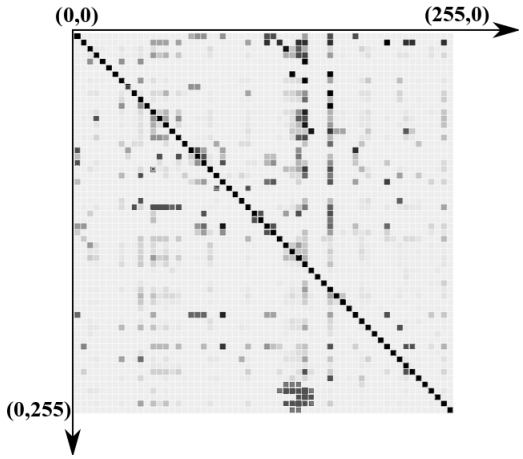


FIGURE 1.33 – Histogramme de co-occurrence de valeurs pour un canal quelconque de l'image.

lorsque l'on travaille en 8 bits, la matrice a pour dimensions : 255 colonnes et 255 lignes ($k = 255$ dans Equ.1.52).

La saillance de l'image est capturée par 1) les éléments de l'histogramme qui sont fortement éloignés de la diagonale (ils correspondent à des paires de pixels à fort contraste, telle que les contours d'un objet) ainsi que 2) les éléments de l'histogramme qui sont proche de la diagonale (correspondant à des co-occurrences de valeurs d'images qui sont rares dans l'image).

$$H = \begin{bmatrix} h(1,1) & h(1,2) & h(1,3) & ... & h(1,k) \\ h(2,1) & h(2,2) & h(2,3) & ... & h(2,k) \\ h(3,1) & h(3,2) & h(3,3) & ... & h(3,k) \\ ... & ... & ... & ... & \\ h(k,1) & h(k,2) & h(k,3) & ... & h(k,k) \end{bmatrix} \tag{1.52}$$

Une fois que les histogrammes de co-occurrences ont été calculés pour l'image, il est possible d'obtenir la valeur d'une fonction de masse p pour tous les couples (m,n) avec $(m,n) \in [0,255]^2$:

$$p(m,n) = \frac{h(m,n)}{\sum_{m'=1}^{k} \sum_{n'=1}^{k} h(m',n')} \tag{1.53}$$

U est l'inverse du nombre d'éléments $p(m,n)$ qui sont plus grands que 0 :

$$U = \frac{1}{\text{card}(\{p(m,n)\ p(m,n) > 0\})} \tag{1.54}$$

La saillance de chaque canal est calculée grâce à la relation donnée dans Equ.1.56.

$$\bar{p} = \begin{cases} 0 \text{ si } p(m,n) = 0 \\ 0 \text{ si } p(m,n) > U \\ U - p(m,n) \text{ si } p(m,n) \leq U \end{cases} \tag{1.55}$$

$$S_c(i,j) = \sum_{i'=i-z}^{i+z} \sum_{j'=j-z}^{j+z} \bar{p}(c(i,j), c(i',j')) \tag{1.56}$$

Les mêmes calculs (Equ.1.56) sont effectués pour les trois canaux de couleurs Lab et un canal spécial d'orientation. Le canal d'orientation permet de capturer l'information de gradient, qui est une donnée importante pour modéliser la saillance visuelle [Lu 2013]. La saillance finale est simplement l'addition de la saillance des quatre canaux, filtré par un filtre gaussien standard :

$$S = \mathbb{G}(S_L + S_a + S_b + S_o) \tag{1.57}$$

Dans la suite, nous nous sommes intéressés en particulier à l'approche computationnelle et notamment aux techniques proposées par Achanta et al [Achanta 2009], Lu et al [Lu 2013] et Katramados et al [Katramados 2011]. La Fig.1.30 donne un exemple de cartes de saillance calculées avec l'approche d'Achanta, de Katramados et de Liu.

1.6.3 Mesure du "caractère objet" et proposition d'objets

La mesure du "caractère objet" (ou "objectness" en anglais) a été introduit pour la première fois par Alexe et al en 2010 [Alexe 2010]. Gu et al proposèrent une approche similaire en 2009 [Gu 2009]. Depuis, d'autres types de mesures ont été proposées [Zitnick 2013][Cheng 2014]. Cette mesure permet de quantifier le "caractère objet" du contenu d'une fenêtre d'analyse. Ainsi, une mesure élevée est retournée pour une fenêtre contenant un objet (de classe quelconque) et une mesure faible est retournée pour une fenêtre contenant du fond. Il est plus rapide de calculer cette mesure pour de larges portions de l'image que de faire une analyse exhaustive de ces zones avec un détecteur de piétons ; cela permet de réduire l'espace de recherche à moindre coût.

La mesure du "caractère objet" est basée sur une analyse bas-niveau du contenu de la fenêtre d'analyse. Cette mesure permet de rapidement savoir si la fenêtre contient potentiellement de l'information visuelle intéressante à partir de caractéristiques visuelles simples. La proposition d'objets ("object proposal") se base sur cette mesure.

Il existe plusieurs approches intéressantes dans la littérature : 1) Gu el al proposèrent de grouper les segments par régions (ou boîtes) dans le but d'obtenir des propositions de d'objets [Gu 2009], 2) Alexe et al utilisent plusieurs informations différentes pour mesurer le "caractère objet" d'une boîte : la saillance, le contraste de couleur, la densité de contours, l'emplacement, la taille et le chevauchement de la boîte avec une segmentation superpixel de l'image [Alexe 2010]. (3) Cheng et al entraînèrent un classifieur linéaire sur des caractéristiques visuelles de contours. La proposition d'objets se fait en évaluant le contenu d'une fenêtre d'analyse avec le classifieur linéaire sur l'image [Cheng 2014]. (4) Zitnick et al proposèrent une mesure du "caractère objet" qui est basée sur une analyse fine des contours contenus dans la fenêtre d'analyse [Zitnick 2013]. La mesure se base sur l'affinité des contours internes avec les contours intersectants de la boîte [Zitnick 2013]. La boîte est balayée sur toute l'image.

La proposition d'objets est une approche qui est apparue récemment dans la littérature. Il existe de plus en plus de travaux sur le sujet. Cela peut être vu comme un nouveau paradigme de recherche d'objets : dans un premier temps on vient (1) rechercher des prototypes d'objets et l'on vient ensuite (2) confirmer ou infirmer la présence de l'objet par l'utilisation d'un détecteur d'objets classique.

1.7 Conclusion et perspectives

Dans ce chapitre, nous avons étudié la détection de piétons utilisée principalement pour les systèmes ADAS. Dans ce contexte d'utilisation, le système de vision est en vue piéton. Nous avons parlé des différentes approches de détection de piétons et parlé plus spécifiquement de l'approche de détection supervisée. Cette approche permet d'apprendre des modèles humains complexes et permet une détection de personnes robuste simplement à partir des pixels de l'image.

Dans un premier temps nous avons étudié différents types de caractéristiques visuelles. À la suite de cela nous avons étudié les approches d'apprentissage les plus utilisées en détection de personnes. Nous avons étudié différentes approches de recherche de personnes dans l'espace ainsi que différentes approches pour réduire l'espace de recherche pour réduire les temps de calcul.

Dans la suite de cette thèse nous reprendrons certaines des techniques étudiées dans ce chapitre pour la détection automatique de personnes à partir de drones, c'est-à-dire, pour la détection de personnes à partir d'une vue aérienne.

CHAPITRE 2

Détection de personnes en vue aérienne

2.1 Le cas aérien

2.1.1 Spécificités de la vue aérienne

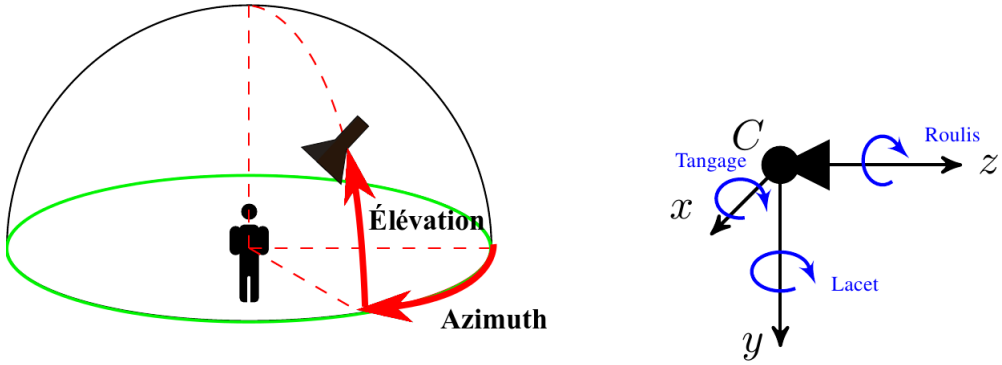


FIGURE 2.1 – Angles d'élévation et d'azimut par rapport à la personne et angles de tangage, de roulis et de lacet par rapport à la caméra. La vue aérienne est définie par la demi-sphère autour du sujet.

FIGURE 2.2 – Effets du roulis et du tangage sur des motifs humains.

Les détecteurs de personnes en vue piéton ne sont pas adaptés, par définition, à la vue aérienne (Fig.2.1.1). En effet, les détecteurs de piétons analysent les images pour trouver des motifs humains verticaux, c'est-à-dire, parallèles au plan image. Dans cette optique, une fenêtre d'analyse rectangulaire de ratio un demi permettant d'englober le motif est utilisée durant la recherche (Fig.1.25) [Dollár 2009a]. De même, à l'entraînement le classifieur du détecteur est entraîné avec des images de personnes parallèles au plan image, où l'angle de tangage (ou alors l'angle d'élévation par rapport à la personne) et l'angle de roulis de la caméra sont nuls (Fig.2.1.2).

En vue aérienne, l'angle d'élévation est non nul comme montré Fig.2.1.1. L'angle de roulis de la caméra peut également être non-nul en raison du mouvement du drone ou de perturbations extérieures ; cela complique encore plus la perception des personnes au sol de la scène.

Pour résumer, le détecteur de piétons ne peut pas être directement transposé dans le cas aérien en raison des variations extrêmes des angles de roulis et de tangage, propre à la vue aérienne.

2.1.2 État de l'art des détecteurs en vue aérienne

Il existe relativement peu de travaux traitant de la détection de personnes à partir de drones, contrairement à la détection de personnes en vue piéton. Dans cette partie, cinq résultats majeurs en détection de personnes en vue aérienne sont présentés.

(1) **(2)**

FIGURE 2.3 – Modèle humain de classification simpliste et modèle humain de classification complexe

En 2008, Rudol et al utilisèrent conjointement une caméra infrarouge et une caméra visible pour la détection [Rudol 2008]. Les régions de l'image les plus chaudes sont identifiées et celles ne correspondant pas à une ellipse sont rejetées. Les régions retenues sont ensuite analysées dans le spectre visible par un détecteur de personnes utilisant des caractéristiques Pseudo-Haar relâchées. Avec cette technique, aussi bien les personnes debout que assises peuvent être détectées.

En 2011, Gaszczak et al utilisèrent également une caméra infrarouge et une caméra visible [Gaszczak 2011]. Leur approche consiste à fusionner des caractéristiques calculées dans le spectre visible et des caractéristiques calculées dans le spectre infrarouge dans le but d'améliorer le niveau de confiance des détections. La caméra infrarouge est utilisée pour extraire des caractéristiques Pseudo-Haar et la caméra visible est utilisée pour extraire des contours afin de confirmer ou d'infirmer la détection. Leur approche permet de détecter des personnes en temps réel. Le système de vision est incliné à 45 degrés vers le bas.

Les modèle humains utilisés dans l'approche de détection de Rudol et al et Gaszczak et al sont trop simplistes pour pouvoir détecter aussi bien des personnes définies sur un faible nombre de pixels que des personnes définies sur un grand nombre de pixels (Fig.2.3). L'utilisation de ces approches est très fortement contrainte.

En 2012, Reilly et al proposèrent une approche de détection de personnes basée sur l'ombre portée des personnes [Reilly 2010]. La méthode utilise un ensemble de contraintes géométriques fournies par l'utilisateur. Dans un premier temps, la normale au plan sol est obtenue par analyse de la scène. On obtient également l'orientation des ombres portées des êtres humains ainsi que le rapport entre la taille des ombres portées et la hauteur des personnes : cela permet d'extraire des régions d'intérêt. Ces régions sont ensuite classifiées en utilisant une combinaison simple d'ondelettes de Haar et un classifieur SVM. Les travaux de Reilly et al font une hypothèse forte sur les conditions d'éclairement [Reilly 2010] ; cela implique une forte dépendance aux conditions climatiques. Cela a pour conséquence de limiter fortement l'utilisation de son approche.

Andriluka et al évaluèrent différentes techniques pour détecter des personnes gisant au sol à partir d'un drone volant à altitude très basse [Andriluka 2010]. Ils montrèrent que les détecteurs de personnes par parties, tels que les détecteurs de personnes entraînés avec un modèle déformable (LSVM), sont bien adaptés ; en effet, les détecteurs de personnes par parties prennent naturellement en compte l'articulation du corps humain. Dans leurs travaux, plusieurs approches de détection sont utilisées pour améliorer les performances. Les données de la centrale inertielle sont également utilisées pour affiner les paramètres de détection. Cependant, le modèle humain utilisé avec l'approche d'Andriluka est trop complexe pour être utilisé aussi bien pour des personnes définies sur un grand nombre de pixels que pour des personnes définies sur un faible nombre de pixels (Fig.2.3). Tout comme les approches de Rudol et al et Gaszczak et al cela restreint considérablement l'utilisation de son approche. De plus, les temps de calcul de l'approche proposée par Andriluka ne permettent pas d'envisager une détection de personnes en temps réel [Andriluka 2010].

Portmann et al proposèrent en 2014 une approche de détection par parties dans le spectre infrarouge [Portmann 2014]. Les personnes sont détectées suivant un modèle constitué de trois parties : la tête, le torse et les pieds. Un balayage exhaustif est effectuée sur toute l'image pour trouver des candidats de têtes. Pour chaque candidat de tête, une analyse plus fine est effectuée autour de celle-ci pour rechercher un modèle de torse. Si un modèle de torse est trouvé : une analyse est effectuée autour du torse pour trouver des pieds. Une personne est détectée si toutes ces parties de corps sont trouvées. Des caractéristiques Pseudo-Haar sont calculées pour trouver chaque partie. Chaque partie est entraînée par AdaBoost. L'approche proposée par Portmann et al ne fonctionne qu'exclusivement dans le spectre infrarouge et n'est pas robuste à des angles extrêmes de roulis et de tangage du système de vision.

2.2 Adaptation de la détection de piétons au cas aérien

2.2.1 Contraintes

Un certain nombre de contraintes doit être pris en compte lors de la conception de notre détecteur de personnes en vue drone. De plus, des contraintes supplémentaires sont inhérentes au projet SEARCH. Un rappel détaillé des contraintes est donné ci-dessous :

1. **Détection robuste aux changements d'orientation de la caméra :** L'inclinaison de la caméra vers le bas, les changements brutaux de caps ainsi que les réorientations de la caméra impliquent des changements importants sur les images : les motifs humains de la scène changent d'apparence et peuvent devenir indétectables pour les approches classiques (Fig.2.2). Le roulis a tendance à "tourner" les motifs humains et le tangage a tendance à "tasser" les motifs humains. Il est nécessaire que l'approche de détection soit robuste aux effets combinés du roulis et du tangage.
2. **Temps de calcul compatible temps réel :** La fréquence des détections doit être proche du temps réel afin de ne pas rater de personnes pendant les campagnes de vols. Dans cette thèse, nous définissons le temps réel comme le temps d'exécution minimum permettant d'avoir suffisamment de réactivité pour détecter toutes les personnes au sol qui sont survolées. Ce temps dépend de la vitesse de vol du drone, du champ de la caméra et de l'altitude du drone. À noter que chaque drone de la flottille doit être en mesure de détecter les personnes ; chaque drone est équipé d'une unité de calcul.
3. **Possibilité de détecter des personnes pour un grand nombre d'échelles :** Les personnes à détecter peuvent êtres proches ou éloignées du système de

vision. Il est donc nécessaire de pouvoir détecter des personnes pour un grand nombre d'échelles ; ceci est important pour "rater" le moins de personnes possible lors de l'analyse des images après acquisition.

4. **Robustesse aux changements de luminosité :** Les conditions climatiques peuvent être changeantes, le détecteur doit pouvoir continuer de fonctionner dans tous les cas : lorsqu'il fait plein soleil comme lorsqu'il pleut. Ceci est particulièrement vrai dans le contexte du projet SEARCH.
5. **Détection de personnes mobiles et immobiles :** L'approche de détection doit être en mesure de détecter aussi bien les personnes mobiles que les personnes immobiles.
6. **Respect des contraintes de charge utile et d'utilisation :** Le dispositif de détection sera fixé rigidement à la structure du drone qui pourra être de type voilure fixe ou voilure tournante. La recherche du temps de vol maximum implique la minimisation de la charge embarquée et l'exclusion de dispositifs additionnels de stabilisation mécanique des caméras qui induisent à la fois surcharge et fragilité.

La plupart des détecteurs de piétons de l'état de l'art : sont en mesure de détecter des personnes pour un grand nombre d'échelles, sont robustes aux changements de luminosité et peuvent détecter des personnes mobiles et immobiles [Dollár 2009a][Dalal 2005][Felzenszwalb 2005][Dollár 2010]. Il parait donc intéressant de partir de ce type de détecteur pour mener à bien la détection aérienne.

2.2.2 De la détection en vue piéton à la détection aérienne

Dans un premier temps le but est de montrer qu'il est possible d'adapter facilement un détecteur de piétons de conception classique au cas aérien. Nous avons donc choisi le détecteur HOG/SVM qui est un détecteur de personnes très populaire [Dalal 2005]. Plusieurs points d'améliorations sont envisagés : 1) un choix optimal des données d'apprentissage, 2) une réduction des temps de calcul.

2.2.2.1 Adapter les données d'apprentissage à la vue aérienne

Les données d'entraînement des détecteurs de piétons sont adaptées à un unique cas d'utilisation : le cas piéton, où les angles de roulis et de tangage de la caméra sont nuls. Un classifieur entraîné avec de telles données ne sera pas assez flexible aux changements de roulis et de tangage. En effet, dans le cas aérien, le roulis et le tangage déforment l'apparence des personnes (Fig.2.2).

Dans cette section, nous étudions les conséquences de l'utilisation de données d'entraînement multi-élévations avec des images de synthèses dans un premier

temps, puis, avec des images réalistes dans un second temps. L'étude est réalisée sur le détecteur HOG/ SVM.

Utilisation d'images synthétiques

Nous avons entraîné plusieurs détecteurs : un détecteur entraîné avec la base de données d'apprentissage piéton INRIA, un détecteur entraîné avec la base de données multi-élevations GMVST [1] ("Generalized Multi-view Synthetic Training", Fig.2.4) [GMVST1 2015], ainsi que trois autres détecteurs entraînés chacun sur une portion différente de la base de données GMVST2 [2] (Fig.2.5) [GMVST2 2015].

FIGURE 2.4 – Exemples d'images d'entraînement de la base de données GMVST. La base de données contient 3600 images d'entraînement positives et 14400 images d'entraînement négatives de 64x128 pixels générées pour plusieurs angles d'élévation. Les images ont été générées avec PovRay, Blender et MakeHuman.

FIGURE 2.5 – Exemples d'images d'entraînement de la base de données GMVST2. La base de données contient trois sous ensembles d'images : des images d'entraînement de 64x64 pixels (3040 positives et 32000 négatives), des images d'entraînement de 64x112 pixels (3940 positives et 20520 négatives) et des images d'entraînement de 64x128 pixels (1520 positives et 12000 négatives). Les images ont été générées avec PovRay, Blender et MakeHuman.

Dans le contexte aérien, notre but est de comparer les performances des deux approches multi-élévations avec l'approche piéton (c'est-à-dire, entraîné avec la base de données INRIA). La première approche multi-élévations (entraîné avec GMVST) balaye l'espace de recherche avec une fenêtre d'analyse de taille 64x128, quel que soit l'angle d'élévation. La deuxième approche multi-élévations (entraîné avec GMVST2) adapte la taille de la fenêtre de détection en fonction de l'angle

1. *Generalized Multi-View Synthetic Training*, Base de données pour l'entraînement multi-vues en image de synthèse
2. *Generalized Multi-View Synthetic Training 2*, Base de données pour l'entraînement multi-vues en image de synthèse

d'élévation ; un détecteur différent est utilisé pour chaque taille de fenêtre d'analyse.

Nous avons évalué les performances des trois approches en utilisant la base de données de test en vue aérienne SyntheticAerialTest1 [3] (Fig.2.6) [SyntheticAerialTest1 2015].

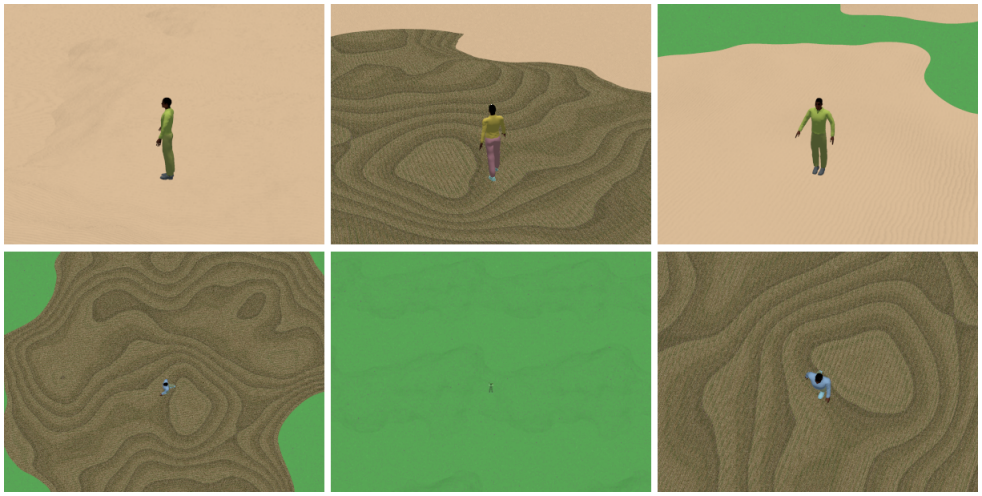


FIGURE 2.6 – Exemples d'images de la base de données SyntheticAerialTest1. La base de données contient 1440 images de test avec angles de vue complexes. Les images ont été générées avec PovRay, Blender et MakeHuman.

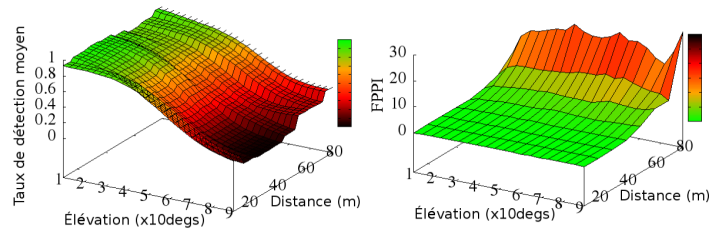
Cette base de données contient des vues de personnes pour différents angles d'élévation. Nous utilisons le critère de PASCAL [4] pour déterminer les faux-positifs et les vrais-positifs.

La base de données SyntheticAerialTest1 contient des images avec des modèles synthétiques humains générés pour plusieurs distances entre 10 et 80m par rapport à la caméra virtuelle après calibration métrique [SyntheticAerialTest1 2015].

Pour le détecteur entraîné avec la base de données INRIA, on constate que le taux moyen de détection chute considérablement à partir de 50 degrés d'élévation quelle que soit la distance (Fig.2.7.1). Le taux est nul pour un angle d'élévation plus grand que 80 degrés. Le nombre de faux-positifs par image (FPPI [5]) augmente également uniformément avec la distance pour atteindre entre 20 et 30 FPPI pour une distance de 80m (Fig.2.7.2).

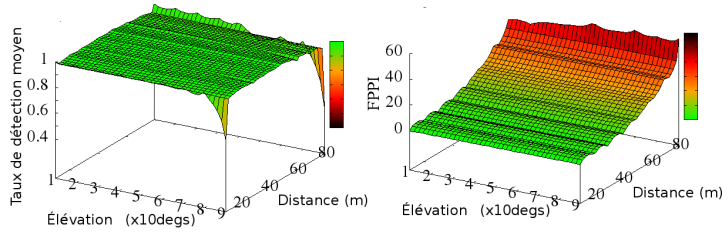
Pour le détecteur entraîné GMVST, on constate que le taux de détection moyen est presque le même quel que soit l'angle d'élévation et quelle que soit la distance

3. *SyntheticAerialTest1*, Base de données de test contenant des images aériennes de synthèse
4. *Pattern Analysis, Statistical Modelling and Computational Learning.*, Analyse de formes, modélisation statistiques et apprentissage
5. *Faux-Positifs Par Image*, Taux de faux-positifs par images



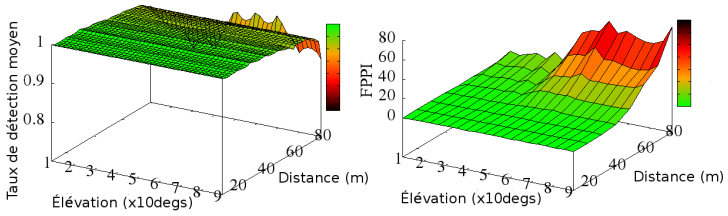
(1) (2)

FIGURE 2.7 – Taux moyen de détection et FPPI avec un détecteur entraîné INRIA sur la base de données de test SyntheseAerialTest1.



(1) (2)

FIGURE 2.8 – Taux moyen de détection et FPPI avec un détecteur entraîné GMVST sur la base de données de test SyntheseAerialTest1.



(1) (2)

FIGURE 2.9 – Taux moyen de détection et FPPI avec les détecteurs entraînés GMVST2 sur la base de données de test SyntheseAerialTest1.

(Fig.2.8.1). Le nombre de FPPI augmente plus fortement en fonction de la distance qu'avec le détecteur entraîné INRIA (Fig.2.8.2). Le FPPI est plus important avec cet entraînement car la base de données GMVST est moins variée que la base de données INRIA (qui est très riche).

Pour les détecteurs entraînés GMVST2, le taux de détection moyen est presque le même (Fig.2.9.1). Il chute légèrement pour des distances supérieures à 70m. Néanmoins, les personnes sont détectées quel que soit l'angle d'élévation et la distance. Le nombre de FPPI augmente considérablement avec l'angle d'élévation et la distance, car on analyse des surfaces de plus en plus petites (Fig.2.9). En

effet, moins la surface est grande, moins on a d'information pour la classification. Fig.2.10 et Fig.2.11 illustrent des résultats obtenus après l'entraînement GMVST et l'entraînement GMVST2.

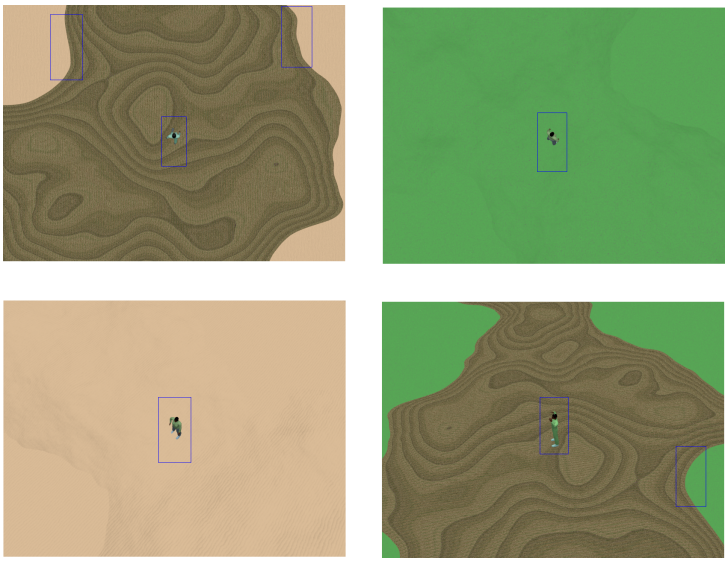


FIGURE 2.10 – Exemples de résultats obtenus sur la base de données de test SyntheticAerialTest1 avec un détecteur HOG / SVM entraîné avec la base de données GMVST.

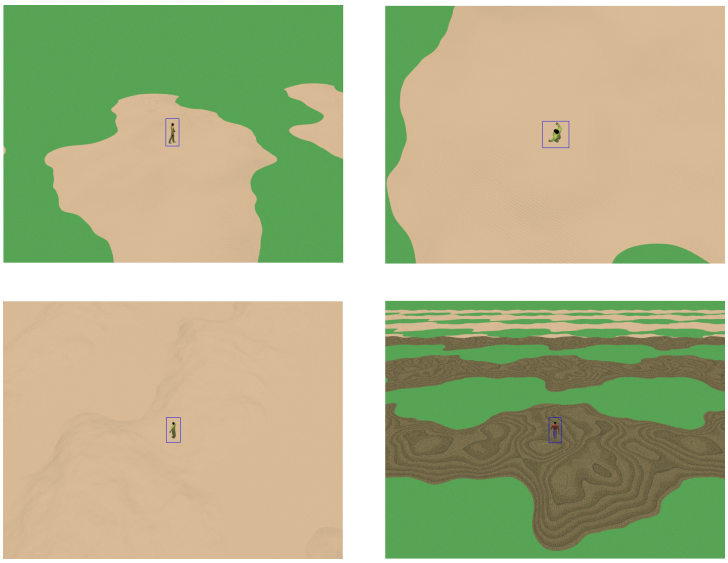


FIGURE 2.11 – Exemples de résultats obtenus sur la base de données de test SyntheticAerialTest1 avec trois détecteurs HOG / SVM entraînés avec la base de données GMVST2.

Utilisation d'images réalistes

Dans un premier temps, nous avons réalisé un test pour comparer graduellement les réponses du détecteur entraîné INRIA et du détecteur entraîné GMVRT1 [6]1 (Fig.2.12) [GMVRT1 2015].

FIGURE 2.12 – Exemples d'images d'entraînement positives de la base de données GMVRT1. La base de données contient 4222 images d'entraînement positives et 8460 images d'entraînement négatives de taille 64x128 pixels.

Nous comparons les réponses pour des angles d'élévation allant de 0 à 90 degrés. Le but de ce test a été de savoir à partir de quel angle d'élévation le détecteur entraîné INRIA commence à faillir et produire des résultats moins bons. Pour les besoins du test, 180 images de personnes prises avec des angles d'élévation entre 0 et 90 degrés ont été utilisées [ElevationTest 2015]. Quelques exemples sont donnés Fig.2.13.

FIGURE 2.13 – Exemples d'images utilisées pour évaluer la robustesse des détecteurs à l'angle d'élévation.

Dans un second temps, nous avons évalué les performances globales du

6. *Generalized Multi-View Realisitic Training dataset 1*, Base de données d'entraînement contenant des images réalistes multi-élévations

détecteur entraîné GMVRT11 en utilisant la base de données AerialTest1 [7] [AerialTest1 2015]. Le but de ce test est de confirmer la pertinence de l'entraînement multi-élévations dans le cas de la détection de personnes en vue aérienne.

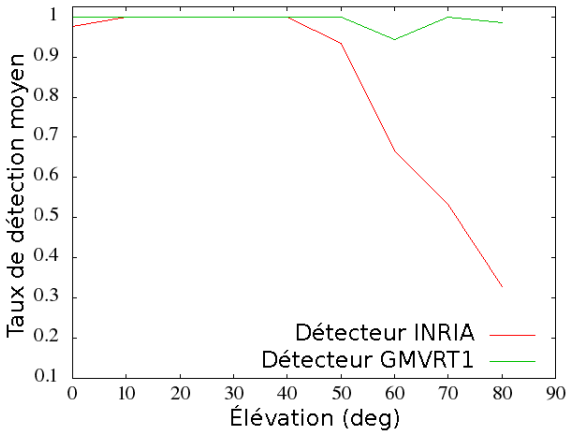


FIGURE 2.14 – Comparaison de la réponse du détecteur entraîné INRIA avec celle du détecteur entraîné GMVRT1 pour différents angles d'élevation.

Le taux de détection moyen pour le détecteur entraîné INRIA commence à décroire à partir de 40 degrés d'élevation et plus (Fig.2.14). Le taux de détection moyen pour le détecteur entraîné GMVRT11 n'est pas sensible à l'angle d'élévation. Le détecteur entraîné GMVRT11 est donc plus robuste aux changements d'angle d'élévation.

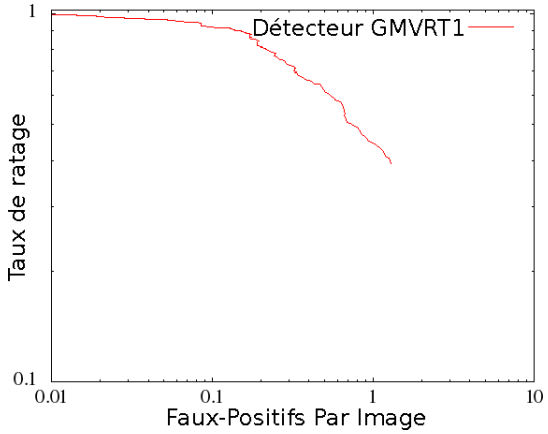


FIGURE 2.15 – Performances globales du détecteur entraîné GMVRT1 testé sur la base de données AerialTest1.

Le comportement du détecteur entraîné GMVRT11 testé sur AerialTest1 est très similaire au comportement d'un détecteur piéton testé sur une base de données

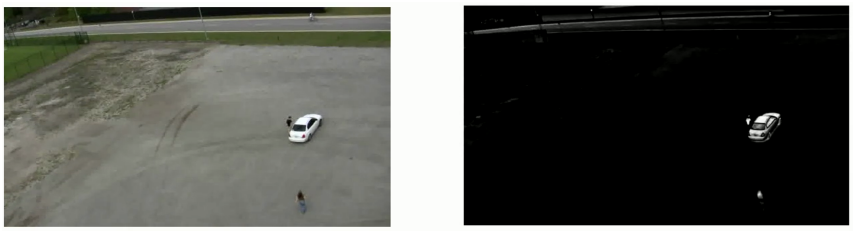
7. *AerialTest1*, Base de données de test contenant des images aériennes

piétons [Dollár 2012] (Fig.2.15). Sur la même base de données de test, le détecteur entraîné INRIA n'obtient jamais un meilleur taux de ratage que 0.9. Cela nous confirme qu'utiliser une base de données multi-élevations améliore les résultats de détection dans le cas aérien.

2.2.2.2 Réduction de l'espace de recherche des solutions

Dans cette partie, pour réduire les temps de calcul, nous proposons de réduire l'espace de recherche des solutions à explorer.

À partir d'un drone, l'intervalle d'échelles de personnes possible dans les images est très grand : suivant que le drone vol à basse altitude, ou à plus haute altitude et suivant comment est inclinée la caméra, les cas sont nombreux. Concrètement, il est nécessaire d'avoir une pyramide d'images avec un grand nombre de niveaux pour être en mesure de détecter dans tous les cas de figures (Fig.1.25). Or, plus une pyramide d'images a de niveaux (et plus particulièrement des niveaux sur-échantillonnés), plus les temps de calcul seront importants, et moins l'objectif temps réel sera atteignable. Pour réduire l'impact de la taille de la pyramide sur les temps de calcul, des optimisations sont possibles. Comme nous l'avons vu dans le chapitre précédant on peut par exemple : calculer les caractéristiques visuelles uniquement pour un nombre restreint de niveaux et faire une approximation des caractéristiques entre ces niveaux. Cependant, la recherche est toujours effectuée de manière exhaustive sur tous les niveaux. Nous avons également vu dans le chapitre précédent que l'espace de recherche peut être considérablement réduit par une analyse bas-niveau de la scène, par extraction de carte de saillances.



(1) **(2)**

FIGURE 2.16 – Extraction de carte de saillance en vue aérienne pour un milieu ouvert.

L'utilisation de la saillance visuelle est particulièrement bien adaptée en environnement naturel et en milieu ouvert (Fig.2.16). Tout objet ou personne présent dans ce type d'environnement aura une valeur locale de saillance élevée en raison de son caractère proéminent. Utiliser la saillance visuelle présente un avantage important : l'extraction des zones d'intérêts ne nécessite que la modalité visible et aucune information de mouvement n'est nécessaire (à noter que l'information de

mouvement ne permettrait pas, de toute manière, de mettre en relief les personnes qui ne sont pas en mouvement dans la scène).

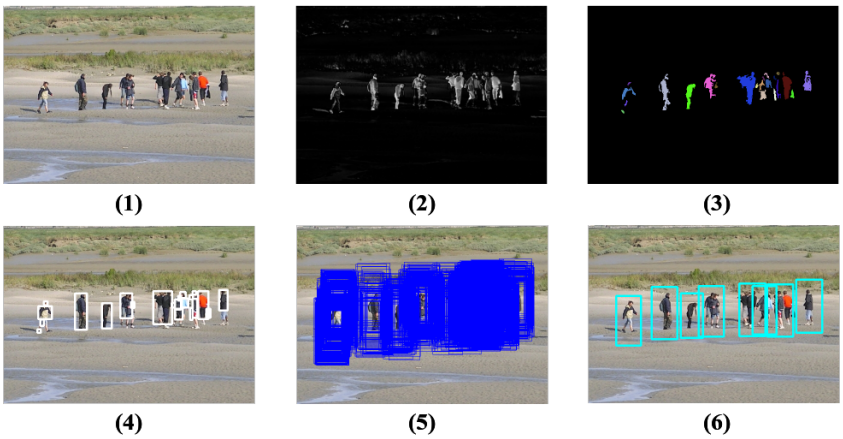


FIGURE 2.17 – Les étapes de la chaîne de traitement "saillance" pour l'analyse de personnes sur les régions saillantes.

Nous avons conçu une chaîne de traitement de l'image utilisant la carte de saillance pour réduire l'espace de recherche (Fig.2.17).

Dans un premier temps, nous extrayons la carte de saillance de l'image d'entrée (Fig.2.17.2). Nous seuillons la carte, puis nous binéarisons la carte seuillée, ensuite nous ne gardons que les amas de pixels de tailles suffisantes (Fig.2.17.3). Les boites englobantes autour des amas sont calculées (Fig.2.17.4). Nous générons des fenêtres d'analyses pour chaque amas ; les centres des fenêtres sont choisis aléatoirement à l'intérieur de chaque boite englobante (Fig.2.17.5). Et pour finir nous traitons toutes les fenêtres d'analyses séparément et fusionnons les résultats à l'aide de l'algorithme "Mean-Shift" (Fig.2.17.6).

Plusieurs paramètres doivent être considérés : la valeur de seuil, le nombre minimum de pixels agglutinés pour garder un amas, le nombre de fenêtres d'analyses à générer par amas et un intervalle de tailles nécessaire à la génération des fenêtres. La valeur de seuil peut être déterminée expérimentalement. Le nombre de pixels minimum pour garder un amas est un paramètre de filtrage. Plus on génère de fenêtres d'analyses par amas, plus cela améliore les résultats de détection mais cela prendra également plus de temps de calcul.

Dans un premier temps nous avons effectué des tests pour choisir l'algorithme de calcul de carte de saillance le plus intéressant. Dans un second temps les performances de la chaîne de traitement sont comparées à celle de l'approche classique.

Choix de l'algorithme de calcul de saillance

Nous avons évalué les capacités de réduction de l'espace de recherche ainsi que le temps de calcul moyen par image nécessaire au traitement. Trois algorithmes ont été testés : l'algorithme d'Achanta [Achanta 2010], de Lu [Lu 2013] et celui de Katramados [Katramados 2011]. Ces trois algorithmes calculent la saillance de manière computationnelle, c'est-à-dire, de manière non-biologiquement inspirée. Nous avons choisi de ne pas utiliser l'information de gradient avec l'algorithme de Lu afin d'avoir un comportement similaire aux autres algorithmes [Lu 2013].

FIGURE 2.18 – Exemples d'images de la base de données de test aérienne AerialTest1. AerialTest1 est constituée de 211 images de test annotées et de résolution 1280x720. Chaque image contient de une à plusieurs personnes prises pour des angles de vue complexes.

Nous avons effectué les tests sur notre base de données de test AerialTest1 [AerialTest1 2015]. Un ratio de réduction espace de recherche finale sur espace de recherche initiale est calculé pour chaque image de la base de données de test. Quatre mesures sont obtenues pour chaque algorithme : le ratio de réduction minimum trouvé (min), le ratio maximum trouvé (max), l'écart type (σ) des ratios de réduction, la moyenne des ratios de réductions (M) et le temps moyen (T) pour chaque image de la base (Tab.2.1).

algorithme	min (%)	max (%)	M (%)	σ (%)	T (s)
Achanta	0.0018	0.1857	0.0178	0.0227	1.34
Lu	0.0064	0.1625	0.0322	0.0339	4.02
Katramados	0.0027	0.6586	0.0401	0.0560	0.08

TABLE 2.1 – Comparaison des ratios de réduction d'espace

Nous avons paramétré le seuil de binéarisation ainsi que le nombre minimal de pixels (pour filtrer en fonction de la taille des régions) en fonction de la base de données de test ; le but étant d'obtenir la saillance la plus élevée sur les personnes tout en ayant le moins possible de bruit environnant. Les paramètres qui ont été choisis

sont : un seuil de binéarisation de la saillance de 0.18 et un nombre minimum de pixels de 30 pour l'algorithme d'Achanta, un seuil de binéarisation de la saillance de 0.20 et un nombre minimum de pixels de 50 pour l'algorithme de Katramados et un seuil de 0.25 et un nombre minimal de pixels de 50 pour l'algorithme de Lu.

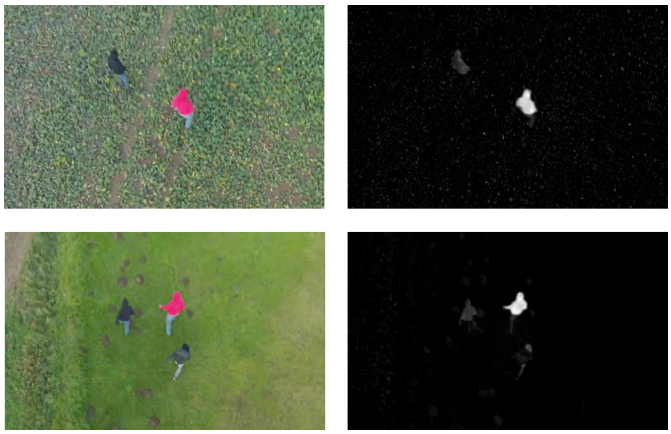


FIGURE 2.19 – Exemples d'extraction de carte de saillances pour des images de la base de données AerialTest1 avec l'algorithme d'Achanta.

L'algorithme de Katramados est le plus rapide (Tab.2.1. Cependant, il s'agit aussi de l'algorithme le plus sensible (l'écart type σ est le plus grand) et l'algorithme génère également beaucoup de bruit (la moyenne M est la plus grande). L'algorithme de Lu est le plus lent et il génère plus de bruit que celui d'Achanta. L'algorithme d'Achanta a les meilleures performances de réduction d'espace (voir exemples Fig.2.19). L'algorithme de saillance que nous retenons ne doit pas nécessairement être le plus rapide, ses performances de réduction de l'espace importent plus. En effet, un espace de recherche moins bien réduit conduira à plus d'opérations et donc fera ralentir les temps de calcul de la chaîne de traitement globale.

Performance de la chaîne de traitement "saillance"

Ici, nous comparons les performances de l'approche de recherche exhaustive par rapport à celles de la chaîne de traitement "saillance". Pour comparer les performances, nous utilisons les courbes ROC. La courbe ROC ("Receiver Operating Characteristic" en anglais, appelée aussi "caractéristique de fonctionnement du récepteur") est une courbe qui permet de visualiser les performances d'un détecteur pour un large éventail de sensibilité de détection, allant : du réglage le moins sensible au réglage le plus sensible. Cela permet d'avoir un aperçu des performances globales d'un détecteur, et donc, de comparer les performances globales de plusieurs détecteurs pour en déduire lequel est le plus efficace. Pour générer une courbe ROC il est nécessaire de disposer d'une base de données de test annotée.

Les annotations sont les emplacements des personnes dans l'image. Ces annotations sont utilisées pour collecter les faux-positifs (détections qui ne recouvrent pas suffisamment des annotations) et les vrais-positifs (détections qui recouvrent suffisamment des annotations). Le critère de PASCAL r est utilisé pour savoir si une détection recouvre suffisamment, ou pas, une détection :

$$r = \frac{aire(\text{boîte de détection}) \cap aire(\text{boîte d'annotation})}{aire(\text{boîte de détection}) \cup aire(\text{boîte d'annotation})} > 0.5 \qquad (2.1)$$

Nous avons effectué les tests sur la base de données de test annotée AerialTest1 [AerialTest1 2015]. Nous avons veillé à ce que l'approche classique et la chaîne de traitement "saillance" soient configurées pour détecter des personnes aux mêmes échelles. L'algorithme d'Achanta a été utilisé pour le calcul de la saillance.

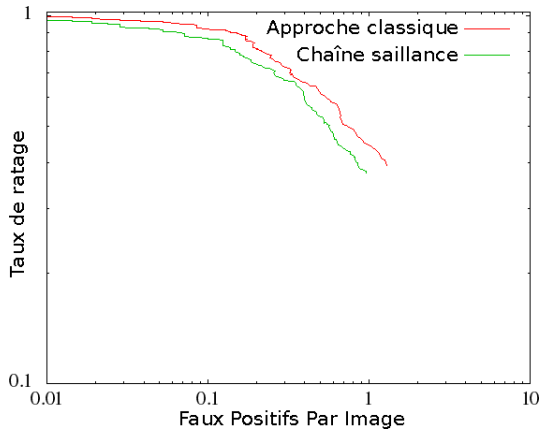


FIGURE 2.20 – Courbe ROC de la chaîne de traitement classique et de la chaîne de traitement "saillance" sur la base de données de test AerialTest1.

Les deux courbes ont une forme similaire. La courbe de la chaîne de traitement "saillance" est en dessous de l'autre ; les performances de détection sont donc légèrement meilleures (Fig.2.20). Cela est expliqué par le fait que notre chaîne de traitement "saillance" génère des fenêtres d'analyses qui sont très serrées les unes avec les autres autour des maximaux de saillances, alors qu'avec l'approche classique les fenêtres d'analyses sont traitées tous les 8 pixels sur l'axe des X et sur l'axe des Y. Le temps de calcul moyen avec l'approche classique est de 18.879 secondes par image (avec notre implémentation du détecteur HOG/ SVM et un choix de pyramide d'images dense). Avec notre chaîne de traitement "saillance" le temps de calcul moyen est de 2.286 secondes par image (avec la même densité d'échelle que pour l'approche classique). À noter que les deux chaînes de traitement utilisent un détecteur adapté à la vue aérienne détaillé Sec.2.2.2.1. Fig.2.21 est un exemple de résultats obtenus avec notre chaîne de traitement "saillance".

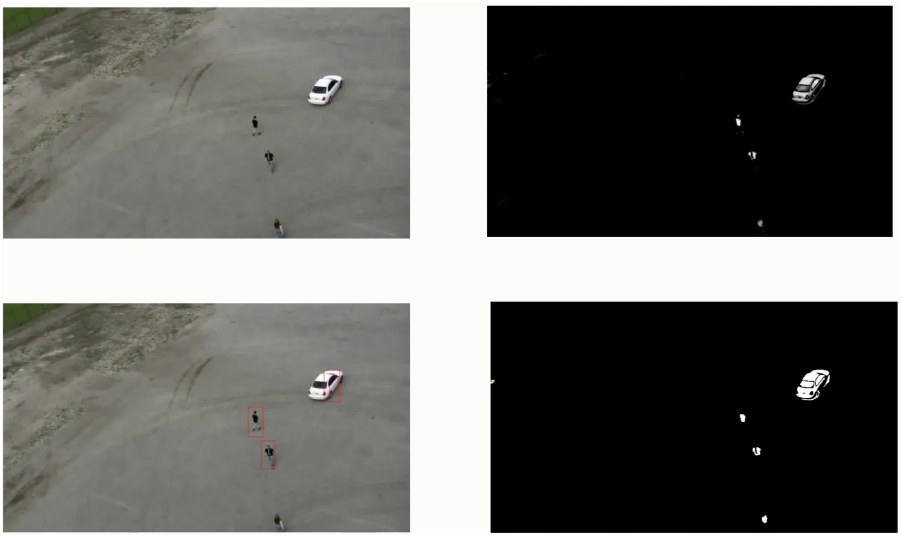


FIGURE 2.21 – Exemple de résultats obtenus avec notre chaîne de traitement "saillance".

2.2.2.3 Adaptation de la fenêtre de recherche

Dans cette partie, pour réduire les temps de calcul, nous proposons d'adapter la taille de la fenêtre de recherche en fonction de l'échelle des personnes à détecter.

Dans le cas aérien, il peut être intéressant d'adapter la fenêtre de détection, par exemple : en modifiant le ratio de la fenêtre de détection en utilisant les données de la centrale inertielle, ou encore, en modifiant la taille de la fenêtre de détection en fonction de la distance. Nous étudions ci-dessous les conséquences de ces adaptations pour le détecteur de piétons HOG/SVM de Dalal et al [Dalal 2005].

1. **Ajuster le ratio de la fenêtre de recherche**

FIGURE 2.22 – Ajustement de la fenêtre de recherche en fonction de l'élévation sur des images de synthéses.

Comme il a été mentionné ci-dessus, l'angle d'élévation (ou angle de tangage, du point de vue du drone) a une incidence sur la forme des personnes. Plus l'angle d'élévation est important, plus la forme des personnes à détecter est tassée : on pourrait donc adapter le ratio de la fenêtre de recherche afin de

rechercher un motif humain tassé (Fig.2.22). Moins l'angle d'élévation est important, moins la forme des personnes à détecter est tassée ; on peut donc revenir progressivement à un ratio un demi.

Nous proposons d'ajuster trois fois la fenêtre de recherche : 64x128 pixels et 8x8 pixels par cellule pour les angles d'élévations de 0 à 40 degrés, 64x112 pixels et 6x6 pixels par cellule pour les angles d'élévations de 40 à 60 degrés et 64x64 avec 16 pixels par cellule pour les angles d'élévations de 60 à 90 degrés. À noter que trois classifieurs différents doivent être entraînés dans ce cas ci, un classifieur pour chaque intervalle d'angle d'élévations.

2. **Utiliser des fenêtres plus petites**

Dans le cas aérien les distances aux personnes peuvent être bien plus importantes que dans le cas piéton. Ceci est particulièrement vrai si l'altitude du drone est importante. L'utilisation (seule) d'une fenêtre de recherche de taille standard (64x128 pixels) ne semble pas adaptée au cas aérien. Utiliser un taille de fenêtre plus petite permettrait d'accélérer les calculs pour la détection de personnes dont l'échelle est petite. Cela accélérerait les calculs à plusieurs niveaux : moins de pixels devraient être analysés et la pyramide d'image nécessiterait des niveaux moins sur-échantillonnés (Fig.2.23).

Nous proposons d'utiliser trois tailles de fenêtre d'analyse différentes : 64x128 avec 64 pixels par cellule, 48x96 avec 36 pixels par cellule et 32x64 avec 16 pixels par cellule ; Les tailles des fenêtres sont fonction de deux contraintes : 1) la nécessité d'avoir un écart suffisant entre les tailles des fenêtres et 2) la nécessité d'avoir un nombre de pixels par cellule toujours supérieur ou égal au nombre de secteurs d'histogramme, c'est-à-dire 9, comme recommandé par Dalal et al [Dalal 2005]). Nos trois tailles sont : Nous avons entraîné un classifieur différent pour chaque taille de fenêtre. L'idée est donc d'ajuster la taille de la fenêtre de détection en fonction de la profondeur de la pyramide d'images : la taille des niveaux de la pyramide d'images n'évolue plus progressivement sur toute la profondeur, mais change en fonction de la taille de la fenêtre de détection utilisée (Fig.2.23).

Une analyse comparative de complexité sur le nombre de classification de pixels permet de se rendre compte de la pertinence d'utiliser des fenêtres d'analyses plus petites dans le cas aérien. Considérerons une caméra avec un capteur CDD 1/3 pouce, une résolution d'image de 640x480 pixels et une lentille de 8mm. En considérant un intervalle de distances entre 10 et 40m pour une personne mesurant 1m70 : 1) l'intervalle d'échelles est entre 0.53 et 1.06 pour la fenêtre de détection 32x64, 2) entre 0.8 et 1.6 pour la fenêtre de détection 48x96 3) entre 1.06 et 2.13 pour la fenêtre de détection standard

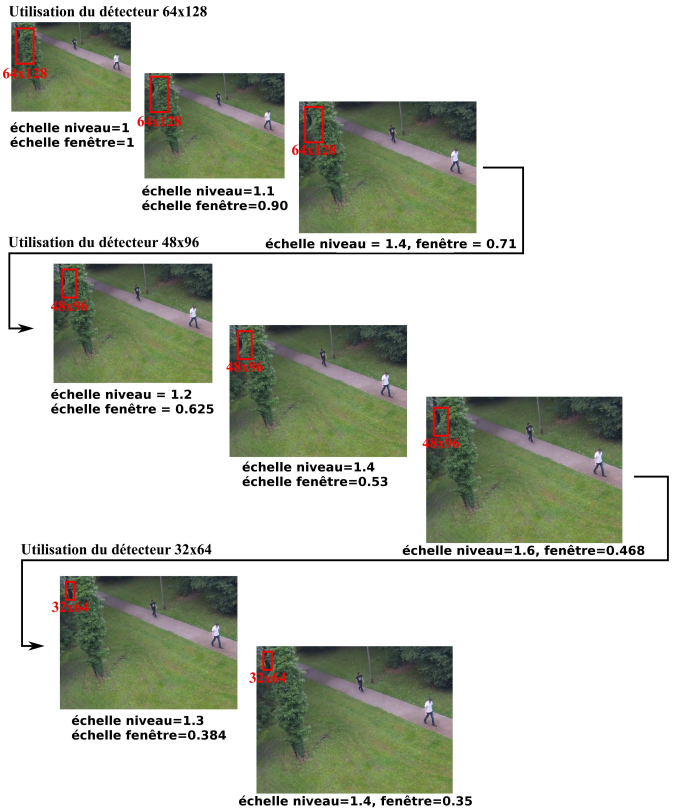


FIGURE 2.23 – Exemple de pyramide d'images pour plusieurs tailles de fenêtres de recherche, la taille des niveaux est adaptée à chaque type de fenêtre de détection. La pyramide d'images contient 8 niveaux. L'échelle de la fenêtre de détection varie entre 1 et 0.35 sur cette pyramide. Après trois niveaux la taille de fenêtre de détection change, ce qui réduit mécaniquement la taille du niveau et donc le nombre d'analyses de pixels. Après trois autres niveaux la taille de la fenêtre de détection est encore de nouveau adaptée.

(64x128 pixels). Le nombre de pixels à classifier pour une échelle est donné par la relation suivante :

$$Complexité(échelle) = surface \times échelle = 640 \times 480 \times s \qquad (2.2)$$

La Fig.2.24 montre la complexité en fonction de l'échelle s donnée. Le nombre total de classification de pixels entre deux niveaux est donné par l'aire sous la courbe. Pour les fenêtres 32x64 et 48x96, l'aire sous la courbe est respectivement 4.06 fois et 1.77 fois plus petite que l'aire sous la courbe de la fenêtre 64x128 (Fig.2.24).

En conclusion, pour ce cas où la personne est très éloignée, la configuration numéro 3 (fenêtre d'analyse 32x64) est plus rapide.

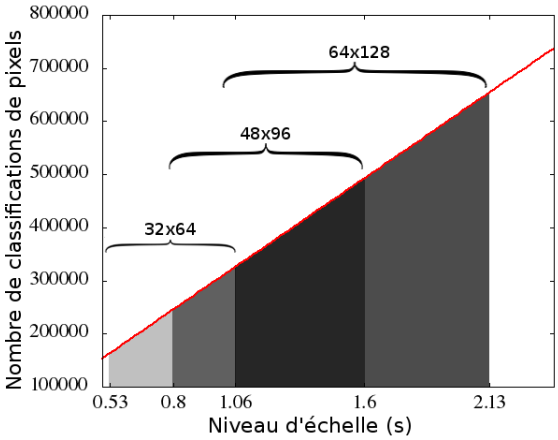


FIGURE 2.24 – Analyse de complexité du nombre de classifications de pixels pour l'utilisation de trois fenêtres de détection différentes pour un même cas.

Dans la suite de cette section nous comparons les performances de détection pour les trois fenêtres de détection 32x64, 48x96 et 64x128. Un apprentissage différent a été effectué pour chaque taille de fenêtre de détection : la base de données d'entraînement a été redimensionnée pour correspondre à la taille de la fenêtre ; l'extraction des caractéristiques a été modifiée également (comme décrit dans 2.2.2.3). La base de données d'entraînement utilisée est la base de données INRIA (Fig.2.25.1) [INRIA 2014].

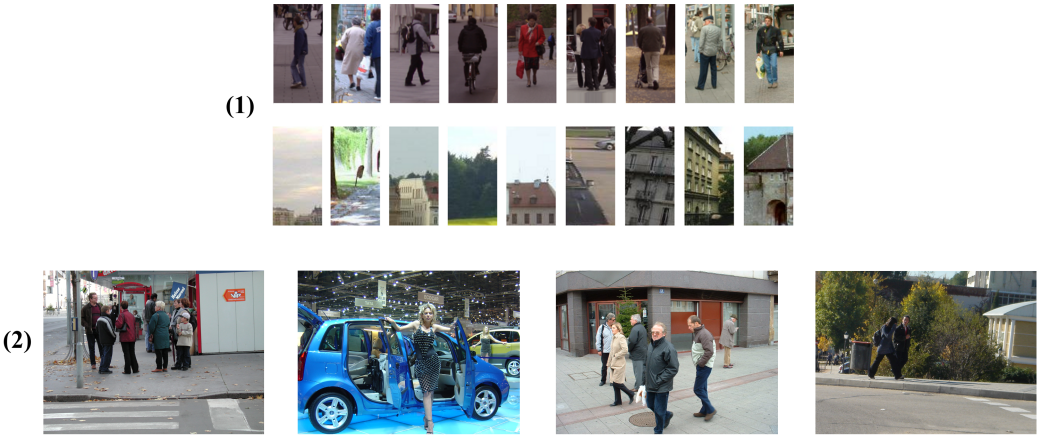


FIGURE 2.25 – Exemples d'images de la base de données INRIA. La base de données contient 289 images de test de personnes en vue piéton, 2416 images d'entraînement positives et 1218 images d'entraînement négatives pleine-résolutions. La taille des images d'entraînement est de 64x128 pixels.

Nous avons effectué les tests sur une version quatre fois sous-échantillonnée de la base de données de test INRIA (Fig.2.25.2) [INRIA 2014]. Le changement de la taille de la fenêtre d'analyse nécessite un changement de configuration de la pyramide d'images. Les pyramides d'images ont été configurées comme montré dans Tab.2.2 afin que l'analyse de l'image s'opère pour les mêmes échelles d'objet.

détecteur	nombre de niveaux	échelle minimum	échelle maximum
64x128	64	0.6	4.26
48x96	64	0.45	3.2
32x64	64	0.3	2.13

TABLE 2.2 – Configurations des pyramides d'images.

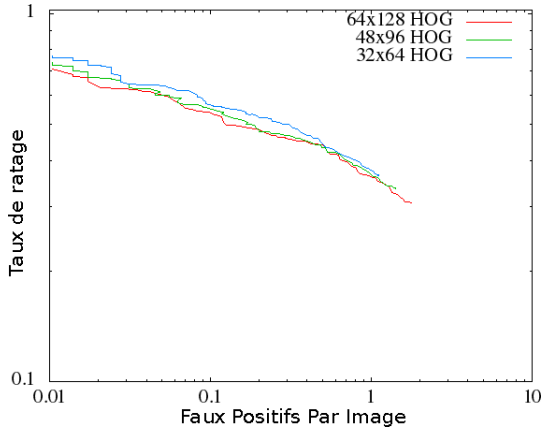


FIGURE 2.26 – Comparaison des performances de détection pour trois tailles de fenêtres d'analyse différentes.

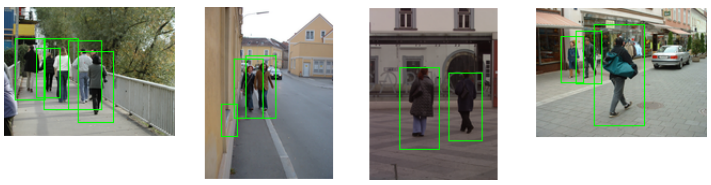


FIGURE 2.27 – Exemples de résultats obtenus sur la base de données de test INRIA sous-échantillonnée.

Les courbes obtenues pour le détecteur 32x64, 48x96 et 64x128 montrent des performances similaires (Fig.2.26). En effet, le taux de ratage pour 1 faux positif par image est le même pour les trois détecteurs. Par contre, le nombre de classification de pixels est moins important pour le classifier 32x64. Fig.2.27 présente quelques résultats obtenus sur la base de test INRIA sous-échantillonnée.

2.3 Vers une détection aérienne robuste complète

Dans cette partie nous proposons une approche de détection originale capable de gérer les angles de lacet, tangage et roulis de la caméra.

2.3.1 Limites de l'approche précédente

L'approche précédente ne corrige que les changements d'aspect résultants d'un angle non nul de tangage de la caméra. En effet, nous avons fait l'hypothèse que l'angle de roulis pouvait être corrigé ; ceci n'est pas forcément possible dans tous les cas : cela dépend de l'équipement du drone (de sa gamme, etc.).

De plus, l'approche précédente est basée sur le détecteur HOG/ SVM de Dalal et al [Dalal 2005]. Bien que le celui-ci soit une référence dans le domaine, il est plutôt long à l'exécution et ses performances de détection ont été surpassées depuis longtemps [Dollár 2009a]. Les détecteurs se basant sur l'approche ICF/SoftCascade (tel que le ACF/SoftCascade) présentent de nombreux avantages : 1) les performances de détection sont meilleures, 2) l'approche SoftCascade permet une classification très rapide des cas et 3) le framework du détecteur est facilement adaptable [Dollár 2009b]. Nous avons choisi d'adapter une approche basée sur l'ICF/SoftCascade.

2.3.2 Apprentissage multi-vues de formes

Dans la littérature on peut distinguer deux types de méthodes d'apprentissage multi-vues : les méthodes basées 3D et les méthodes basées 2D.

Les méthodes basées 3D se basent sur un modèle 3D de synthèse de l'objet, ou, sur un modèle 3D réaliste de l'objet. Yan et al proposèrent d'entraîner le détecteur en utilisant des caractéristiques visuelles calculées pour chaque vue 2D d'un modèle 3D reconstruit [Yan 2007]. À la détection, les caractéristiques sont mises en correspondances grâce à un codebook. Liebelt et al proposèrent d'obtenir les caractéristiques visuelles à partir d'un modèle 3D de synthèse de l'objet en associant à chaque vue un ensemble de caractéristiques visuelles [Liebelt 2008]. La détection est assurée par une mise en correspondance des caractéristiques calculées et des caractéristiques synthétiques grâce à un vote probabiliste 3D.

Avec ce type de méthode on apprend une version unique de l'objet qui n'est pas généralisable à d'autres objets de la même classe. Or, dans le cas de la détection de personnes, les vêtements et la morphologie des personnes diffèrent selon les individus ; la classe d'objet "humain" contient un très grand nombre de variantes. Il faudrait disposer d'un très grand nombre de modèles 3D humains et revoir l'étape d'apprentissage, ce qui serait extrêmement fastidieux à réaliser. Ce type de

méthode est plus adapté à la détection d'instance d'objet.

Les méthodes basées 2D utilisent plusieurs vues 2D de l'objet 3D et partent du principe qu'il y a invariance de caractéristiques pour les vues proches. En 2005, Huang et al proposèrent une variante vectorielle de l'algorithme d'apprentissage AdaBoost qui est appelée "VectorBoosting" [Huang 2005]. Cette approche permet d'apprendre un modèle 3D d'une classe d'objet en apprenant les différentes vues de celle-ci. Les images d'entraînement ont une double labélisation : une labélisation de type (cas positif ou négatif) et une labélisation de vue pour les cas positifs (de côté, de face, etc). La labélisation des vues est hiérarchisée sous la forme d'un arbre : chaque feuille de l'arbre correspond à une vue et les noeuds de l'arbre correspondent à des groupements de vues similaires (de plus en plus similaires à mesure que l'on descend l'arbre). Cette hiérarchisation permet un partage intelligent des caractéristiques lors de l'apprentissage et facilite donc l'apprentissage d'un modèle multi-vues. Les différentes vues de l'objet doivent être annotées à la main, ce qui est très fastidieux et ne garantit pas un apprentissage réellement optimisé. En 2006, Thomas et al proposèrent d'utiliser un Modèle de Forme Implicite ("Implicit Shape Model" en anglais) pour chaque vue 2D de l'objet 3D [Thomas 2006]. Les points d'intérêts invariants entre les modèles de vues voisines sont utilisés dans le but de permettre une détection multi-vues. Les classes d'objets que l'on peut apprendre et détecter avec cette approche sont essentiellement des classes d'objet rigides, telles que que la classe d'objet voiture ou la classe d'objet moto, par exemple. En 2007, Wu et al proposèrent une approche d'apprentissage similaire à celle du "VectorBoosting" mais qui ne nécessite pas de labéliser les vues manuellement [Wu 2007]. Cette approche s'appelle le "Cluster Boosting Tree". En 2010, Villamizar et al proposèrent une approche de détection d'objets invariantes à la rotation [Villamizar 2010]. La détection n'est pas robuste à des changements de points de vues complexes mais seulement aux rotations planes. Par la suite, nous nous sommes inspirés de la méthode "Cluster Boosting Tree" pour concevoir notre détecteur multi-vues.

Le Cluster Boosting Tree (CBT)

Le CBT[8] est un algorithme d'apprentissage basé sur AdaBoost. Cet algorithme a la spécificité de faire automatiquement des sous-groupes de données d'entraînement dans le but d'optimiser l'apprentissage d'une classe d'objet complexe [Wu 2007]. Le CBT est basé sur l'hypothèse que des vues différentes d'une même classe d'objet peuvent partager des caractéristiques communes. Avec le CBT, tout comme avec le VectorBoosting, les données d'entraînement sont structurées en un

8. *Cluster Boosting Tree*, Groupes de boosting d'arbre

arbre binaire.

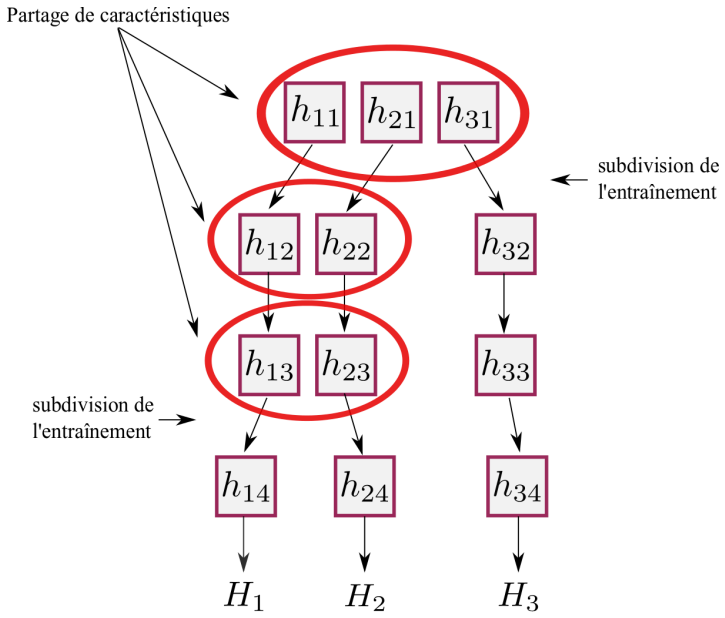


FIGURE 2.28 – Structure en arbre d'un classifieur entraîné par le Cluster Boosing Tree.

1. **Apprentissage**
   À l'apprentissage, le pouvoir de classification de chaque classifieur faible est estimé. Si le pouvoir de classification du classifieur faible est trop bas cela signifie que l'algorithme n'a pas pu trouver une classification faible avec un bon pouvoir discriminant. Dans ce cas, le CBT subdivise les éléments d'entraînement en deux sous-ensembles disjoints d'éléments d'entraînement en utilisant un algorithme de groupement non-supervisé. L'algorithme de groupement utilisé est l'algorithme k-means (algorithme des k-moyennes, en français). En subdivisant le problème on facilite l'apprentissage. Après chaque subdivision, l'apprentissage de nouveaux classifieurs faibles continue en parallèle sur les deux nouveaux sous-ensembles de données. Cette subdivision donne naissance à deux nouvelles branches d'arbre (Fig.2.28). Ainsi, la structure du classifieur est un arbre binaire qui se construit de manière dynamique au fur et à mesure des subdivisions. Chacun des chemins (allant du noeud racine aux feuilles de l'arbre) est appelé un canal. À noter qu'après chaque subdivision les classifieurs faibles précédemment appris ont leurs seuils réajustés en fonction des données de chaque sous-groupe.

   Concrètement, le CBT commence l'apprentissage en considérant toutes les données d'entraînement en même temps (c'est-à-dire, avec toutes les vues, sans distinction). Le CBT apprend les classifieurs faibles en série jusqu'à

ce que le pouvoir de classification des classifieurs faibles soit trop bas. Le pouvoir de classification d'un classifieur faible est inversement proportionnel à Z ([E.Schapire 1999]) :

$$Z = \sum_{m=0}^{M} \sum_{n=0}^{N} w_n \times exp(-\alpha_m y_n h_m(x_n)) \tag{2.3}$$

Où M est le nombre de classifieurs faibles entraînés, N le nombre d'éléments d'entraînement, y_n est la labélisation de l'élément d'entraînement n (si $y_n = 1$ alors c'est un élément d'entraînement positif, si $y_n = -1$ c'est un élément négatif), h_m est le m-iéme classifieur faible et x_n est le vecteur contenant les caractéristiques de l'élement n.

Si les trois derniers classifieurs faibles appris ont un pouvoir de classification trop bas (Z supérieur à un seuil Θ_Z, Fig.2.29.2), alors le CBT procédera à une subdivision des données d'entraînement en appliquant un algorithme k-means (avec k=2) aux données d'entraînement évaluées par les classifieurs faibles qui ont été appris jusqu'à présent (Fig.2.28). L'apprentissage des classifieurs faibles continue séparément pour chaque groupe de données d'entraînement. Si le pouvoir de classification décroît à nouveau, l'opération est répétée, etc. Les différents groupes que crée automatiquement le CBT peuvent être considérés comme des vues abstraites de la classe d'objet à apprendre.

Intuitivement Z tendra rapidement vers sa valeur maximale 1 si la base de données d'entraînement contient des vues très différentes de l'objet (Fig.2.29.2). Car l'algorithme d'apprentissage peinera rapidement à trouver des caractéristiques visuelles communes à toutes les vues. Le CBT corrige ce problème en groupant les données d'entraînement par affinité afin de continuer à apprendre des classifieurs faibles à fort pouvoir de discrimination. À noter que lorsqu'il y a peu de variations intra-classe Z tend vers une valeur seuil (Fig.2.29.1).

2. **Prédiction**

Le classifieur entraîné par le CBT est formulé comme suit :

$$H(x) = [H_1(x), H_2(x), ...H_C(x)] \tag{2.4}$$

$$H_c(x) = \sum_{t=0}^{T} h_{(c,t)}(x) \tag{2.5}$$

Où T est le nombre de classifieurs faibles. On dit qu'un exemple est accepté par le canal c du classifieur si la propriété suivante est vérifiée :

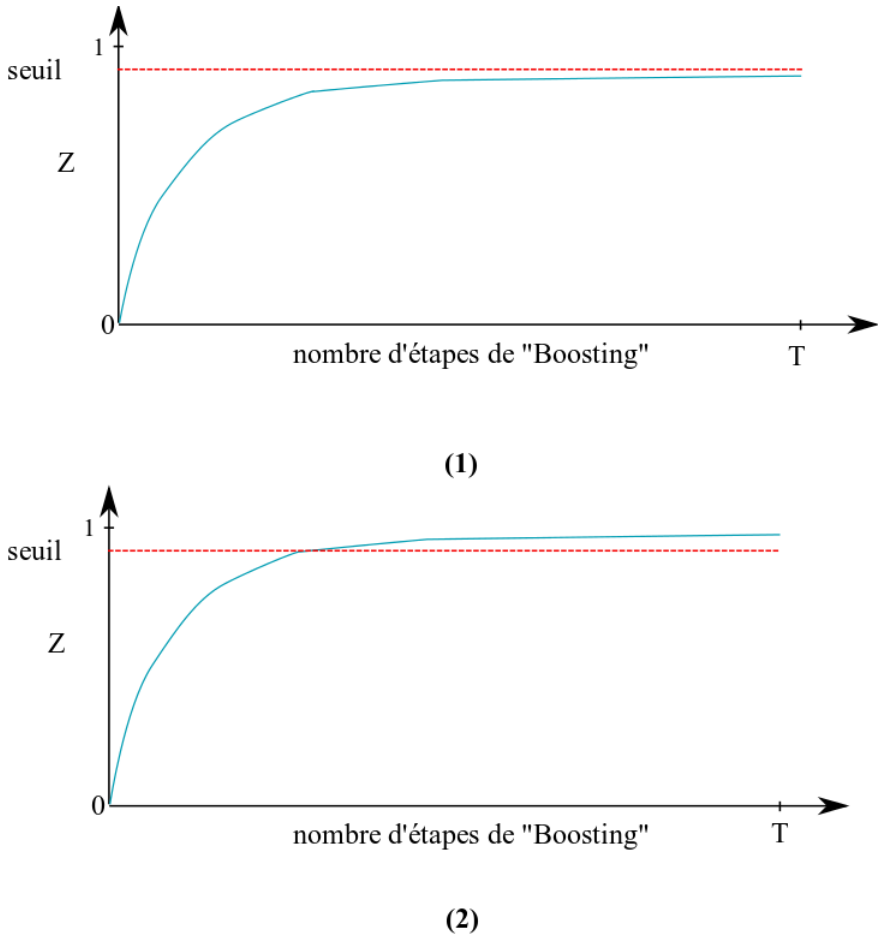


FIGURE 2.29 – Évolution du pouvoir de classification des classifieurs faibles en fonction du nombre d'étapes de "Boosting".

$$H_c(x) = \sum_{t=0}^{T} h_{(c,t)}(x) > \Theta_Z \tag{2.6}$$

Un exemple est classifié comme étant positif si un seul canal accepte l'exemple. Il n'est pas nécessaire d'analyser les autres canaux dans ce cas là. Dans le cas contraire, si aucun canal n'accepte l'exemple, l'exemple sera classifié comme négatif.

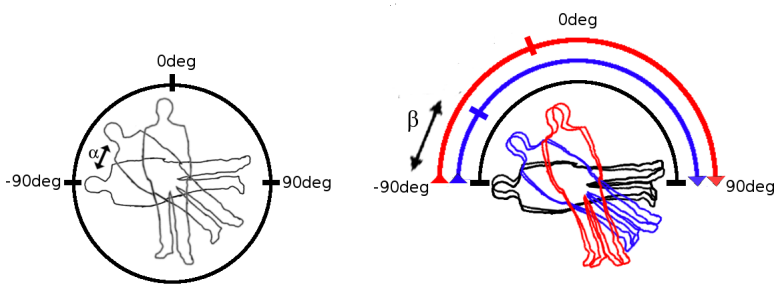


FIGURE 2.30 – Étalement angulaire des données d'entraînement pour simuler l'effet du roulis.

2.3.3 Notre détecteur : le "Pitch and Roll-trained Detector"

Nous avons conçu le détecteur "Pitch and Roll-trained Detector" ou PRD [9] (détecteur entraîné roulis et tangage, en français) pour détecter les personnes au sol quels que soient les angles de roulis et/ou de tangage du système de vision du drone. L'apprentissage du classifieur est effectué en utilisant une variante de l'algorithme CBT. Les caractéristiques visuelles utilisées par le détecteur sont les caractéristiques de canaux intégraux qui sont rapides à calculer et robuste aux changements d'illumination. À noter que, les caractéristiques de canaux agrégés (ACF) peuvent également être utilisées.

Apprentissage

Notre algorithme d'apprentissage nécessite que les données d'entraînement contiennent des vues de personnes prises pour différents angles de roulis et de tangage du drone. Les changements de forme résultant d'un changement d'angle de tangage ne peuvent pas être simulés : il est nécessaire d'utiliser des images de personnes prises pour différents angles de tangage. Les changements de forme résultants d'un changement d'angle de roulis peuvent être facilement reproduits par une simple rotation des images. Concrètement, une fois que les données d'entraînement multi-élévations sont chargées en mémoire, celles-ci sont dupliquées un certain nombre de fois puis étalées plusieurs fois (Fig.2.30) sur 180 degrés d'angle et à partir de plusieurs angles de départs β. Aucune rotation n'est effectuée pour les angles entre 90 et -90 degrés.

Nous calculons les caractéristiques dans une fenêtre d'analyse circulaire de rayon 64 pixels pour chaque élément d'entraînement. Cela permet de ne s'inté-

9. *Pitch and Roll-trained Detector*, Détecteur de personnes entraîné pour les angles de roulis et de tangage

resser qu'aux caractéristiques visuelles contenues au centre des images d'entraînement. De plus, la forme circulaire permet une extraction aisée des caractéristiques quel que soit l'angle de roulis.

Notre algorithme d'apprentissage est basé sur le CBT : les éléments d'entraînement sont subdivisés quand le pouvoir de classification des classifieurs faible diminue et les seuils des classifieurs faibles précédemment appris sont réajustés en fonction de chaque sous-groupe (Fig.2.31). Le type de classifieur faible que nous avons utilisé ici est un arbre binaire de décision de profondeur 2. Le classifieur final est également sous la forme donnée dans Equ.2.4 : c'est-à-dire, sous la forme d'un ensemble de canaux. Nous calculons une trace de rejet pour chaque canal du classifieur ; chaque canal peut donc être considéré comme un sous-classifieur SoftCascade.

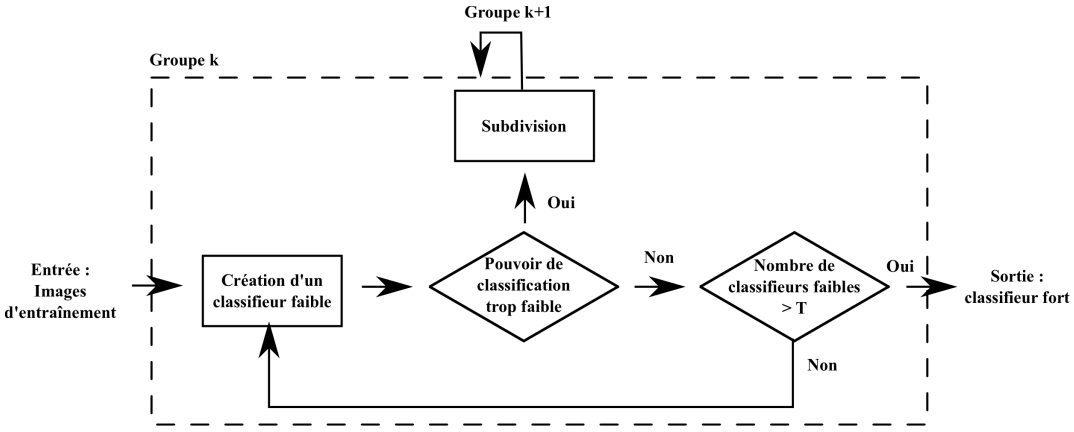


FIGURE 2.31 – Principe de fonctionnement de l'apprentissage du classifieur : les classifieurs faibles sont appris au fur et à mesure, quand le pouvoir de classification est trop bas, les images d'entraînement sont subdivisées en groupes et les apprentissages continués en parallèle.

Le pseudo-code de notre algorithme d'entraînement est présenté ci-après. La base de données contenant tous les cas positifs est notée S_+^1 et la base de données contenant les cas négatifs est notée S_-. Le pouvoir de classification du classifieur faible t du canal c est noté : $h(c,t).Z$. Les différents paramètres sont : l'identifiant du canal c, le classifieur faible initial à partir duquel est commencé l'entraînement

pour le canal c $inital\_t(c)$ et le nombre de canaux.

Algorithm 6: Notre algorithme d'apprentissage

Data: Base de données multi-élévations
Result: Classifieur multi-vues

$nombre\_de\_canaux = 1$
$initial\_t(1) = 0$
for $c = 0$ *jusqu'à* $nombre\_de\_canaux$ **do**
    (Ré)initialiser les poids des éléments d'entraînement de S_+^c à $\frac{1}{card(S_+^c)}$
    (Ré)initialiser les poids des éléments d'entraînement de S_- à $\frac{1}{card(S_-)}$
    for $t = 0$ *jusqu'à* T **do**
        Trouver $h_{c,t}$ minimisant l'erreur pondérée sur S_+^c et S_-
        Mettre à jour les poids des éléments d'entraînement
        if $h_{c,t}.Z > \Theta_Z$ *et* $h_{c,t-1}.Z > \Theta_Z$ *et* $h_{c,t-2}.Z > \Theta_Z$ **then**
            S_+^c subdivisé avec k-moyennes en S_+^c et $S_+^{nombre\ de\ canaux+1}$
            for $t' = 0$ *jusqu'à* t **do**
                $h(nombre\ de\ canaux + 1, t') = h(c, t')$
                réajuster les seuils de $h(c, t')$ pour S_+^c et S_-
                réajuster les seuils de $h(nombre\ de\ canaux + 1, t')$ pour $S_+^{nombre\ de\ canaux}$ et S_-
            end
            $initial\_t(\ nombre\ de\ canaux + 1) = t$
            $nombre\ de\ canaux$ = $nombre\ de\ canaux + 1$
        end
    end
end
for $c = 0$ *jusqu'à* $nombre\_de\_canaux$ **do**
    Construire une trace de rejet
end

Prédiction

La phase de prédiction de notre approche est légèrement différente de celle d'un classifieur entraîné avec le CBT ; elle est faite en fonction des traces de rejets des canaux. Concrètement, il suffit que tous les classifieurs faibles d'un seul canal soient passés pour que l'exemple soit classifié comme étant positif. Si aucun canal ne passe en entier l'exemple est classifié comme étant négatif.

2.3.4 Expérimentations

Dans un premier temps, nous avons étudié la pertinence du détecteur PRD pour la détection de personnes en vue aérienne. Notre détecteur PRD a été entraîné avec un seuil $\Theta_Z = 0.98$ et un nombre de classifieurs faibles maximal par canal $T = 500$. L'apprentissage a généré 6 canaux différents. Dans un second temps, nous avons optimisé son paramètrage pour obtenir les meilleurs performances.

Pertinence du détecteur PRD pour la détection de personnes en vue aérienne

Nous avons comparé les performances de quatre détecteurs : 1) le détecteur RD [10] (entraîné pour être seulement robuste au roulis), 2) le détecteur PD [11] (entraîné pour être seulement robuste au tangage), 3) le détecteur PRD (entraîné pour être robuste à la fois au roulis et au tangage) et 4) le détecteur de piétons ICF/SoftCascade de Dollàr et al [Dollár 2009b]. Les détecteurs RD et PD ont été entraînés en utilisant le même algorithme d'entraînement que le PRD. Le détecteur RD a été entraîné en utilisant une version 128x128 de la base de données d'entraînement INRIA, et le détecteur PD a été entraîné en utilisant notre base de données GMVRT2 [12]2 (sans la simulation du roulis à l'apprentissage). La Fig.2.32 donne quelques exemples de notre base de données GMVRT2 [GMVRT2 2015].

FIGURE 2.32 – Exemples d'images d'entraînement positives de la base de données GMVRT2. La base de données contient 3846 images d'entraînement positives et 13280 images d'entraînement négatives de taille 128x128. Les images ont été extraites de plus d'une centaine de vidéos.

Le PRD a été entraîné en utilisant la base de données GMVRT2 et tel qu'il a été décrit dans Sec.2.3.3. Notre base de données d'apprentissage GMVRT2 est beaucoup plus riche que la base données GMVRT1 ; cette base de données contient plus de cas et est adaptée à l'usage d'une fenêtre d'analyse circulaire.

Le taux de détection moyen du détecteur ICF commence à chuter à partir de 35 degrés d'angle d'élévation (Fig.2.33). Le taux est nul à partir de 60 degrés d'élévation. On observe que les détecteurs PD et PRD ont un taux de détection relativement

10. *Roll Detector*, Détecteur de personnes entraîné pour l'angle de roulis
11. *Pitch Detector*, Détecteur de personnes entraîné pour l'angle de tangage
12. *Generalized Multi-View Realistic Training dataset 2*, Base de données d'entraînement contenant des images réalistes multi-élévations très diverse

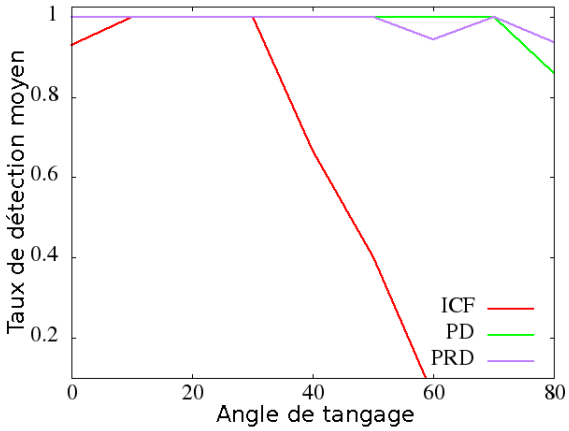


FIGURE 2.33 – Comparaison de la robutesse au changement d'angle d'élévation des détecteurs PRD, PD et ICF.

similaire quel que soit l'angle d'élévation. La robustesse à l'angle d'élévation a été testée sur un ensemble d'images prises pour différents angles d'élévation compris entre 0 et 80 degrés [ElevationTest 2015].

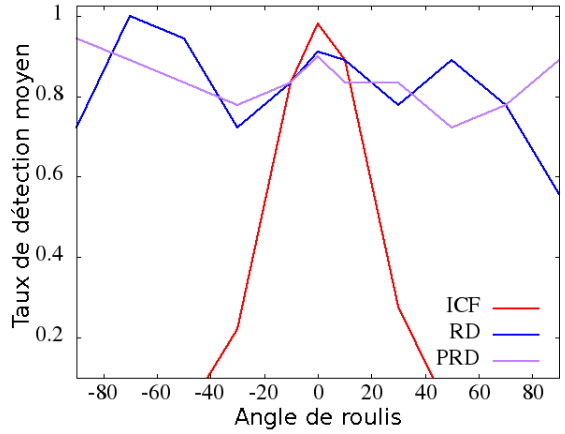


FIGURE 2.34 – Comparaison de la robustesse à l'angle de roulis du détecteur PRD, RD et ICF.

La robustesse à l'angle de roulis a été testée sur 180 images de test prises à des angles de roulis entre -90 et 90 degrés [RoulisTest 2015]. Quelques exemples sont donnés en Fig.2.35). Le taux de détection moyen du détecteur ICF est nul entre -40 degrés et 40 degrés d'angle de roulis (Fig.2.34). Le taux est acceptable entre -20 degrés et 20 degrés d'angle de roulis. À contrario, les détecteurs RD et PRD ont un taux de détection relativement stable entre -90 et 90 degrés d'angle de roulis. Cependant, le détecteur PRD semble plus stable que le détecteur RD.

FIGURE 2.35 – Exemples d'images utilisées pour tester la robustesse au roulis des détecteurs (ici les roulis sont respectivement : -90, -70, -30, 30, 70 et 90 degrés).

Nous avons aussi comparé les performances globales des quatre détecteurs en utilisant notre base de données AerialTest1 (Fig.2.36). Fig.2.37 montre quelques résultats qualitatifs.

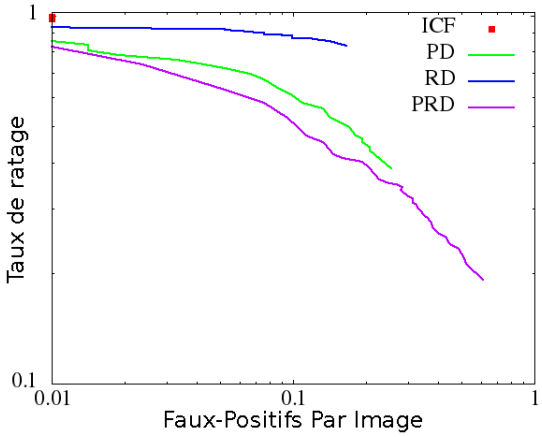


FIGURE 2.36 – Comparaison des performances globales des détecteurs PRD, PD, RD et ICF.

Le détecteur de piétons ICF échoue totalement sur la base de données AerialTest1. On peut observer une légère amélioration des performances de détection avec le détecteur RD. L'amélioration est cependant bien plus importante avec le détecteur PD. Les meilleures performances sont obtenues quand le tangage et le roulis sont considérés durant la phase d'apprentissage ; le détecteur PRD a la meilleure courbe ROC.

Nous avons décidé de ne pas évaluer le temps de calcul moyen par image du détecteur RD en raison de ses faibles performances de détection. Le détecteur PRD testé est 1,75 fois plus lent que le détecteur ICF/SoftCascade de Dollàr et al (Tab.2.3). Ce ralentissement est dû au nombre plus important de classifieurs faibles qui doit être évalué avec le détecteur PRD. Le détecteur PRD testé contient six fois plus de classifieurs faibles que le détecteur ICF/SoftCascade originel (qui en contient 1000). L'approximation des caractéristiques visuelles permet au détecteur PRD testé d'atteindre le même temps de calcul moyen que celui du détecteur ICF.

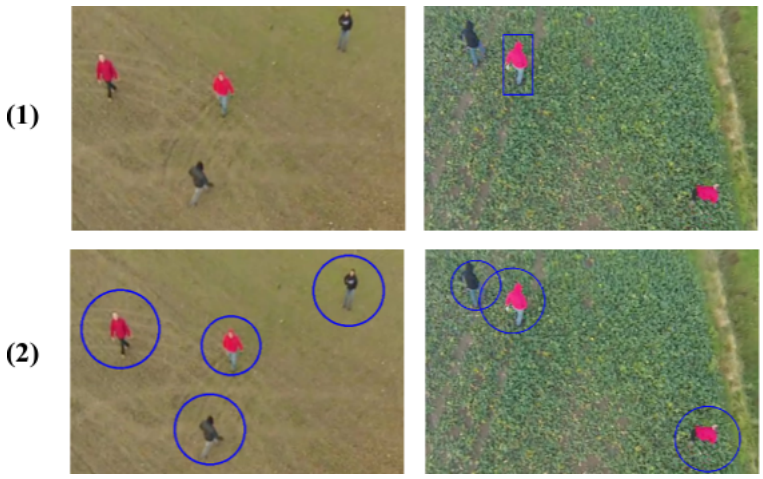


FIGURE 2.37 – Comparaison qualitative des résultats obtenus avec le détecteur ICF/SoftCascade (première colonne) et avec le détecteur PRD (deuxième colonne) sur la base de données AerialTest1.

	ICF	PD	PRD
Sans approx.	T	T	1.75×T
Avec approx.	0.35×T	0.38×T	1.05×T

TABLE 2.3 – Comparaison du temps de calcul moyen par image pour les détecteurs PRD, PD et ICF.

Paramétrisation du détecteur PRD

Choisir les paramètres d'apprentissage T et Θ_Z optimaux du détecteur PRD nécessite de tester un grand nombre de configurations différentes.

Nous avons avons testé les performances du détecteur PRD pour plusieurs configurations d'entraînement : en faisant varier le nombre de classifieurs faibles T de 300 à 500 et le seuil Θ_Z entre 0.950 et 0.99. En effet nous avons observé que pour $T < 300$ les performances sont dégradées sérieusement et pour $T > 500$ les performances de détection restent équivalentes. Nous avons également remarqué expérimentalement qu'en dessous de $\Theta_Z = 0.950$ le groupage ne se faisait pas de manière optimale.

Au vu des résultats, nous pouvons observer que le paramètre T a peu d'influence sur les performances du détecteur (Fig.2.38) ; les performances ne sont pas guidées par le choix d'une valeur spécifique de T. Nous avons choisi de faire varier T entre un minimum $T = 300$ et un maximum $T = 500$, nous avons en effet observé expérimentalement qu'il n'était pas nécessaire de dépasser $T = 500$ et que $T = 300$ était le minimum nécessaire pour un bon apprentissage par canal.

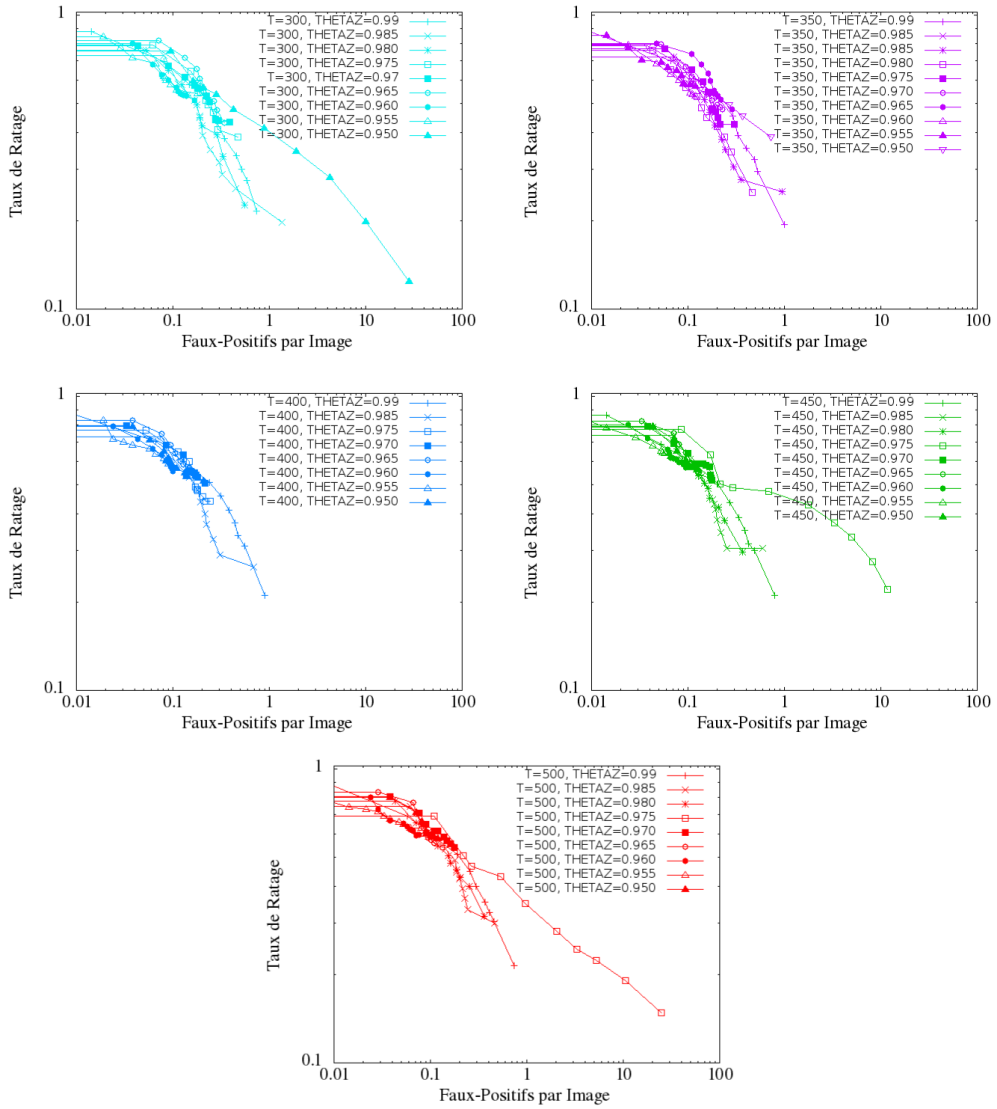


FIGURE 2.38 – Évolution des performances du PRD en fonction de T et Θ_Z

Le paramètre influant le plus les performances est Θ_Z : une légère variation de cette valeur entraînera une subdivition des images d'entraînement positives plus tôt ou plus tard, et cela a généralement des conséquences en cascade ; le nombre de canaux peut augmenter ou diminuer considérablement et les performances peuvent également être affectées. Par conséquent, pour choisir les meilleurs paramètres d'apprentissage du PRD nous conseillons de tester les performances pour un grand nombre de configurations et enfin de choisir la meilleure configuration. À noter qu'il est conseillé de choisir une configuration qui ne génère pas trop de canaux, au risque d'alourdir considérablement les temps de calcul à la détection. Nous pouvons observer que les meilleures performances sont obtenues pour une valeur de

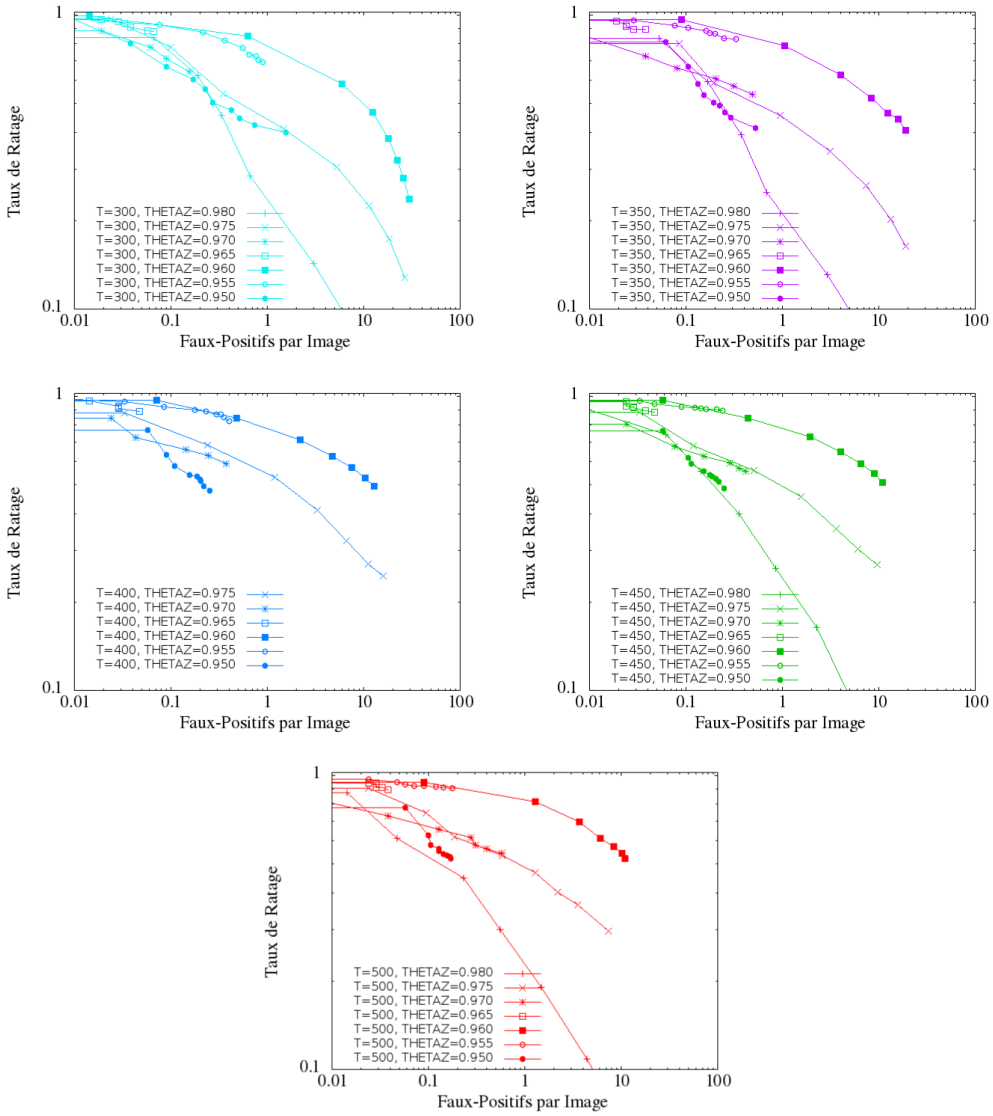


FIGURE 2.39 – Évolution des performances du PRD/ACF en fonction de T et Θ_Z

Θ_Z proche de 0.980 quelque soit le nombre de classifieurs faibles T.

Nous avons effectué les même tests avec un PRD utilisant les caractéristiques de canaux agrégés (ACF) (Fig.2.39). Nous pouvons également observer que T a peu d'impact sur les performances du détecteur. Ici aussi le paramètre Θ_Z a une grande influence sur les performances. Dans ce cas précis, nous pouvons observer trois groupes de courbes qui sont fonction de Θ_Z. Ces groupages permettent de mieux identifier les valeurs de Θ_Z les plus aptes à permettre un entraînement optimal.

Nous pensons que cette différence de comportement (par rapport au PRD classique) s'explique par le fait que le PRD classique utilise des caractéristiques visuelles qui sont générées aléatoirement et de manière disparate dans l'espace (les

caractéristiques de canaux intégraux). Le PRD/ACF, quant à lui, utilise des caractéristiques qui sont densément générées, permettant ainsi une sépration plus fine entre les éléments d'entraînement et donc un meilleur groupement de ceux-ci en fonction de Θ_Z. Les meilleures performances sont également obtenues pour une valeur de Θ_Z proche de 0.980.

2.4 Conclusion et perspectives

Pour conclure :

1. Nous avons proposé une chaîne de traitement basée sur la réduction de l'espace de recherche par une analyse de la saillance de la scène. Le traitement ne nécessite aucun suivi temporel ; il peut se faire image après image. Aucun capteur supplémentaire n'est nécessaire. Cette approche est particulièrement bien adaptée pour les milieux ouverts et non-encombrés. Elle permet de réduire considérablement les temps de calcul.
2. Nous avons observé qu'un entraînement multi-élévations permettait d'adapter facilement un détecteur de piétons de référence (en l'occurrence le détecteur HOG/SVM) du cas piéton au cas aérien. Le détecteur entraîné avec une base de données multi-élévations détecte des personnes au sol quel que soit l'angle de tangage du système de vision du drone.
3. Nous avons vu qu'il n'était pas nécessaire d'adapter la forme de la fenêtre d'analyse en fonction de l'angle d'élévation. De plus, cela complique la conception du détecteur (obligeant à avoir un détecteur pour chaque tranche d'angle d'élévation). Nous avons observé que le taux moyen de détection était similaire à celui de l'approche multi-élévations qui n'adapte pas la forme de la fenêtre en fonction de l'angle d'élévation. Avec cette approche, le taux de FPPI explose en fonction de l'angle d'élévation et de la distance.
4. Nous avons vu que changer la taille de la fenêtre de détection permettait de réduire considérablement le nombre de traitement de pixels et donc permettait de gagner du temps d'exécution. Les performances de détection de personnes éloignées ne sont que légèrement détériorées avec la réduction de la fenêtre d'analyse.
5. Nous avons proposé un détecteur (le PRD) permettant une détection des personnes au sol quels que soient les angles d'élévation et de roulis combinés du système de vision par rapport au sol. Le détecteur PRD peut cependant générer un nombre important de faux-positifs dans les milieux encombrés en raison de sa plus grande sensibilité de détection.

Les expérimentations précédentes montrent les limites de la détection de personnes dans le spectre visible. La prise en compte d'une modalité supplémentaire,

telle que la modalité infrarouge, permettrait d'étendre les capacités de détection du détecteur. Concrètement, cela permettrait de :

1. Faire baisser le taux de FFPI du détecteur PRD pour la détection aérienne, dans le but d'utiliser le détecteur dans des milieux plus encombrés.
2. Étendre la détection de personnes à la nuit et permettre une détection dans des milieux où les conditions d'éclairage changent très fortement.
3. Accélérer les temps de calcul du PRD par une analyse plus adéquate de l'espace de recherche.

CHAPITRE 3

Détection de personnes dans le spectre visible et infrarouge

3.1 Le spectre infrarouge

L'analyse du spectre infrarouge présente des avantages indéniables pour la détection de personnes : l'être humain étant une source d'infrarouge constante, la recherche de ces sources dans la scène aide considérablement la détection. Dans cette section nous détaillons la relation qu'il existe entre la température du corps humain et les longueurs d'onde qu'il émet dans l'infrarouge. Nous détaillons également les différents détecteurs existant pour capter les infrarouges.

3.1.1 La relation entre la température et la longueur d'onde

Tout objet (ou corps) ayant une température supérieure au zéro absolu émet un rayonnement électromagnétique. Ce rayonnement électromagnétique est la conséquence de l'agitation des atomes constituant l'objet. Un objet dont la température est à 0° Kelvin ne transmet pas de rayonnement électromagnétique, car ses atomes sont figés [Bruhat 1968].

Le rayonnement électromagnétique d'un objet constitué de plusieurs matériaux comporte plusieurs longueurs d'onde, c'est-à-dire, une pour chaque type d'atome constituant l'objet. Pour simplifier l'analyse, on considère un corps constitué d'un seul matériau qui vérifie la propriété suivante : pour une température donnée, le corps émet le maximum d'énergie. Ce corps idéal s'appelle le corps noir [Bardon 1998][Bruhat 1968].

La longueur d'onde maximale λ_{max} pour un corps noir de température t (en degré Kelvin) est donnée par la loi de Wien [Bardon 1998] :

$$\lambda_{max} = \frac{2.898 \times 10^{-3}}{t} \tag{3.1}$$

Cette loi illustre la relation entre la longueur d'onde et la température. Elle a été déduite de la loi de Planck sur le rayonnement électromagnétique d'un corps noir donnée dans Equ.3.2. Les constantes de l'équation sont : la constante de Planck $h = 6,62606957 \times 10^{-34} J.s$, la vitesse de lumière dans le vide $c = 299792458 m.s^{-1}$ et la constante de Boltzmann $k = 1,3806488 \times 10^{-23} J.K^{-1}$.

$$M(\lambda, t) = \frac{2hc^2\lambda^{-5}}{exp^{(\frac{hc}{k\lambda t})} - 1} \tag{3.2}$$

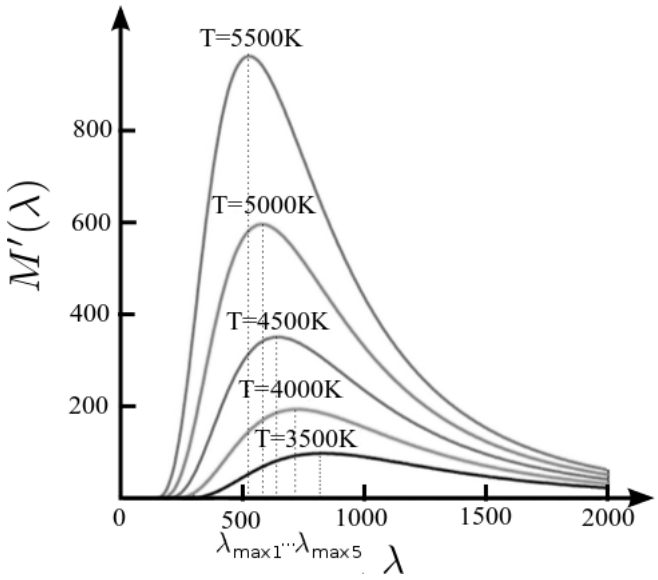


FIGURE 3.1 – Densité d'énergie spectrale en fonction de λ.

Fig.3.1 illustre les valeurs λ_{max} obtenus pour les températures $T = 5500K$, $T = 5000K$, $T = 4500K$, $T = 4000K$ et $T = 3500K$. Les λ_{max} sont obtenus pour les maximaux des fonctions $M'(\lambda) = M(\lambda, t = T)$.

L'émissivité ε est le ratio entre l'énergie rayonnée par un corps (M_c) et l'énergie rayonnée par le corps noir donnée par la loi de Planck (Equ.3.3) [Bardon 1998][Bruhat 1968]. Elle permet d'approximer le rayonnement électromagnétique d'un corps non-idéal (ou corps gris, dans le cas où l'émissivité est constante). L'émissivité du corps noir est égale à 1. L'émissivité de la peau humaine est $\varepsilon = 0.985$ [Rapport Mikron Company].

$$\varepsilon = \frac{M_c}{M} \tag{3.3}$$

Prenons le cas concret de la mesure de la température d'une personne. Le domaine spectral infrarouge étendu est situé entre $750nm$ et $1000\mu m$. La température du corps noir oscille dans ce cas entre 2,898°K (-270.252°C) et 3864°K (3590.85°C). Considérons que la température à la surface de la peau humaine se situe entre 301.15°K (28°C) et 306.15°K (33°C). Faisons l'hypothèse que les radiations ne sont pas réfléchies avec le fond. Dans ce cas, la température mesurée par un capteur idéal sera donc comprise entre $\varepsilon \times$301.15°K (297,83°K) et $\varepsilon \times$306,15°K (302,78°K). On constate que les températures sont comprises entre la température

minimale et la température maximale d'un corps noir rayonnant dans le spectre infrarouge étendu. Plus précisément, la longueur d'onde oscille donc entre 9871nm et 10,0036μm, ce qui correspond au domaine spectral des infrarouges longs (voir ci-dessous Tab.3.1).

	Visible	Inf. proche	Inf. court	Inf. moyen	Inf. long	Inf. lointain
min	350(nm)	750(nm)	1400 (nm)	3000 (nm)	8000 (nm)	15 (μm)
max	750(nm)	1400(nm)	3000 (nm)	8000 (nm)	15 (μm)	1000 (μm)

TABLE 3.1 – Domaine spectral visible et domaines spectraux infrarouge.

La température du corps humain est, dans la plupart des cas, plus importante que la température du fond (Tab.3.1). Le contraste de température entre les êtres humains et le fond est suffisamment important pour dissocier les êtres humains du fond grâce à l'analyse des images infrarouges.

3.1.2 Les détecteurs à infrarouge

Les détecteurs à infrarouge peuvent être classés en deux catégories : 1) les détecteurs à photons (le rayonnement absorbé par la matière interagit avec les électrons de celle-ci) et 2) les détecteurs thermiques (le rayonnement absorbé change les propriétés physiques de la matière et engendre un changement du courant électrique le traversant) [Rogalski 2002].

(1) Les détecteurs à photons ont une grande réactivité et une grande sensibilité au rayonnement infrarouge. En imagerie, les détecteurs à photons de type semi-conducteur sont les plus utilisés (le semi-conducteur Tellurure de mercure-cadmium est le plus répandu ; il est opérationnel pour une très large portion du spectre infrarouge). Cependant, un refroidissement est nécessaire pour son bon fonctionnement [Rogalski 2002]. De plus, la conception d'une caméra utilisant ce type de détecteur est en général complexe. Ces caméras (que l'on appelle aussi : caméras infrarouges refroidies) consomment plus d'énergie et ont un encombrement et un poids plus important (Fig.3.2). Le prix d'une caméra infrarouge refroidie est très élevé ; ce qui restreint considérablement son accessibilité sur le marché et le développement d'applications utilisant cette technologie.

(2) Historiquement, les détecteurs thermiques sont les premiers détecteurs à infrarouge. On peut scinder les détecteurs thermiques en deux sous-classes : les détecteurs pyroélectriques et les détecteurs bolométriques [Rogalski 2002]. Les détecteurs pyroélectriques permettent de mesurer un changement de polarisation qui est induit par le rayonnement infrarouge. Les détecteurs bolométriques permettent de mesurer un changement de la résistance électrique qui est induit par le rayonnement infrarouge (le dioxide de Vanadium est un exemple de matériau utilisé pour réaliser un détecteur bolométrique). Le principal avantage des détecteurs

FIGURE 3.2 – Exemple de caméra infrarouge refroidie FLIR A3500sc.

thermiques par rapport aux détecteurs à photons est qu'il n'est pas nécessaire de refroidir le détecteur. De plus, la conception de ce type de caméra est en général plus simple. En conséquence, les caméras utilisant ce type de détecteur (que l'on appel aussi : caméras infrarouges non-refroidies) sont en général moins chères, plus petites et consomment moins d'énergie. Ces caméras sont donc idéales pour la robotique mobile. Bien que longtemps mal considérées, le développement récent de détecteurs thermiques avec une résolution très large a permis de montrer qu'il était possible d'obtenir des images infrarouges de bonne qualité à une fréquence d'acquisition vidéo de 30Hz et plus [Rogalski 2002][Flir 2015]. Le développement de ce type de détecteur laisse envisager une grande démocratisation des caméras infrarouges dans les prochaines années.

FIGURE 3.3 – Exemple de caméra infrarouge non-refroidie FLIR tau 2.

3.2 Analyse simultanée du spectre visible et infrarouge

L'analyse d'un seul domaine spectral présente certaines limitations. Alors que, combiner plusieurs domaines spectraux complémentaires permet d'étendre le domaine d'utilisabilité du système de détection et ainsi d'améliorer ses performances.

3.2.1 Avantages et inconvénients

Dans cette section, les avantages et les inconvénients de chaque domaine spectral sont discutés. La combinaison des deux domaines spectraux est discutée dans

un deuxième temps.

3.2.1.1 Spectre visible

Une grand richesse d'information est accessible en analysant le spectre visible ; il est possible d'extraire de la scène l'information : de gradient, de contraste, de couleur, etc. Toutes ces données permettent une analyse fine de la scène comme nous avons pu le voir dans les chapitres précédents. Cependant, toute cette richesse d'information n'est accessible que si il y a suffisamment de luminosité dans la scène. Une caméra opérant dans le spectre visible devient désuète quand il n' a pas de luminosité. Une faible luminosité résulte en une information appauvrie : le contraste dans l'image est moins grand, la palette de couleur des pixels est restreinte, etc. Le manquement de luminosité a un impact négatif direct sur le bon fonctionnement d'algorithmes complexes d'analyses de scène, tels que les détecteurs de personnes.

3.2.1.2 Spectre infrarouge

Le rayonnement infrarouge est plus stable dans le temps que rayonnement visible. En effet, les sources de rayonnement infrarouge sont beaucoup moins impactées par la luminosité : une source de rayonnement infrarouge émettra quelle que soit la luminosité, c'est-à-dire jour et nuit. Cependant, les caméras infrarouge peuvent avoir certaines limitations techniques ; elles peuvent saturer si le rayonnement infrarouge est trop fort. De plus, si deux sources de rayonnement infrarouge se confondent dans la scène alors il devient difficile de les dissocier. L'habillement des personnes peut également avoir un impact négatif sur la détection de personnes dans le spectre infrarouge. À noter que le rayonnement infrarouge est réfléchi au contact du verre.

3.2.1.3 Combiner les spectres visible et infrarouge

Analyser le domaine spectral visible et infrarouge simultanément revient à étendre le domaine spectral du système de vision (Tab.3.1). Ainsi, lorsque les deux modalités ont un fonctionnement fiable, une plus grande richesse d'information est accessible. Lorsqu'une des deux modalités est défectueuse, le domaine spectral du système stéréoscopique est restreint au domaine spectral de la modalité en bonne état de fonctionnement (Tab.3.2).

Les caractéristiques visuelles extraites dans le spectre visible sont complémentaires de celles extraites dans le spectre infrarouge. L'être humain est une source constante de chaleur ; en toute logique, les caractéristiques visuelles basées sur le contraste de température de la scène aident donc à la détection de personnes dans la scène. Le contraste de température peut être suffisant pour détecter une personne ;

cependant, il s'agit d'une information de bas niveau qui peut être facilement parasitée. À contrario, la richesse des caractéristiques visuelles du spectre visible permet de rechercher des motifs d'objet très complexes, avec plus de précision. Les caractéristiques visuelles extraites dans le spectre visible ne sont pas dépendantes de la biologie du corps humain. Cependant, cette grande richesse alourdit l'analyse de la scène.

Luminosité	Forte	Normale	Faible	Nuit
Caméra visible	+++	++++	++	
Caméra infrarouge	+	++	+++	++++
Combinaison	++	+++	+++	++

TABLE 3.2 – Fiabilité des systèmes de vision en fonction de la luminosité.

3.2.2 Les différents systèmes de vision bi-modaux.

Afin de capter simultanément les spectres visible et infrarouge d'une scène, il est nécessaire que les caméras partagent le même champ de vue.

Deux configurations sont généralement envisagées pour permettre cela : 1) la première consiste à aligner les axes optiques des deux caméras et 2) la deuxième consiste à disposer les axes optiques de manière à ce qu'ils soient coplanaires [St-Laurent 2012][Hartley 2004].

(1) Deux approches existent pour aligner les axes optiques. La première nécessite l'utilisation d'un miroir au germanium qui réfléchit les ondes visibles et laisse traverser les ondes infrarouges [St-Laurent 2012]. Ainsi, grâce à ce miroir, il est possible de virtuellement aligner les deux axes optiques en plaçant de chaque côté du miroir les deux caméras C1 et C2 (Fig.3.4). La taille du miroir doit être adaptée en fonction du champ de vue des caméras, ce qui peut poser un problème d'encombrement. À noter que le miroir absorbe une partie des ondes visibles et réfléchit une partie des ondes infrarouges, ce qui nous éloigne d'une exploitation maximale des capteurs.

La deuxième approche consiste à utiliser un optique catadioptrique pour aligner les deux modalités (Fig.3.5.1). Fig.3.5.2 montre un exemple de système de vision conçu avec cette approche [Bergeron 2004]. Avec cette approche, le champ de vue est très réduit et la zone proche du centre optique n'est pas visible, ce qui restreint considérablement son utilisabilité.

Quelles que soient les approches, l'alignement des deux axes optiques est techniquement très complexe à réaliser.

(2) La deuxième configuration est un cas particulier des systèmes de vision stéréoscopiques [Hartley 2004]. Les deux axes optiques sont coplanaires, et orientés

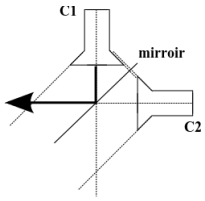
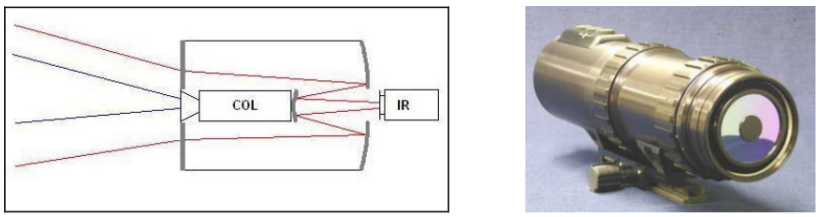


FIGURE 3.4 – Caméras visible C1 et infrarouge C2 ayant leurs axes optiques alignés grâce à l'utilisation d'un miroir au germanium.



(1) (2)

FIGURE 3.5 – Optique catadioptrique pour aligner les modalités visible et infrarouge et exemple de caméra conçue avec cette approche.

vers la scène. Faire légèrement converger les axes optiques permet d'aligner les deux modalités à une distance donnée. Cela peut s'avérer utile si la distance entre les objets de la scène et le système de vision est connue par avance. Dans le cas où la distance entre les objets et le système n'est pas connue, il est préférable que les axes optiques soient parallèles (Fig.3.6) ; cela revient à faire l'hypothèse que l'alignement des objets de la scène a lieu à l'infini. Dans la suite de cette thèse, nous appellerons le système de vision où les axes optiques des caméras visible et infrarouge sont parallèles : système stéréoscopique hétérogène. À noter qu'il est préférable d'utiliser des caméras ayant des champs de vue comparables pour profiter au maximum du champ de vue commun de celles-ci.

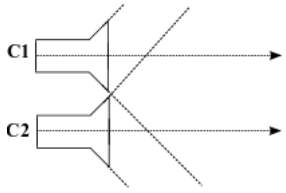


FIGURE 3.6 – Caméras visible C1 et infrarouge C2 ayant leurs axes optiques parallèles pour former un système stéréoscopique hétérogène.

Cette configuration est moins contraignante que la configuration précédente notamment sur le choix des caméras à utiliser. De plus, cette configuration est plus facilement réalisable. Dans la suite de cette thèse, nous avons opté pour la réalisation d'un système stéréoscopique hétérogène.

3.3 Notre système de vision

Notre système stéréoscopique hétérogène est constitué d'une caméra visible GoPro 3 et d'une caméra infrarouge Flir Tau 2 (Fig.3.7). Les centres optiques des deux caméras sont à 5cm l'un de l'autre. Les propriétés des deux caméras sont détaillées dans la suite de cette section.

FIGURE 3.7 – Système stéréoscopique composé de la caméra GoPro 3 et de la caméra Flir Tau 2.

Caméra GoPro 3 hd silver

- Résolution maximale : 1080p (1920x1080)
- Modes possibles : 1080p (champ de vue : 160°), 960p (1280x960, champ de vue : 126°) et 720p (1280x720, champ de vue : 90°)
- Fréquence d'acquisition maximale : 60Hz
- Capteur : CMOS
- Lentille d'origine : 16mm
- Sortie analogique : 640x480 (30Hz)
- Alimentation : Par batterie

Nous avons changé la lentille dans le but de se rapprocher du champ de vue de la caméra infrarouge et de supprimer la distorsion en barrilet de la caméra. La nouvelle lentille est une lentille restreint le champ de vue à 36 degrés.

Caméra Flir Tau 2

- Résolution maximale : 640x480
- Type d'infrarouge détecté : infrarouge long et lointain
- Fréquence d'acquisition : 7.5 Hz (x4 à la sortie)
- Format de sortie : NTSC 30Hz et PAL 25Hz
- Lentille : 19mm (champs : 32 degrés en largeur et 26 degrés en hauteur)
- Type de caméra infrarouge : non-refroidie (détecteur thermique haute résolution)

– Alimentation : Extérieure 5V DC

La fréquence de sortie de la caméra infrarouge est de 30Hz mais la fréquence d'acquisition est de 7,5Hz : les images sont dupliquées par groupe de quatre avant la sortie vidéo. Notre modèle de caméra a une limitation matérielle qui ne permet pas de l'utiliser avec une fréquence d'acquisition de 30Hz. Pour profiter au maximum de cette caméra il est nécessaire de demander une autorisation spéciale aux autorités militaires du pays exportateur, les États-Unis d'Amérique.

3.3.1 La géométrie du système stéréoscopique

Physiquement, les deux caméras sont liées rigidement et forment un système stéréoscopique hétérogène tel que montré Fig.3.8. Les deux caméras sont positionnées dans un même plan, avec leurs axes optiques quasiment parallèles ; elles sont orientées dans le même sens.

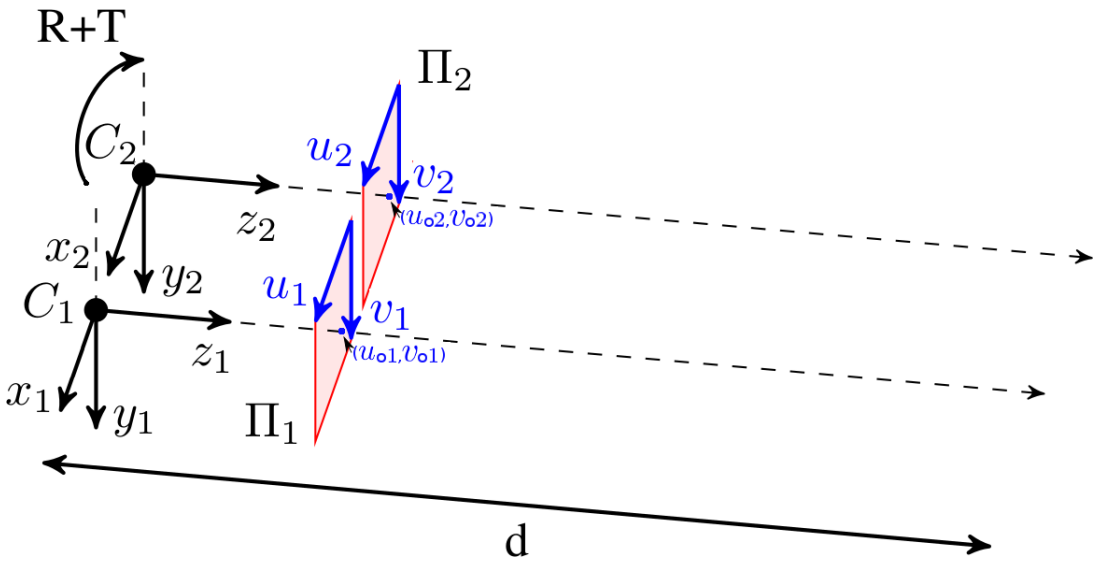


FIGURE 3.8 – Schéma du système de vision stéréoscopique hétérogène, muni d'une caméra visible (C_1) et d'une caméra infrarouge (C_2).

C_1 et C_2 sont respectivement les centres optiques de la caméra visible et la caméra infrarouge (Fig.3.8). Π_1 et Π_2 sont les plans images. P correspond à l'emplacement arbitraire d'une personne à détecter devant le système stéréoscopique. R correspond à la rotation de la caméra de C_1 par rapport à la caméra C_2, et T correspond à la translation de la caméra C_1 par rapport à la caméra C_2. La distance entre le système stéréoscopique et l'emplacement P est définit par d.

3.3.2 La synchronisation des caméras

Il est nécessaire que l'on puisse trouver au même instant les mêmes motifs humains situés aux mêmes endroits des images infrarouge et visible. Cela nécessite une étape de synchronisation des caméras. Les caméras doivent être synchronisées temporellement - et - spatialement. Il peut donc parfois être difficile de savoir si un décalage entre les motifs du spectre infrarouge et ceux du spectre visible est du à un problème de synchronisation temporelle ou un problème de synchronisation spatiale. Il est recommandé de commencer par synchroniser les caméras temporellement pour faciliter la synchronisation spatiale.

3.3.2.1 La synchronisation temporelle

Le décalage temporel des images est la conséquence d'une différence des fréquences d'acquisition des caméras et du moment où débute l'acquisition sur chaque caméra. Ce décalage temporel peut entraîner un décalage spatial si le système de vision et/ou les personnes sont en mouvement.

Par exemple, une personne peut ne pas se trouver au même endroit de l'image suivant que l'on analyse la modalité visible ou la modalité infrarouge si l'individu est en mouvement et que la synchronisation temporelle n'est pas parfaite. Fig.3.9.1 illustre le cas où la synchronisation n'est pas parfaite et la personne est en mouvement. Fig.3.9.2 illustre le cas où la synchronisation temporelle est parfaite.

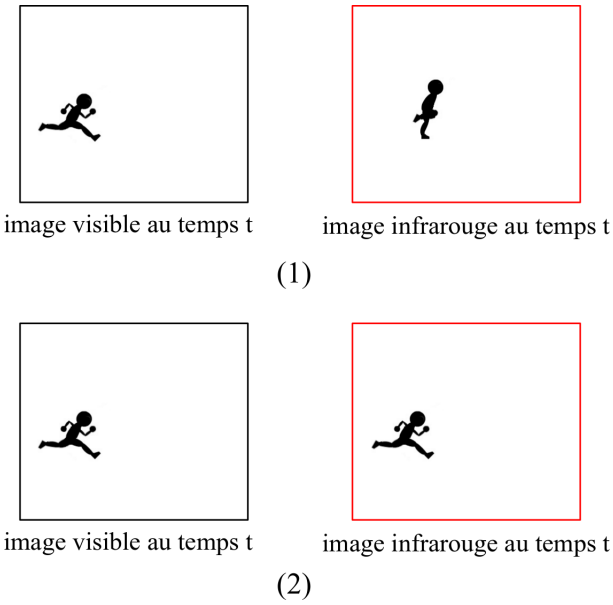


FIGURE 3.9 – Exemple d'une mauvaise et d'une bonne synchronisation temporelle de l'infrarouge et du visible.

Il y a plusieurs approches pour corriger le décalage temporel lorsque les deux

chaînes d'acquisition vidéo sont différentes telles que : 1) utiliser la sortie "trigger" d'une des caméras pour piloter l'acquisition d'images de l'autre caméra (maître/esclave) ou encore 2) utiliser un "top" visuel de référence généré à la sortie des signaux pour mettre en correspondance les images dans un second temps. La solution (1) peut avoir une incidence négative sur le temps d'exposition de la caméra esclave. La solution (2) nécessite l'utilisation d'une référence commune aux deux caméras.

Dans notre cas, nous avons obtenu une très bonne synchronisation temporelle en construisant deux chaînes d'acquisition similaires : les flux vidéo sont récupérés via les sorties analogiques des caméras et les flux vidéo sont ensuite convertis en signaux numériques via deux convertisseurs vidéo analogique numérique identiques. Avec cette approche, le décalage temporel est au maximum d'une image. Nous avons obtenu des performances de synchronisation optimales en branchant les deux convertisseurs sur deux ports USB qui ne partagent pas le même bus de transfert de données afin d'éviter les conflits. Le convertisseur vidéo analogique numérique est le Terratec Grabby Rev 2.

La duplication des images infrarouge est un autre aspect important à prendre en compte lors de l'analyse des paires d'images. Les images infrarouge sont dupliquées quatre fois pour atteindre une fréquence finale de sortie de 30Hz. Par conséquent, seule la première image de chaque groupe de quatre doit être traitée (avec l'image visible correspondant) comme montré dans Fig.3.10.

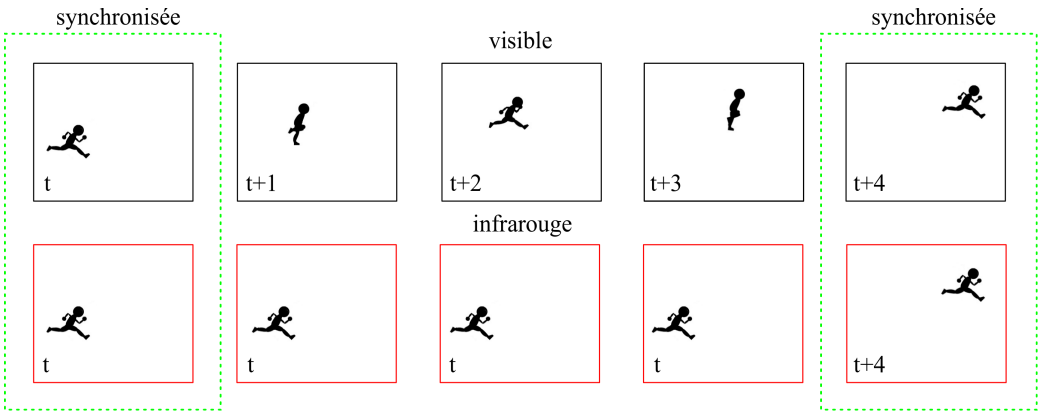


FIGURE 3.10 – Sélection des bonnes paires d'images infrarouge et visible après la synchronisation temporelle des caméras.

3.3.2.2 La synchronisation spatiale

La taille apparente des objets dans une image varie selon les paramètres intrinsèques des caméras, c'est-à-dire, la distance focale, les facteurs d'échelle de l'image et les coordonnées du centre optique. Le projection des objets de la scène

dans les images est fonction de la transformation rigide qui existe entre les deux caméras, c'est-à-dire, la position et l'orientation d'une caméra par rapport à l'autre. Il faut compenser ces variations pour synchroniser spatialement les images.

La synchronisation spatiale, tout comme la synchronisation temporelle, est une étape préliminaire importante. Une mauvaise synchronisation spatiale ne permettra pas une bonne mise en correspondance des détections obtenues dans la modalité visible et dans la modalité infrarouge. Une paire d'images est bien synchronisée spatialement si il y a superposition quasi-parfaite entre l'image prise dans le visible et l'image prise dans l'infrarouge.

D'après Krotosky et al, il existe plusieurs manières de synchroniser un système stéréoscopique hétérogène [Krotosky 2007], c'est-à-dire : 1) par un recalage global des images visible et infrarouge ne permettant une synchronisation parfaite des objets qu'à une profondeur particulière de la scène, 2) en utilisant, en plus du système stéréoscopique hétérogène, un système permettant d'obtenir l'information de profondeur de la scène pour reprojeter chaque objet dans l'autre modalité en fonction de sa profondeur dans la scène, 3) en sélectionnant des régions d'intérêt dans la modalité visible et la modalité infrarouge puis en les mettant en correspondance par des transformations homographiques locales permettant ainsi de recaler les objets de la scène pour plusieurs profondeurs ou 4) en utilisant une transformation homographique à l'infini, permettant de recaler les objets de la scène qui sont à l'infini par rapport au système de vision.

Notre choix s'est porté sur la transformation homographique à l'infini pour quatre raisons : nous ne pouvons faire aucune hypothèse sur le positionnement des personnes dans la scène, nous ne disposons pas d'un système matériel pour obtenir l'information de profondeur, nos sujets sont éloignés du système de vision et nous avons souhaité limiter au maximum les temps de calcul nécessaire à la synchronisation.

La transformation homographique à l'infini nécessite de faire l'hypothèse que les personnes de la scène sont à l'infini par rapport au système de vision. Pour cela, la distance entre les deux caméras ("baseline", en anglais) doit être très inférieure à la distance qu'il y a entre le système stéréoscopique et les personnes de la scène ($d <<< T$, dans Fig.3.8) [Krotosky 2007].

Plusieurs paramètres sont nécessaire au calcul de la transformation homographique à l'infini, dont :

$$R = Rotation(\gamma/z) \times Rotation(\beta/y) \times Rotation(\alpha/x) \tag{3.4}$$

R est la matrice de rotation entre la caméra C_1 et la caméra C_2, elle est le produit de la rotation γ autour de l'axe z, de la rotation β autour de l'axe y et de la rotation α autour de l'axe x (Fig.3.7). Où le repère (x,y,z) est le repère monde.

$$K_1 = \begin{bmatrix} k_{1u} & 0 & u_{o1} \\ 0 & k_{1v} & v_{o1} \\ 0 & 0 & 1 \end{bmatrix} K_2 = \begin{bmatrix} k_{2u} & 0 & u_{o2} \\ 0 & k_{2v} & v_{o2} \\ 0 & 0 & 1 \end{bmatrix} \tag{3.5}$$

K_1 et K_2 sont respectivement les matrices des paramètres intrinsèques des caméras 1 et 2 (Fig.3.8). Les points principaux des caméras 1 et 2 sont respectivement (u_{o1}, v_{o1}) et (u_{o2}, v_{o2}). Les paramètres k_u et k_v sont respectivement le facteur d'échelle horizontal et vertical.

L'homographie infinie est calculée comme suit :

$$H_\infty = K_2 \times R \times K_1^{-1} \tag{3.6}$$

Chaque pixel est re-projeté comme décrit ci-dessous :

$$\begin{pmatrix} u_2 \\ v_2 \\ w_2 \end{pmatrix} = H_\infty \times \begin{pmatrix} u_1 \\ v_1 \\ 1 \end{pmatrix} \tag{3.7}$$

Les matrices K_1 et K_2 peuvent être respectivement obtenues grâce à la calibration de la caméra 1 et de la caméra 2. La matrice de rotation R est obtenue suite à la calibration des paramètres extrinsèques des deux caméras. Il existe beaucoup de méthodes de calibration, nous avons utilisé la "toolbox" Matlab de Jean-Yves Bouguet pour la calibration des paramètres intrinsèques et extrinsèques des deux caméras par un calibrage multi-plan avec une mire [Bouguet]. À noter qu'il est également possible de calculer l'homographie infinie en utilisant deux rectangles de référence positionnés et orientés différemment dans l'espace [Kim 2006]. L'avantage de cette méthode est qu'il n'est pas nécessaire de calibrer les caméras. Cependant, nous avons trouvé que cette approche ne donnait pas une estimation très précise de l'homographie infinie. Nous pensons que cela est due à la nature hétérogène du système stéréoscopique ; les coins des rectangles sont localisés avec moins de précision dans l'infrarouge.

La Fig.3.11 montre un échantillon des images qui ont été utilisées pour calibrer les caméras visible et infrarouge. La calibration de la caméra infrarouge a nécessité la chauffe de la mire pour produire un contraste entre les carreaux blancs et noirs du damier. La mire fut chauffée par deux projecteurs hallogènes disposés à proximité. Nous avons constaté que le meilleur contraste était obtenu après environ 20 secondes de chauffe pour une courte durée.

Des exemples de paires d'images synchronisées sont données Fig.3.12.

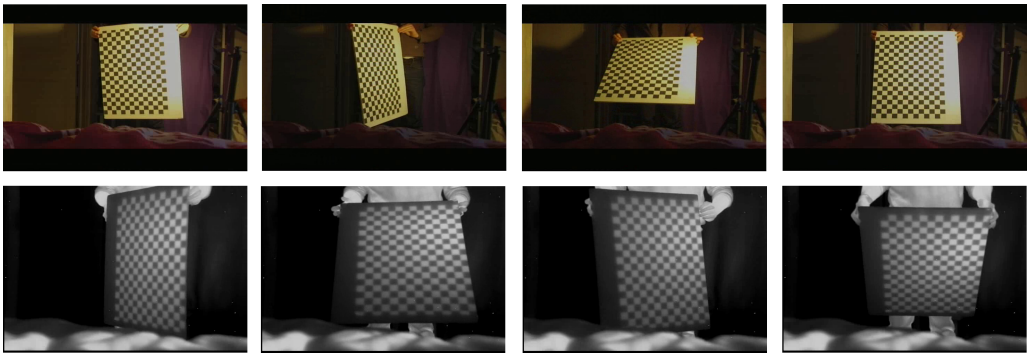


FIGURE 3.11 – Exemples d'images utilisées pour la calibration des caméras visible et infrarouge.

3.4 État de l'art

Dans l'imaginaire collectif, la caméra infrarouge est l'outil idéal pour détecter les personnes, que cela soit de manière automatique (utilisant un détecteur de personnes) ou non (par la supervision d'un opérateur) ; il suffirait donc d'utiliser une caméra infrarouge pour détecter les personnes dans toutes les situations. En pratique, cela est plus compliqué (Tab.3.2) ; on observe cependant que les caméras visibles et les caméras infrarouges sont complémentaires (Tab.3.2). Une étude approfondie des approches collaboratives permettrait de profiter au maximum de cette complémentarité, qui peut être exploitée aussi bien à l'entraînement qu'à la détection.

Dans cette section, nous passerons en revue différentes approches de détection de personnes dans le spectre infrarouge seul. Puis, nous nous intéresserons aux approches collaboratives de détection de personnes, et pour finir, nous nous intéresserons aux approches collaboratives d'apprentissage.

3.4.1 Détection de personnes dans le spectre infrarouge.

Un certain nombre de travaux a vu le jour concernant la détection automatique de personnes dans le spectre infrarouge. Ces travaux émergèrent parallèlement à la démocratisation progressive des caméras infrarouges de grandes résolutions (320x240 pixels de résolution dans un premier temps, puis, plus récemment, 640x480 pixels de résolution) et à haute fréquence d'acquisition. Presque tous ces travaux concernent la détection de piétons pour les systèmes ADAS. On remarque cependant qu'il existe un faible nombre de travaux concernant la détection dans l'infrarouge, en comparaison à ceux existant dans le visible. Cela peut être expliqué par deux choses : 1) le prix des caméras infrarouges est encore trop important, 2) la résolution des caméras infrarouges est encore trop faible.

FIGURE 3.12 – Exemples de paires d'images visible infrarouge fusionnées obtenues après notre synchronisation spatiale. Nous observons que les objets suffisamment éloignés du système de vision sont bien synchronisés.

Concernant les performances de détection, il n'existe pas d'étude comparant la détection de personnes dans le visible à la détection de personnes dans l'infrarouge.

La suite de cette section expose différents travaux majeurs concernant la détection de personnes dans le spectre infrarouge.

Détection de personnes dans l'infrarouge basée sur les caractéristiques de forme locale.

En 2007, Zhang et al proposèrent un des premiers travaux sur la détection de personnes dans l'infrarouge basé sur un apprentissage supervisé . Les auteurs utilisent le fait que la silhouette des personnes est similaire dans les deux modalités [Zhang 2007b]. Les auteurs testèrent les mêmes méthodes que celles employées pour la détection de personnes dans le visible. Ils observèrent que les méthodes employées pour la modalité visible s'appliquent également très bien pour la modalité infrarouge. Cependant, les performances de celles-ci diffèrent. Dans leur étude, plusieurs combinaisons de caractéristiques visuelles et de classifieurs ont été testées pour en déduire la meilleure configuration. Les caractéristiques visuelles qui ont été testées sont : les histogrammes de gradients orientés (HOG)s et les mor-

ceaux de contours (en anglais : "edgelets"). Deux classifieurs différents ont été testés : le classifieur SVM et le classifieur AdaBoost (entraînés sur des images infrarouges). Ils montrèrent que les meilleures performances sont obtenues avec le détecteur HOG/SVM.

La principale conclusion de leur travaux est qu'il est possible d'obtenir des performances de détection dans l'infrarouge qui sont comparables à celles obtenues dans le visible en appliquant les techniques de détection utilisées dans le visible. Cependant, la combinaison optimale de caractéristiques visuelles et de classifieurs n'est pas forcément la même que dans le visible. Ces travaux suggèrent qu'il n'est pas nécessaire d'inventer des méthodes radicalement différentes pour la détection de personnes dans l'infrarouge. À noter qu'en 2012, Konigs et al effectuèrent une étude dont les conclusions montrèrent que la détection supervisée est plus performante que la segmentation classique pour détecter les personnes dans l'infrarouge en milieu extérieur [Konigs 2012].

Caractéristiques visuelles invariantes au contraste pour la détection dans l'infrarouge.

Olmeda et al proposèrent un nouveau type de caractéristiques visuelles invariantes au contraste de température ; ces caractéristiques sont utilisées pour une détection de personnes plus robuste dans le spectre infrarouge long/lointain [Olmeda 2012b]. En effet, les auteurs constatèrent que le contraste de température entre les personnes et le fond de la scène diffèrent selon les conditions d'éclairage et la température ambiante. De plus, avec certaines caméras infrarouges et/ou avec certains réglages le contraste est dynamiquement adapté en fonction des objets présents dans la scène. C'est-à-dire, si un objet plus chaud entre dans la scène, le dégradé de niveau de gris va être dynamiquement adapté pour prendre en compte le nouvel objet. Tout cela a pour conséquence de rendre les caractéristiques visuelles basées sur une analyse des gradients moins robuste dans l'infrarouge. Les auteurs proposèrent donc de nouvelles caractéristiques visuelles basées sur la théorie de la congruence de phase [Kovesi 2000].

La théorie de la congruence de phase permet une invariance aux changements d'illumination et aux changements d'échelle. Olmeda et al s'inspirèrent des histogrammes de gradients orientés pour proposer les histogrammes d'énergies de phase orientés ("Histogram of Oriented Phase Energy", en anglais ou "HOPE [1]"). Les caractéristiques HOPE sont calculées dans des cellules réparties sur toute la fenêtre d'analyse. Les cellules sont groupées en blocs et les blocs sont normalisés, tout comme pour les caractéristiques HOG [Olmeda 2012b].

1. *Histogram of Oriented Phase Energy*, Histogramme d'orientation des énergies de phase

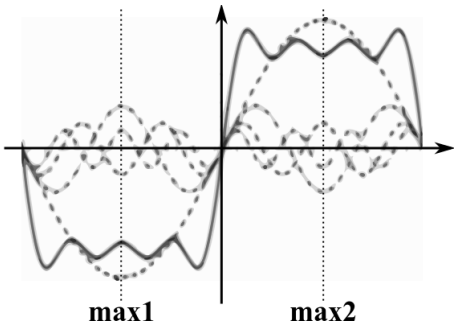


FIGURE 3.13 – Décomposition en composantes de fourier (pointillés) d'un signal (en trait plein), la congruence de phase est maximale en $max1$ et $max2$.

– **Congruence de phase**

Les maximaux de congruence de phase se situent là où un grand nombre de composantes de Fourier sont en phases [Olmeda 2012b] ($max1$ et $max2$ Fig.3.13). Les coins et les contours sont les endroits de l'image où les composantes de Fourrier sont en phase au maximum. La congruence de phase est plus robuste aux changements de contraste car elle est moins affectée par celui-ci [Kovesi 2000].
Morrows et Owens définirent en 1987 la congruence de phase PC d'un signal à une position x comme suit [Kovesi 2000] :

$$PC(x) = max_{\Phi'(x)\in[0,2\pi]} \frac{\sum_n A_n cos(\Phi_n(x) - \Phi'(x))}{\sum_n A_n} \tag{3.8}$$

Considérons que le signal est composé de n composantes de Fourier. A_n et $\Phi_n(x)$ représentent respectivement l'amplitude et la phase en x de la ni-ème composante de Fourrier. $\Phi'(x)$ est la valeur qui maximise l'équation en x, il s'agit de la moyenne des n angles de phase pondérés par leurs amplitudes respectives. En 2000, Kovesi et al observèrent que la congruence de phase proposée par Morrows et Owens (Equ.3.8) appliquée au calcul de caractéristiques est de mauvaise qualité pour les contours flous [Kovesi 2000]. Ils proposèrent donc une nouvelle formalisation de la congruence de phase $PC'(x)$ qui a la propriété d'être plus précise pour la vision par ordinateur :

$$PC'(x) = \frac{\sum_n W(x)\lfloor A_n(x)\Delta\Phi_n(x) - T\rfloor}{\sum_n A_n(x) + \varepsilon} \tag{3.9}$$

$$\Delta\Phi(x) = cos(\Phi_n(x) - \Phi'(x)) - |sin(\Phi_n(x) - \Phi'(x))| \tag{3.10}$$

$A_n(x)$ représente l'amplitude en x du signal, T est une estimation de l'influence du bruit, ε est une petite valeur pour éviter une division par défaut et $W(x)$ est une fonction sigmoide de pondération [Kovesi 2000]. À noter que la notation mathématique $\lfloor x \rfloor$ signifie que x est égale à lui même si x est positif, ou égale à 0 si x est négatif.
La congruence de phase peut être calculée par convolution de filtres Gaussiens logarithmiques et en utilisant la définition de la congruence de phase en deux dimensions :

$$PC_2(x,y) = \frac{\sum_o \sum_n W_o(x,y) \lfloor A_{no}(x,y)\Delta\Phi_{no}(x,y) - T \rfloor}{\sum_o \sum_n A_{no}(x,y) + \varepsilon} \tag{3.11}$$

De meilleures performances de détection dans l'infrarouge sont obtenues en utilisant les histogrammes de congruence de phase orientés, appelés également histogrammes d'énergie de phase orientés (ou HOPE, en anglais) [Olmeda 2012b]. Pour construire un histogramme de phase orientés le processus est le même que pour construire un histogramme de gradients orientés, à la différence que les secteurs sont remplis en fonction de l'angle de phase. De lourds calculs sont nécessaires à la construction d'un HOPE (analyse fréquentielle, nombreux calculs de sommes, etc.) ; pour le moment, les caractéristiques HOPE ne sont pas un bon choix pour le temps réel. Lorsque les temps de calcul sont importants, il est préférable d'utiliser des caractéristiques plus traditionnelles (basées sur une analyse de contour par exemple). Les améliorations apportées par les HOPE semblent peu importantes au vue de l'explosion des temps de calcul [Olmeda 2013].

Détection de personnes dans l'infrarouge en utilisant les caractéristiques de canaux agrégés

En 2014, Brehar et al proposérent une approche de détection de personnes dans l'infrarouge en deux étapes : 1) dans un premier temps des régions d'intérêt sont extraites en utilisant l'information de contour et d'intensité sur l'infrarouge, 2) dans un deuxième temps les régions d'intérêt sont analysées par un dérivé du détecteur de personnes ACF/SoftCascade pour l'infrarouge (que l'on appelera IR-ACF [2]/SoftCascade) [Brehar 2014].

(1) Lors de la phase d'extraction des régions d'intérêt, seules les contours verticaux sont gardés, afin de supprimer les parties de l'image telles que les voitures, les bâtiments, la route, etc. Plusieurs opérations morphologiques de fermeture sont effectuées dans le but d'obtenir des régions homogènes. Cette approche n'est pas adaptée dans le cas où l'orientation de la caméra n'est plus compatible avec une

2. *Infrared Aggregate Channel Features*, Caractéristiques de canaux agrégés pour l'infrarouge

vue piétonne. Si l'angle de roulis est trop important, la suppression de toute information autre que verticale supprimera aussi les personnes de la scène ; cela n'est pas adapté à une détection de personnes en vue aérienne.

(2) Le détecteur IR-ACF/SoftCascade utilise des caractéristiques visuelles légèrement différentes des caractéristiques de canaux agrégés classiques. À la différence des caractéristiques ACF, seuls huit canaux d'images sont traités. Les trois canaux de couleur L, U et V sont abandonnés au profit du canal niveau de gris normalisé [Brehar 2014]. Les auteurs obtiennent des performances intéressantes malgré la simplicité de l'approche (30Hz avec les caractéristiques IR-ACF approximées et la réduction d'espace de recherche pour un cas d'utilisation simple).

Utilisation d'une représentation éparse pour la détection de personnes dans l'infrarouge

En 2014, Qi et al proposèrent d'utiliser un classifieur basé sur une représentation éparse des données : le classifieur K-SVD [3], utilisant deux dictionnaires [Qi 2014]. La représentation éparse présente l'avantage d'être plus robuste aux données corrompues [Elhamifar 2011] ; cela permet donc une plus grande robustesse au bruit et/ou aux variations d'apparences humaines dans l'infrarouge.

À la détection, deux distances sont calculées : une distance entre les caractéristiques de l'image d'entrée et le dictionnaire "classe fond" et une distance entre les caractéristiques de l'image d'entrée et le dictionnaire "classe humain". La distance la plus petite indique la classe de l'image d'entrée.

Ils testèrent l'approche avec les caractéristiques HOG, les caractéristiques HOPE et avec les caractéristiques de mots éparses (ou "HSC [4]", voir [Qi 2014]). Les performances du HOG/ K-SVD et du HOPE/ K-SVD furent comparées aux performances du HOG/SVM et du HOG/K-SVM, entre autre. Tous les détecteurs étant évidemment entraînés avec des images d'entraînement infrarouge. Qi et al obtinrent les meilleurs performances avec le détecteur HOG/K-SVD.

Bien que cette approche de détection soit intéressante, il semblerait qu'elle soit beaucoup moins rapide que l'approche IR-ACF/SoftCascade. En effet, deux distances doivent être calculées, une pour le fond et une pour la classe humain. De plus, les caractéristiques HOPE testées ne sont pas les plus rapides de l'état de l'art.

Caractéristiques calculées autour de points d'intérêt pour la détection rapide dans l'infrarouge

3. *K-Singular Value Decomposition*, k-Décomposition en Valeur Singulière
4. *Histogram of Sparse Code*, Histogramme de Code Éparse

Olmeda et al proposèrent en 2012 une approche basée sur le calcul de caractéristiques calculées autour de points d'intérêt pour la détection rapide de personnes dans l'infrarouge [Olmeda 2012a]. L'approche consiste à (1) réduire l'espace de recherche par une extraction de régions d'intérêt et (2) à traiter les régions d'intérêt pour détecter les individus de la scène.

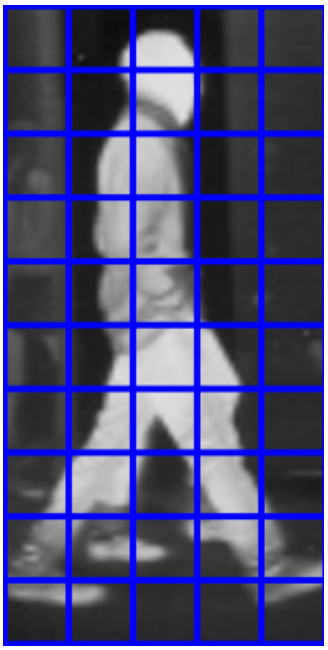


FIGURE 3.14 – Répartition des 50 blocs dans une image d'entraînement positive.

(1) Des points d'intérêt sont cherchés dans la scène dans le but de trouver des régions d'intérêt : autour de chaque point d'intérêt est calculé un ensemble de 5x10 blocs HOPE (chaque bloc contient 3x3 cellules et les blocs ne se chevauchent pas, voir Fig.3.14). Pour chacun des 5x10 blocs de l'ensemble, un classifieur a été entraîné. Lors de l'étape de recherche de régions d'intérêt, chacun des 5x10 blocs est évalué par les 5x10 classifieurs (50x50 évalutations au total). Parmi toutes les évaluations, la classification ayant le meilleur résultat sert de référence pour créer une région d'intérêt. Cela est fait en respectant la position d'origine du classifieur obtenant la meilleure évaluation, ainsi si le meilleur résultat est obtenu pour le bloc (5,5) avec le classifier entraîné sur le bloc (2,1) alors la région d'intérêt générée sera translatée de 3 blocs vers la gauche et de 4 blocs vers le bas par rapport aux blocs calculés autour du point d'intérêt.

(2) Les régions d'intérêt sont évaluées par un classifieur entraîné pour 10x5 blocs HOPE dans le but de confirmer ou d'infirmer la présence ou la non-présence de personnes dans ces régions.

L'approche est rapide car elle combine une réduction de l'espace de recherche à la classification. Grâce à cela, un plus faible nombre d'évaluations de classifieur

est effectué. De plus, les caractéristiques visuelles ne sont pas recalculées pour plusieurs niveaux de pyramides d'échelle (car il n'y a pas de pyramide d'images). À noter que calculer les caractéristiques sur plusieurs niveaux demande beaucoup plus de temps de calcul (d'où la rapidité de l'approche présentée) mais, cela permet un calcul plus exact des caractéristiques [Dollár 2009b].

3.4.2 Approches collaboratives pour la détection

Durant la phase de détection, la complémentarité des deux modalités du système stéréoscopique peut être exploitée dans le but d'avoir accès à une plus grande richesse d'information et d'accroître les performances de détection du système. En théorie, plus on a de sources d'information différentes (permettant à chaque fois d'avoir une vue différente de la scène) plus on robustifie la détection en la rendant moins dépendante d'une seule source d'information. De plus, la juxtaposition d'indices de présence humaine dans différentes modalités renforce la détection des personnes.

Il est possible d'exploiter la richesse d'information de deux manières : 1) de manière décorrelée, en procédant à une phase de détection dans le visible, puis, dans l'infrarouge et en combinant les résultats obtenus dans les deux modalités ou, 2) de manière liée, en fusionnant l'information visible et l'information infrarouge pour ensuite procéder à une seule phase de détection.

(1) La première approche revient à fusionner les détections obtenues dans le visible et celles obtenues dans l'infrarouge (Fig.3.15).

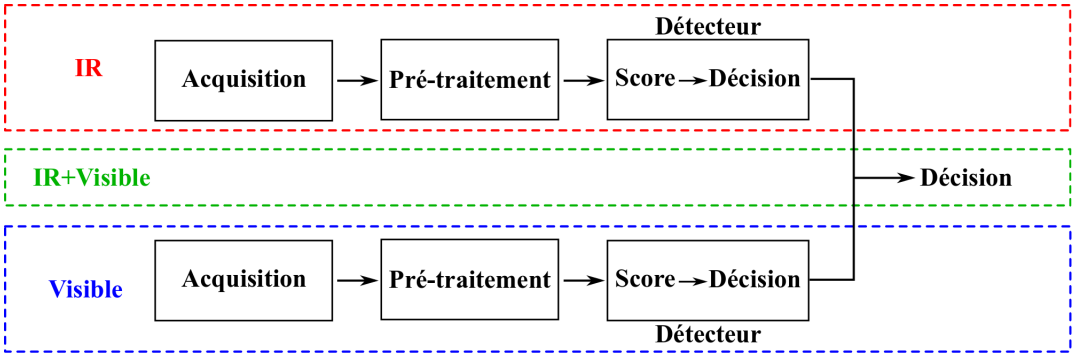


FIGURE 3.15 – Chaîne de traitement de la fusion après la détection.

(2) La deuxième approche consiste à fusionner les modalités visible et infrarouge en amont puis à procéder à la phase de détection (Fig.3.16).

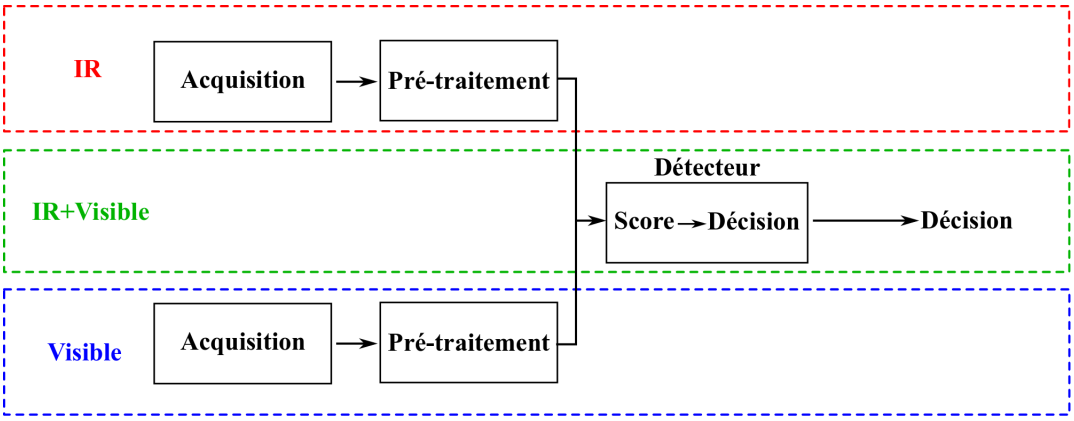


FIGURE 3.16 – Chaîne de traitement de la fusion des modalités avant la détection.

3.4.2.1 Fusion des détections

Étonnamment, il existe relativement peu de travaux concernant la fusion des détections de personnes. Dans cette section est présenté un travail récent de référence sur le sujet.

Fusion de croyances pour la détection de piéton.

En 2014, Xu et al proposèrent une méthode pour combiner un très grand nombre de détecteurs de piétons dans le but d'améliorer les performances de détection [Xu 2014]. Dans leurs travaux le détecteur de piétons est considéré comme étant une boite noire. Ils partirent du principe que chaque détecteur détecte un ensemble différent de personnes (bien qu'il y ait intersection entre les ensembles) et donc que logiquement, en combinant les détecteurs de la bonne façon, il est possible d'augmenter le nombre de détections.

Dans un premier temps, il est nécessaire de calibrer les résultats des détecteurs car ceux-ci sont tous différents. La calibration va consister à transformer les résultats en probabilités (c'est-à-dire en des valeurs comprises entre 0 et 1). Les auteurs testèrent deux méthodes de calibration : (1) une méthode utilisant la regression logistique ainsi qu'une fonction sigmoïde et (2) une méthode de calibration non-paramétrique minimisant l'erreur moyenne quadratique [Xu 2014].

La deuxième étape consiste à fusionner les résultats finaux de tous les détecteurs. Xu et al se basèrent sur la théorie de Dempster-Schafer. Cette théorie utilise des fonctions de masse définies comme suit :

$$\sum_{A \subseteq \Omega} m(A) = 1 \text{ et } m(\emptyset) = 0 \tag{3.12}$$

Où $\Omega = \{\omega_1, \omega_2, ..., \omega_N\}$ avec les ω_n pour $n \in N$ représentant les états des N

détecteurs. Dans le cas de la détection de personnes $\Omega = \{0, 1\}$, autrement dit, il y a seulement deux états pour un détecteur : détection ou non-détection. L'ensemble vide est noté $\emptyset$.

Chacun des détecteurs est associé à une fonction de masse. Les résultats calibrés du détecteur (noté $f(r)$) sont liés à la fonction de masse par les relations suivantes [Xu 2014] :

$$m(\{1\}) = f(r) \text{ et } m(\{0, 1\}) = 1 - f(r) \tag{3.13}$$

Les masses des détecteurs peuvent être combinées entre elles en utilisant la règle de combinaison de Dempster :

$$(m_1 \oplus m_2)(\emptyset) = \frac{1}{1-\kappa} \sum_{B \cap C} m_1(B) m_2(C) \text{ et } (m_1 \oplus m_2)(\emptyset) = 0 \tag{3.14}$$

$$\kappa = \sum_{B \cap C = 0} m_1(B) m_2(C) \tag{3.15}$$

m_1 et m_2 sont deux fonctions de masse différentes, $A \in \Omega$ et $A \neq \emptyset$.

La règle de combinaison de Dempster nécessite cependant qu'il y ait indépendance entre les détecteurs, cela n'est pas forcément le cas, surtout lorsque des détecteurs qui partageant des propriétés communes sont utilisés (le détecteur ICF/SoftCascade et le détecteur ACF/SoftCascade, par exemple). Pour pallier cela, les auteurs proposèrent d'utiliser une règle $t - norm$ optimisée [Xu 2014].

Ces travaux prouvent qu'il est possible d'améliorer considérablement les performances de détection en utilisant la théorie de la croyance pour combiner un grand nombre de détecteurs de personnes différents dans le spectre visible. Cependant, le temps de calcul augmente à chaque ajout d'un détecteur dans le système.

3.4.2.2 Fusion des modalités pour la détection

La fusion de données est un domaine très large, il existe un grand nombre de travaux à ce sujet dans la littérature. Nous nous intéresserons ici essentiellement aux travaux traitant de la fusion du visible et de l'infrarouge pour la détection de personnes.

Détection de personnes basée perception par fusion du visible et de l'infrarouge.

En 2004, Jiang et al proposèrent un des premiers travaux sur la fusion du visible et de l'infrarouge pour aider un opérateur à détecter des personnes [Jiang 2004].

Aucun détecteur supervisé, ou autre méthode, n'est utilisé pour détecter automatiquement les personnes, il s'agit essentiellement d'une méthode d'amélioration de la perception des personnes. À noter que les paires d'images visibles et infrarouges sont synchronisées avant la fusion des données.

Dans un premier temps, le contraste de perception D est calculé pour toute la scène tel qu'il est montré ci-dessous :

$$D(x,y) = \frac{S_{vis}(x,y) - S_{ir}(x,y)}{S_{vis}(x,y) + S_{inf}(x,y)} \tag{3.16}$$

Les valeurs S_{vis} et S_{inf} sont des valeurs de saillance calculées comme suit :

$$S_{vis}(x,y) = |I_{vis}(x,y) - I_{vis\_moyenne}(x,y)| \tag{3.17}$$

$$S_{ir}(x,y) = |I_{ir}(x,y) - I_{ir\_moyenne}(x,y)| \tag{3.18}$$

Où I_{vis} et I_{ir} sont respectivement les images visible et infrarouge. Les valeurs $I_{vis\_moyenne}$ et $I_{ir\_moyenne}$ sont calculées comme suit :

$$I_{vis\_moyenne}(x,y) = \frac{1}{M \times M} \sum_{s=-a}^{a} \sum_{t=-b}^{b} I_{vis}(x+s, y+t) \tag{3.19}$$

$$I_{ir\_moyenne}(x,y) = \frac{1}{M \times M} \sum_{s=-a}^{a} \sum_{t=-b}^{b} I_{ir}(x+s, y+t) \tag{3.20}$$

Où $a = (M-1)/2)$ et $b = (M-1)/2$ et $M \in \{3,5,7,9,11\}$.

Un seuil T (fixé à 0.25) est utilisé pour calculer les poids W_{vis} et W_{ir} qui sont nécessaires à la fusion du visible et de l'infrarouge. Si $D(x,y) > T$ alors $W_{vis}(x,y) = 1$ et $W_{ir}(x,y) = 0$, sinon $W_{vis}(x,y) = (1 - \frac{S_{ir}(x,y)}{S_{vis}(x,y)}) \times 0.5$ et $W_{ir}(x,y) = 1 - W_{vis}(x,y)$.

L'image fusionnée finale est calculée de la manière suivante :

$$I_{fusion}(x,y) = W_{vis}(x,y) I_{vis}(x,y) + W_{ir}(x,y) I_{ir}(x,y) \tag{3.21}$$

Cette fusion des images visible et infrarouge améliore la perception des personnes dans l'environnement. Cependant, l'approche n'est pas adaptée dans le cas où la modalité infrarouge est saturée ou dans le cas où la luminosité n'est pas suffisamment importante pour exploiter la modalité visible.

Pyramide laplacienne pour la détection dans les spectres visible et infrarouge

En 2009, Gilmore et al proposèrent de fusionner le visible et l'infrarouge dans le but d'améliorer les performances finales de détection [Gilmore 2009]. La méthode de fusion utilisée est basée sur la pyramide laplacienne permettant d'obtenir des images fusionnées avec un niveau de détail élevé.

Dans un premier temps, deux pyramides d'images sont créées. La première pyramide d'images est créée en procédant à plusieurs opérations de réduction. L'opération de réduction consiste à combiner sous-échantillonnage et convolution de l'image du niveau précédent :

$$G_{(k+1)}(x,y) = \sum_{u=-p}^{p} \sum_{v=-p}^{p} K(u,v) G_k(2x+u, 2y+v) \tag{3.22}$$

Où G_0 est l'image d'entrée et K est un petit noyau Gaussien (3x3 ou 5x5, c'est-à-dire, p=3 ou 5).

La deuxième pyramide est créée en procédant à plusieurs opérations d'expansion E. Une opération d'expansion consiste à combiner sur-échantillonnage et convolution de l'image G_{k+1} :

$$E_k(x,y) = \sum_{u=-p}^{p} \sum_{v=-p}^{p} K(u,v) G_{k+1}(floor((x+u)/2), floor((y+v)/2)) \tag{3.23}$$

Une pyramide laplacienne L est construite niveau par niveau en utilisant les deux pyramides d'images précédentes :

$$L_k(x,y) = G_k(x,y) - E_k(x,y) \tag{3.24}$$

Pour chaque modalité, une pyramide laplacienne est calculée. La fusion des pyramides laplaciennes est effectuée niveau par niveau en prenant le pixel visible ou infrarouge qui a la valeur absolue la plus importante [Gilmore 2009]. Cette fusion multi-niveaux permet de mettre en valeur un plus grand nombre d'objets dans la scène, c'est-à-dire, des objets proches, comme des objets lointains.

Dans les travaux de Gilmore et al, la fusion des images est une étape de prétraitement de la détection. La détection combine analyse du flux-optique thermique (appelée flux de chaleur) et classification (le classifieur SVM est entraîné avec des contours de personnes) [Gilmore 2009]. Tel que le problème a été défini dans l'article, l'analyse du flux de chaleur implique que le système de vision ne soit pas en mouvement.

3.4.2.3 Conclusion

Dans cette section : 1) nous avons vu que les performances de détection peuvent être améliorées significativement en fusionnant les résultats de plusieurs détecteurs de conception différentes (l'ajout d'un nouveau détecteur au système ne dégradant pas les performances globales) et 2) nous avons vu également que la fusion du visible et de l'infrarouge permettait d'améliorer la perception de la scène dans le but final de faciliter la détection des personnes.

(1) Il pourrait être intéressant de fusionner les résultats d'un détecteur opérant sur le spectre visible avec les résultats d'un détecteur opérant sur le spectre infrarouge. Cependant, utiliser plusieurs détecteurs fait inévitablement augmenter le temps de calcul. On s'éloigne un peu plus de la compatibilité temps-réel à chaque ajout d'un détecteur au système.

(2) La fusion permet d'obtenir une information enrichie à partir de sources d'information différentes. Ainsi, avec une telle approche, il est possible de mettre en valeur certaines propriétés visuelles de la scène plutôt que d'autres. Cependant, la fusion entraîne également une perte d'information qui pourrait être utile : une partie de l'information visible et de l'information infrarouge est perdue dans le processus de fusion. Il serait préférable de traiter séparément l'information visible et l'information infrarouge pour ne pas perdre de l'information qui pourrait être utile à la phase de détection.

3.4.3 Collaboration de détecteurs à l'apprentissage

Les entraînements de plusieurs détecteurs peuvent être renforcés mutuellement par un processus semi-supervisé appelé co-entraînement ("co-training", en anglais). Le co-entraînement a été proposé pour la première fois par Blum et Mitchell en 1998 comme une approche pour améliorer l'entraînement de plusieurs classifieurs différents [Blum 1998]. Les classifieurs doivent traiter des "vues de données" qui sont différentes et complémentaires. Bien que le co-entraînement ait été développé à l'origine pour les applications web, il existe un certain nombre de travaux utilisant le co-entraînement dans le contexte de la détection d'objets. À noter qu'il semblerait que le co-entraînement n'ait pas encore été utilisé avec un détecteur infrarouge lointain.

L'idée du co-entraînement est assez simple : deux (ou plus) classifieurs peuvent se renforcer les uns avec les autres en apprenant des résultats de détection de chacun (Fig.3.17). Cependant, deux hypothèses sont requises : 1) les classifieurs doivent être conditionnellement indépendants et 2) ils doivent être initialement faiblement pré-entraînés [Blum 1998].

Dans la suite de cette section des travaux majeurs en détection d'objet utilisant le co-entraînent seront présentés.

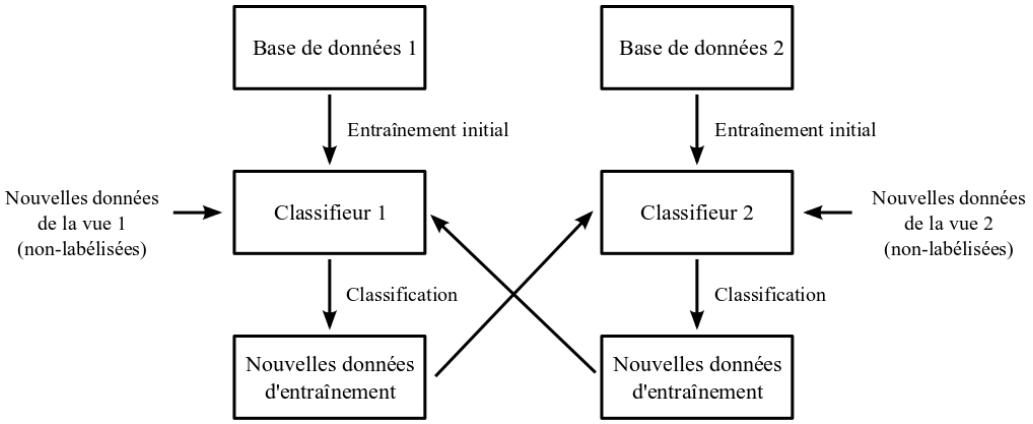


FIGURE 3.17 – Principe de fonctionnement du co-entraînement de deux classifieurs.

Utilisation du co-entraînement pour une amélioration semi-supervisée de détecteurs.

En 2003, Levin et al proposèrent un des premiers travaux de co-entraînement appliqué à la détection d'objet [Levin 2003]. Deux classifieurs différents sont utilisés dans leur travail : un classifieur dont l'entrée est l'information de niveau de gris de l'image et un classifieur dont l'entrée est la soustraction de fond temporelle de la scène. Les deux classifieurs sont tous les deux entraînés en utilisant une variante de l'algorithme AdaBoost : le LogAdaBoost (lequel procède à la sélection des caractéristiques visuelles par un algorithme de regression logistique).

D'une manière générale il est important de filtrer les nouvelles images d'entraînement acquises pendant le co-entraînement ; cela est nécessaire pour ne pas polluer le renforcement des classifieurs avec de nouvelles données d'entraînement incorrectement labélisées (par exemple : un cas négatif qui serait considéré comme étant une image d'entraînement positive). Dans ce but, Les auteurs proposèrent de calculer pour chaque classifieur un seuil de confiance permettant de filtrer les faux-positifs des vrai-positifs. Ces seuils sont calculés après l'entraînement initial des classifieurs.

Les auteurs firent l'hypothèse que pour chaque classifieur il existe une valeur seuil $\theta > 0$ telle que la probabilité que x soit une détection soit proche de 1. Cette valeur seuil θ vérifie la propriété suivante :

$$H(x) = \sum_{m=0}^{T} \alpha_m h_m(x) < \theta \tag{3.25}$$

Où $\forall m \in \{0, T\}$ h_m est le classifieur faible m et α_m est son poids.

Suivant ce principe, deux seuils sont calculés après l'entraînement initial de chaque détecteur : un seuil θ_p correspondant au résultat maximal obtenu pour une image d'entraînement négative sur une base de données de validation et un seuil θ_n correspondant au résultat minimal obtenu par une image d'entraînement positive sur la base de données de validation.

Ainsi, il est possible de filtrer les faux-positifs des vrai-positifs grâce à θ_p (vrai positif si $H(x) > \theta_p$) et de filtrer les faux-négatifs des vrai-négatifs avec θ_n (vrai-négatif si $H(x) < \theta_n$).

Les tests effectués par les auteurs montrèrent que l'approche permet de co-entraîner avec succès deux détecteurs faiblement entraînés jusqu'à atteindre les performances de détection de l'état de l'art [Levin 2003].

Apprentissage multi-instances à partir de plusieurs caméras.

En 2010, Roth et al proposèrent une approche d'apprentissage semi-supervisée qui se base sur l'analyse de différents points de vue de la scène [Roth 2010]. Cette approche nécessite l'utilisation simultanée de plusieurs caméras ; chaque caméra est associée à un détecteur de personnes (Fig.3.18). La personne filmée se situe à des endroits différents de l'image suivant la caméra : elle est centrée dans l'image pour la caméra $C1$, elle est à la droite de l'image pour la caméra $C2$ et elle est à la gauche de l'image pour la caméra $C3$. Les faux-positifs sont en rouge et les vrais positifs en vert. Les détections obtenues sont projetées et mises en correspondance dans les autres vues par transformation géométrique (homographies). Les détections projetées sont utilisées comme autant de nouveaux cas d'apprentissage différents pour renforcer les détecteurs ; ainsi plus de nouvelles données d'entraînement peuvent être générées.

Il peut arriver que les transformations géométriques ne permettent pas une re-projection géométrique très précise des détections dans les autres vues [Roth 2010]. Cela peut avoir une mauvaise incidence à l'apprentissage lorsque ces images sont réutilisées pour le renforcement des détecteurs. Pour éviter cela, Roth et al proposèrent d'utiliser le formalisme d'instance d'apprentissage multiple ("Multiple Instance Learning" ou MIL [5], en anglais). Le MIL permet de gérer l'ambiguïté dans les données d'entraînement. Avec le MIL, les éléments d'entraînement sont groupés en sacs, chaque sac contient un nombre arbitraire d'éléments d'entraînement. Il y a deux sortes de sacs : les sacs négatifs et les sacs positifs. Un sac négatif ne contient que des éléments d'entraînement négatifs et un sac positif contient au moins un élément d'entraînement positif. Le sac contient les différentes vues d'une même détection. L'algorithme d'apprentissage utilisé est le MILBoost, qui est un dérivé de AdaBoost. La méthode proposée par les auteurs est capable d'estimer la

5. *Multiple Instance Learning*, Instance d'apprentissage multiple

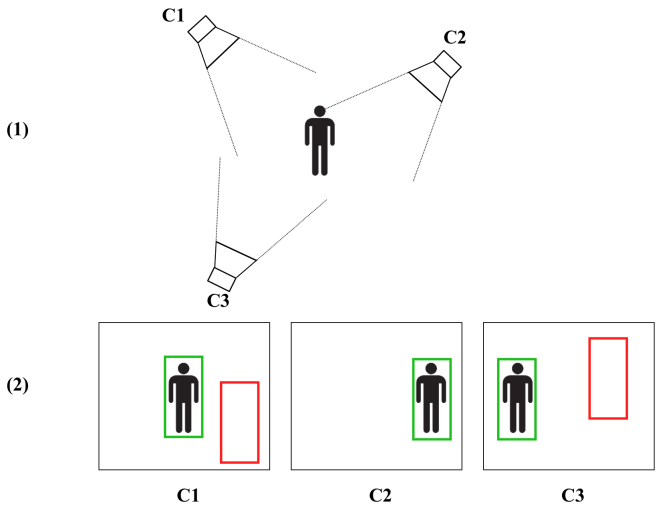


FIGURE 3.18 – Co-entraînement en utilisant plusieurs points de vue.

classe du sac ainsi que l'instance dans le sac qui est la plus intéressante (l'image de la vue où la personne est la mieux alignée) [Roth 2010].

Étant donnée que cette méthode nécessite l'utilisation de plusieurs caméras et la connaissance de l'emplacement de chacune des caméras dans la scène ; cela est une contrainte physique importante si on souhaite que l'environnement derrière les personnes change (ce qui est important pour l'apprentissage). Le système est moins facilement utilisable en pratique qu'une ou (plusieurs) caméra(s) mobile(s).

3.5 Notre approche collaborative d'apprentissage des détecteurs visible et infrarouge

Le co-entraînement du détecteur visible et du détecteur infrarouge permet de régler un problème caractéristique auquel on fait face lorsque l'on souhaite entraîner un détecteur de personnes dans le spectre infrarouge : le manque de données d'entraînement infrarouge de qualité disponibles. Or, nous savons que les performances de détection sont très dépendantes de la richesse des données d'entraînement. Il existe peu de bases de données disponibles en libre usage, certains auteurs ne préférant pas les divulguer librement. Certaines bases de données ne sont pas assez riches ou alors trop spécifiques à un cas d'utilisation [ETHZ 2014]. Les caméras infrarouge évoluant rapidement, les données d'entraînement disponibles peuvent ne pas correspondre aux caractéristiques de la caméra utilisée. Nous avons donc constitué manuellement une petite base de données infrarouges dans le but de procéder à l'entraînement initial du détecteur infrarouge [ATI 2015]. Nous avons fait de même pour le détecteur visible, bien qu'il existe un grand nombre de bases de

données visible [ATV 2015].

Pour un fonctionnement idéal, le co-entraînement nécessite que deux conditions soient remplies : (1) les détecteurs doivent opérer sur des "vues de données" différentes et (2) ils doivent être pré-entraînés. La condition (1) est intrinsèquement remplie dans notre cas car nous analysons deux modalités différentes en parallèle : le spectre visible et le spectre infrarouge long/lointain. La condition (2) nécessite la constitution d'une base de données d'entraînement infrarouge et d'une base de données d'entraînement visible pour un pré-entraînement des détecteurs.

Nous détaillons dans cette section l'approche de co-entraînement infrarouge / visible que nous avons proposée durant cette thèse. Le co-entraînement des détecteurs nous a permis de renforcer nos détecteurs et, d'une manière générale, d'acquérir quasi-automatiquement plus de données d'entraînement.

3.5.1 Notre co-entraînement des détecteurs infrarouge et visible

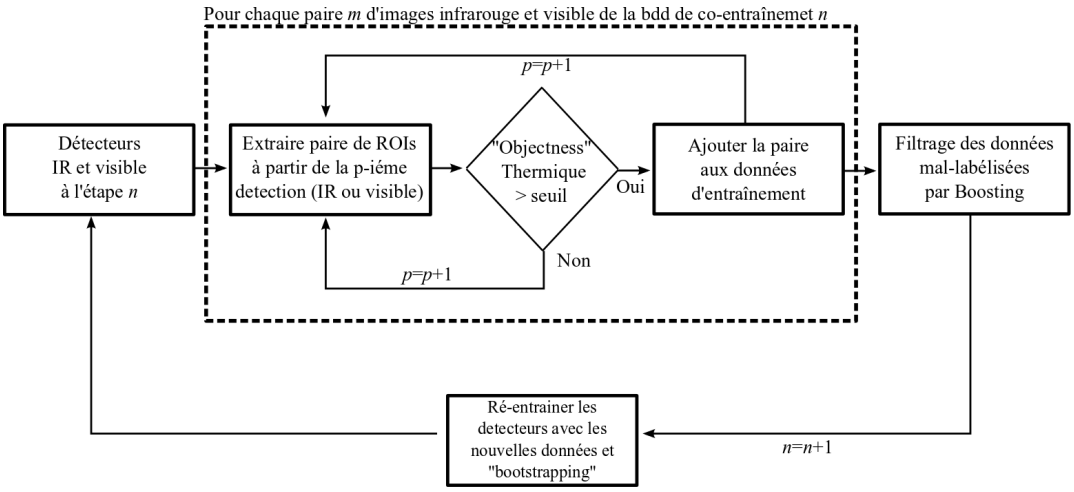


FIGURE 3.19 – Chaîne de traitement du co-entraîneur infrarouge / visible

La chaîne de traitement proposée fonctionne ainsi :

1. Deux détecteurs nouvellement entraînés (un détecteur infrarouge IR-ACF/SoftCascade [Brehar 2014] et un détecteur visible ACF/SoftCascade [Dollár 2014]) traitent les paires d'images de co-entraînement d'indice m (Fig.3.19).

2. Pour chaque modalité et pour chaque éventuelle détection p (dans n'importe laquelle des modalités) on projette la détection dans l'autre modalité. La position de la fenêtre de détection est réajustée en procédant à un recalage local

de l'image. Deux régions d'intérêt (ROI [6]) sont extraites : la détection d'origine ainsi que la détection projetée dans l'autre modalité.

3. Une mesure du "caractère objet" (ou "objectness", en anglais) basée sur l'information thermique est calculée au voisinage et à l'intérieur de la région d'intérêt du spectre infrarouge. Cette mesure est comparée à un seuil, si la mesure est supérieure à ce seuil alors les régions d'intérêt sont rajoutées aux bases de données d'entraînement : la région visible est ajoutée à la base de données visible et la région infrarouge est ajoutée à la base de données infrarouge.
4. Les bases de données sont ensuite filtrées pour supprimer les intrus qui auraient pu être ajoutés malencontreusement à l'étape précédente.
5. Les détecteurs sont ensuite ré-entraînés avec les nouvelles données d'entraînement. Ils sont ensuite renforcés plusieurs fois par des étapes de "bootstrapping" consistant à extraire de nouvelles données d'entraînement négatives.

L'opération peut être répétée n fois, c'est-à-dire, autant de fois qu'il y a de bases de données de co-entraînement (contenant chacune d'entre elles des paires synchronisées d'images infrarouge et visible de personnes).

Nous allons détailler trois étapes déterminantes de cette chaîne de traitement : le recalage local des détections projetées, le calcul de la mesure du "caractère objet" thermique, ainsi que le filtrage des bases de données d'entraînement.

Recalage local des détections projetées

Il est important que les nouvelles données d'entraînement soient bien alignées pour ne pas dégrader l'entraînement des détecteurs. En effet, il peut arriver que le contenu d'une détection projetée ne soit pas parfaitement aligné par rapport au contenu de la détection d'origine. Cela peut arriver quand les personnes sont proches du système de vision ; il y a donc dans ce cas contradiction avec l'hypothèse formulée lors de la synchronisation spatiale (personnes à l'infini du système stéréoscopique). Un léger décalage peut également apparaître si la synchronisation temporelle n'est pas assez précise et si les personnes bougent rapidement (Fig.3.20.1).

Un traitement particulier est nécessaire pour recaler le contenu de la détection projetée avec le contenu de la détection d'origine (Fig.3.20.2). Nous avons décidé d'utiliser l'approche proposée par Kim et al en 2008 qui permet le recalage d'images multi-capteurs [Kim 2008].

L'approche proposée a certaines similitudes avec les approches statistiques classiques de recalage d'images par maximisation de l'information mutuelle

6. *Region Of Interest*, Région d'intérêt

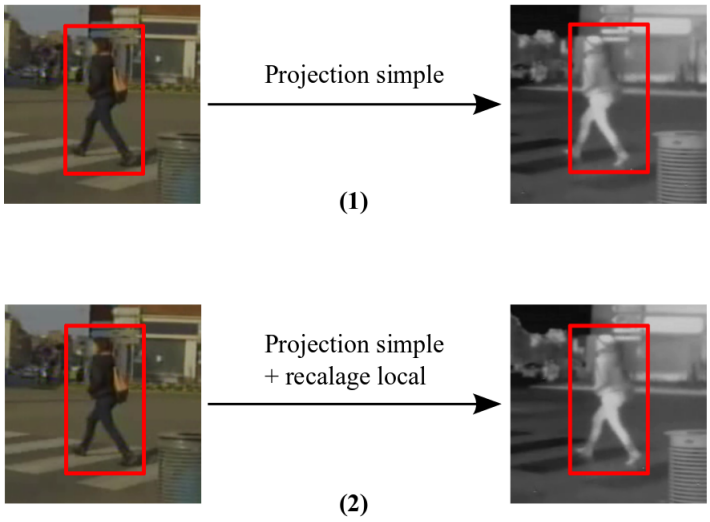


FIGURE 3.20 – Exemple d'une détection projetée décalée et exemple de la même détection projetée corrigée localement.

[Maes 1997]. Ici, plutôt que de maximiser une fonction objectif d'information mutuelle, Kim et al proposèrent de minimiser une fonction objectif qui est fonction de l'information mutuelle et d'une entropie conjointe (Equ.3.31). Cette entropie conjointe est fonction de l'intensité des deux images ainsi que du degré de similarité des contours entre les deux images. L'idée principale de cette approche est d'utiliser l'information de contour comme une information supplémentaire pour recaler des images multi-capteurs. Cela est justifié par l'invariance de l'orientation des contours dans le visible et l'infrarouge ; la magnitude des contours est quant à elle différente dans les deux modalités. L'orientation des contours des images est calculée par analyse des vecteurs propres [Kim 2008].

Avant de rentrer plus profondément dans les détails de la méthode, il est nécessaire de rappeler la définition de l'entropie de Shanon (Equ.3.26) ainsi que la définition de l'entropie conditionnelle (Equ.3.27) :

$$H(X) = -\sum_{x \in \mathbb{X}} p(x) \times log_2(p(x)) \tag{3.26}$$

$$H(X/Y) = \sum_{(x,y) \in (\mathbb{X},\mathbb{Y})} p(x,y) \times log_2(\frac{p(x)}{p(x,y)}) \tag{3.27}$$

Ici, p est simplement l'histogramme d'occurence de d'intensité normalisé dans l'image visible X (ou infrarouge) et $p(x,y)$ est un l'histogramme de co-occurence des intensités x et y entre l'image infrarouge X (ou Y) et l'image visible Y (ou

X). À noter que les images sont en niveaux de gris et que les histogrammes sont normalisés entre 0 et 1 pour être considérés comme des probabilités.

Soient I et V respectivement l'image infrarouge et l'image visible. Recaler les images revient à minimiser l'entropie conditionnelle $H(I/V)$ (Equ.3.28) et l'entropie conditionnelle $H(V/I)$ (Equ.3.29) dans le but de statistiquement maximiser l'information commune aux deux images.

$$H(I/V) = H(I,V) - H(V) \tag{3.28}$$

$$H(V/I) = H(I,V) - H(I) \tag{3.29}$$

Minimiser l'équation Equ.3.30 revient à maximiser la similarité de contours entre les images connaissant I et V.

$$H(O/I,V) = H(I,V,O) - H(I,V) \tag{3.30}$$

L'entropie conjointe $H(I,V,O)$ nécessite la construction d'un histogramme à trois dimensions $p(x,y,o)$ où o correspond à la similarité d'orientation entre les pixels de valeurs x et y. Si les orientations sont similaires, o est proche de 0, dans le cas contraire, o est proche de 1 [Kim 2008]. Nous avons préféré que o prenne des valeurs entières non-signées (entre 0 et 255) dans notre implémentation de l'algorithme. Nous avons aussi préféré utiliser à la place un histogramme à deux dimensions construit à l'aide de "l'image o" dans le but d'alléger les temps de calcul de l'entropie.

La fonction objectif finale $E(I,V)$ est simplement l'addition des trois entropies conditionnelles décrites au dessus ; sa minimisation permet de maximiser l'information mutuelle d'intensité et la similarité de contours entre les images en même temps. La fonction objectif finale est définie comme suit :

$$\begin{aligned} E(I,V) &= H(I/V) + H(V/I) + H(O/I,V) \\ &= H(I,V,O) - (H(I) + H(V) - H(I,V)) \\ &= H(I,V,O) - M(I,V) \end{aligned} \tag{3.31}$$

Où $M(I,V)$ est la définition traditionnelle de l'information mutuelle.

L'entropie finale $E(I,V)$ est ensuite pondérée pour éviter les minimaux locaux lors de l'optimisation de la fonction [Kim 2008].

Dans leurs travaux, les auteurs minimisèrent $E(I,V)$ grâce à la méthode de Nelder-Mead [Nelder 1965]. Cette méthode consiste à transformer de manière itérative un simplexe (objet polytote ayant une dimension supérieure à celle de l'espace traité) jusqu'à convergence. La fonction objectif est évaluée en chacun des sommets du simplexe, des règles de transformation sont ensuite appliquées à

chaque itération pour transformer le simplexe afin que celui-ci se rapproche du minimum.

La méthode de Nelder-Mead nécessite de bien définir le simplexe initial au risque de converger vers un minimum local. Dans notre cas, nous avons observé qu'il était parfois nécessaire de redéfinir le simplexe initial suivant les cas, au risque de mal converger. Bien que cette méthode ait l'avantage d'être très rapide, nous avons préféré utiliser une approche "brute de force" consistant à parcourir de manière exhaustive l'espace des solutions avec un pas minimal. Nous avons observé un recalage optimal des images plus stable au cours du temps, mais au prix d'une importante augmentation des temps de calcul.

Sélection de vrais positifs : calcul de la mesure du "caractère objet" thermique

Il est nécessaire d'écarter les faux-positifs pour ne pas les ajouter aux images d'entraînement positives ; sans quoi les performances des détecteurs chuteraient considérablement après chaque ré-entraînement. Nous avons décidé d'utiliser une mesure du "caractère objet" calculée dans l'infrarouge dans le but de filtrer les paires de détections originales/projetées qui sont des fausse-positives de celles qui sont des vrai-positives. Le principe est de jauger la probabilité que le contenu thermique de la détection infrarouge correspond à celui d'un être humain. Or, dans l'infrarouge, l'être humain peut être associé à un objet chaud et fermé. En procédant à une analyse des contours il est possible de calculer une telle mesure ; Zitnick et al proposèrent récemment une mesure du "caractère objet" basée sur une analyse des contours mais pour le cas visible [Zitnick 2013]. L'analyse des contours dans l'infrarouge est justifiable par le fait que l'être humain est une source constante de chaleur et que par conséquent les contours de celui-ci apparaissent nettement par rapport au reste de la scène. De plus, nous avons remarqué que l'analyse des contours est simplifiée dans l'infrarouge car il y a beaucoup moins d'ambiguïtés d'appartenance des contours entre les objets de la scène, les contours correspondant aux séparations de température entre les objets de la scène. Une mesure élevée du "caractère objet" infrarouge du contenu de la détection côté infrarouge indiquera une forte probabilité de présence d'un objet chaud fermé, la paire de détections pourra donc être ajoutée aux bases de données d'entraînement.

La mesure thermique que nous proposons est basée sur la mesure de Zitnick et al ; nous l'avons adaptée à notre cas d'utilisation, c'est-à-dire au filtrage des paires de détections. Dans la suite de cette sous-section nous allons : 1) décrire la mesure EB (ou Edge-Boxes) de Zitnick et al et 2) décrire notre mesure du "caractère objet" calculée dans l'infrarouge que nous avons appelée la mesure CAO [7] ("Centered and

7. *Centered and Anti-Overflow objectness*, Mesure du caractère objet centrée et sans déborde-

Anti-Overflow Objectness", en anglais).

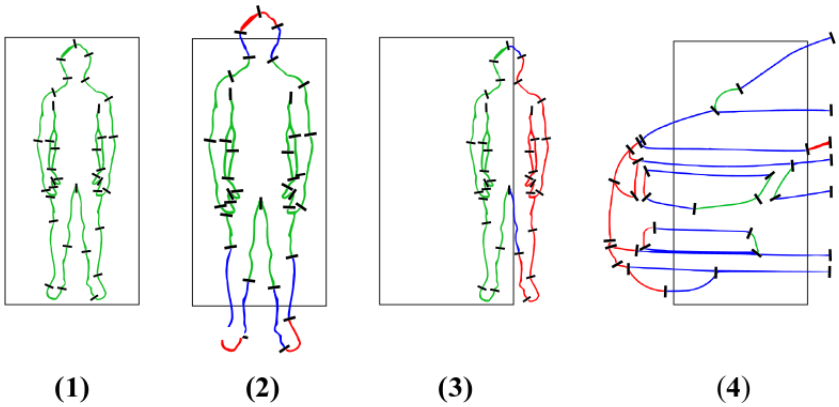


FIGURE 3.21 – Les trois types de contours : externe (rouge), traversant (bleu) et interne (vert).

1. **La mesure EB**

   La mesure EB [8] est basée sur le nombre de contours pleinement connectés qui est une indication que la boîte d'analyse (détection dans notre cas) puisse contenir un objet [Zitnick 2013]. La mesure EB est calculée comme suit : 1) dans un premier temps les pixels de bord sont extraits de l'image et des contours sont formés en regroupant les pixels de bord ayant des valeurs d'orientation similaires, 2) les petits contours sont fusionnés entre eux pour former des contours plus grands, 3) une mesure d'affinité est calculée pour chaque paire de contours, l'affinité entre les contours s_k et $s_{k'}$ est élevée si l'angle entre les moyennes de contours $\theta_{kk'}$ est similaire aux orientations des contours θ_k et $\theta_{k'}$ (Equ.3.32). L'affinité est nulle si les contours sont séparés par plus de deux pixels. Le paramètre γ est un paramètre de sensibilité, les auteurs conseillent de l'initialiser à 2.

   $$a(s_k, s_{k'}) = |cos(\theta_k - \theta_{kk'})cos(\theta_{k'} - \theta_{kk'})|^{\gamma} \tag{3.32}$$

   Il y a trois types de contours : externe, traversant et internes (Fig.3.21). Les contours externes sont intégralement extérieurs à la boîte d'analyse (où fenêtre de détection) ; ils ne contribuent pas au calcul de la mesure EB. Les contours traversant contribuent indirectement au calcul, ce sont les contours traversant les limites de la boîte d'analyse. Les contours internes contribuent directement au calcul, ce sont les contours intégralement compris à l'intérieur de la boîte d'analyse.

ment

8. *Edge-Boxes score*, Mesure "Edge-Boxes"

Pour chaque contour s_k un poids $w(s_k)$ est calculé : les contours externes et traversant ont un poids nul, les contours internes ont un poids variant entre 0 et 1 calculé comme suit :

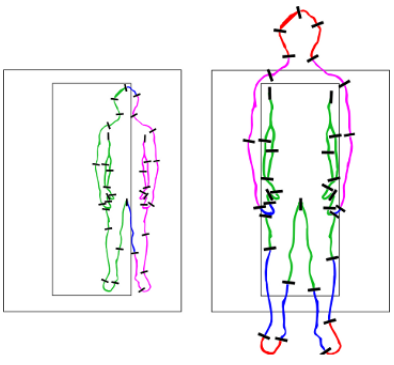
$$w(s_k) = 1 - \max_{T} \prod_{j}^{|T|-1} a(t_j, t_{j+1}) \tag{3.33}$$

L'équation précédente signifie que le poids d'un contour interne est abaissé si le contour interne s_k est connecté à un contour traversant par un chemin d'affinités non-nulles ; T est un chemin ordonné de contours, où $t_1 = s_k$ et t_T est un contour traversant. Le chemin d'affinités non-nulles maximal est retenu pour le calcul du poids.

La mesure EB est calculée comme suit :

$$H = \frac{\sum_k w(s_k) m_k}{2(b_l + b_h)^\kappa} \tag{3.34}$$

Le paramètre κ est un paramètre de sensibilité, les auteurs conseillent également de l'initialiser à 2. Les valeurs m_k sont les sommes des valeurs de magnitude des pixels du contour s_k. Il est nécessaire de connaître les propriétés géométriques b_l et b_h qui sont respectivement la largeur et la hauteur de la boîte. De Fig.3.21.1 à Fig.3.21.4 : la mesure EB décroît car le nombre de contours traversant augmente et le nombre de contours internes diminue. La mesure maximale est trouvée lorsque le contenu est un objet (au sens large) fermé et qui est entièrement compris dans la boîte.



(1) **(2)**

FIGURE 3.22 – Boîtes d'analyse utilisées pour le calcul de la mesure CAO, en violet : les contours externes affiliés.

2. **La mesure CAO**

   La mesure EB peut être relativement élevée pour un objet non-centré ou pour un objet dépassant de la boîte d'analyse. Dans notre chaîne de traitement, cela aurait pour conséquence d'accepter comme nouvelles données d'entraînement positives des détections mal-centrées sur les personnes en raison de mesures élevées. Ce phénomène est particulièrement important dans l'infrarouge, car il y a un fort contraste entre la scène et les êtres humains.

   Nous avons adapté la mesure EB dans le but d'élargir la différence de mesure entre les faux-positifs et les vrai-positifs. Nous avons nommé notre approche : la mesure du "caractère objet" centré et anti-débordement ou mesure CAO ("Centered and Anti-overflow objectness", en anglais). La mesure CAO calculée dans l'infrarouge est élevée pour les objets chauds et centrés de la scène. De tels objets sont typiquement des détections de personnes centrées.

$$H' = H - \frac{\sum_{k=0}^{card(S_e)} w(s_k) m_k}{2(b_w + b_h)^\kappa} \tag{3.35}$$

   Deux zones d'analyse sont considérées pour calculer la mesure CAO (Fig.3.22). Ces deux zones d'analyse sont centrées par rapport à la fenêtre de détection d'origine : la zone d'analyse intérieure est plus petite que la fenêtre de détection et la zone d'analyse extérieure est plus grande. Les contours se trouvant dans la zone d'analyse extérieure ayant un chemin d'affinités non-nulles avec un des contours traversant de la zone d'analyse intérieure sont les contours externes affiliés (ensemble noté S_e). Les contours externes affiliés ont une contribution négative à la mesure du "caractère objet" H' finale (Equ.3.35). Un contour externe affilié a un poids non nul qui est calculé de la même manière que dans Equ.3.33. Seuls les contours externes affiliés ont une influence sur le calcul de la mesure ; cela permet de faire baisser la valeur de la mesure CAO de manière importante lorsqu'il y a un débordement quelconque de l'objet, en évitant de faire baisser la mesure pour des contours externes isolés.

3. **Alimentation de la base de données :**

   Nous générons de nouvelles données positives d'entraînement en utilisant la mesure CAO. Si la mesure CAO est élevée il y a une forte indication que la paire soit un vrai-positif. Dans ce cas la paire peut être ajoutée aux bases de données. Un seuil élevé sur la mesure CAO ne permet de garder qu'un sous-ensemble des vrai-positifs. Cependant, ce sous-ensemble est

largement suffisant pour alimenter le co-entraîneur en nouveaux éléments d'entraînement positifs. À noter qu'une mesure basse ne signifie pas forcément que la paire soit un faux-positif, la mesure ne permet donc pas d'extraire de nouvelles données d'entraînement négatives.

Nous générons de nouvelles données d'entraînement négatives de deux manières : 1) en générant de manière aléatoire des nouveaux cas à partir d'images négatives pleine-résolution (c'est-à-dire des images ne contenant aucun être humain et 2) par des phases de "bootstrapping", dont le but est de générer de nouvelles images d'entraînement négatives à partir de faux-positifs de détection.

Filtrage des bases de données d'entraînement par boosting

La mesure CAO étant une analyse bas-niveau des détections, il peut arriver que certaines paires fausse-positives soient ajoutées à tort aux bases de données d'entraînement positives. Une seconde étape de filtrage est nécessaire dans le but d'identifier et de supprimer les éléments d'entraînement positifs mal-labélisés. Nous avons décidé d'adapter l'algorithme du filtrage du bruit par "boosting" à notre cas d'utilisation [Zhong 2005]. Le "boosting" est très sensible aux éléments mal-labélisés et c'est cette propriété qui est utilisée dans l'algorithme pour les identifier. De plus, nous utilisons également une approche basée "boosting" pour entraîner les détecteurs ACF/SoftCascade et IR-ACF/SoftCascade de notre chaîne de traitement. Cela signifie que cette approche de filtrage est particulièrement bien adaptée à notre cas car : 1) les éléments mal-labélisés identifiés avec cette méthode de filtrage sont les mêmes qui font dériver l'apprentissage des détecteurs de manière importante et 2) si il reste quelques éléments mal-labélisés après le filtrage, nous pouvons être relativement sûr que ceux-ci auront une influence négligeable à l'entraînement (car ils auront eu une faible importance au filtrage du bruit par "boosting").

Le principe général de l'algorithme est d'associer à chaque élément d'entraînement un compteur de bruit qui est ensuite incrémenté au fur et à mesure de l'analyse. Les éléments d'entraînement ayant des décomptes anormalement grands sont susceptibles d'être des éléments mal-labélisés. Ils sont des candidats potentiels à la suppression.

L'algorithme du filtrage par "boosting" est définit comme suit :

Algorithm 7: Filtrage du bruit par "Boosting"

Data: Base de données d'entraînement (D_+, D_-)
Result: Décomptes de bruit pour tous les éléments d'entraînement

1 Initialiser les décomptes totaux de bruit tc_n à 0 $\forall n \in D_+ \cup D_-$
2 Initialiser les poids $w_n = 0$
3 **for** $t = 0$ *jusqu'à* T **do**
4 **for** $j = 0$ *jusqu'à* m **do**
5 (S_+, S_-) = sousEnsembleAléatoire$((D_+, D_-))$
6 $w_n = \frac{1}{card(S_+ \cup S_-)}, \forall n \in (S_+, S_-)$
7 H^j_{sc} = entraînerClassifieurSoftCascade$((S_+, S_-))$
8 **end**
9 Remettre les décomptes locaux de bruit nc_n à 0 $\forall n \in D_+ \cup D_-$
10 **for** $n = 0$ *jusqu'à* $card(D_+, D_-)$ **do**
11 **for** $j = 0$ *jusqu'à* m **do**
12 **if** $H^j_{sc}(n)$ *!=* $label(n)$ **then**
13 $nc_n = nc_n + 1$
14 **end**
15 **end**
16 $tc_n = tc_n + nc_n$
17 **end**
18 **for** $n = 0$ *jusqu'à* $card(D_+, D_-)$ **do**
19 $w_n = w_n \times e^{nc_n}$
20 **end**
21 Normaliser les poids $w_n = \frac{w_n}{\sum_{n \in D_+ \cup D_-} w_n}$
22 **end**

La base de données d'entraînement (D_+, D_-) contient des éléments mal-labélisés. Le filtrage est appliqué pour la base de données d'entraînement visible et la base de données d'entraînement infrarouge. Lorsque la base de données visible est filtrée les caractéristiques visuelles ACF sont utilisées, lorsque la base de données infrarouge est filtrée les caractéristiques visuelles IR-ACF sont utilisées. Les caractéristiques visuelles sont extraites lors de l'apprentissage du classifieur (ligne 7) et lors de l'évaluation des éléments (ligne 12). Plusieurs classifieurs sont entraînés sur des sous-ensembles aléatoires différents de (D_+, D_-), exactement m classifieurs différents sont entraînés (ligne 4). Chaque élément d'entraînement est évalué par les m classifieurs.

Le décompte local de bruit est augmenté (ligne 13) lorsqu'un élément d'entraînement est mal classifié (c'est-à-dire que le label est différent de la classe trouvée). Les décomptes totaux de bruit sont mis à jour au fur et à mesure en ajoutant les

décomptes locaux (ligne 16). Les poids de "boosting" des éléments fortement mal-classés par les m classifieurs sont augmentés et les poids des éléments bien classés sont diminués (ligne 19). Les prochains m classifieurs entraînés essayeront de réduire les poids élevés : c'est le principe du "boosting". Les résultats des filtrages appliqués sur la base de données d'entraînement visible et infrarouge sont croisés pour améliorer la suppression des éléments mal-labélisés. Pour filtrer les images visibles et les images infrarouges nous avons choisi les paramètres $T = 50$ et $m = 10$. Dans notre cas, nous avons observé qu'il suffisait de filtrer les 5% des éléments ayant les plus forts décomptes pour supprimer le bruit des bases de données d'entraînement.

3.5.2 Expérimentations

Dans cette section les bénéfices apportés par notre approche de co-entraînement multimodale sont évalués. Dans un premier temps les performances de l'algorithme de filtrage du bruit par "boosting" que nous avons adapté au cas de la détection de personnes sont testés. Puis, dans un second temps la chaîne de traitement complète est évaluée pour plusieurs itérations de co-entraînement.

Filtrage de bruit par "boosting" adapté à la classification de personnes.

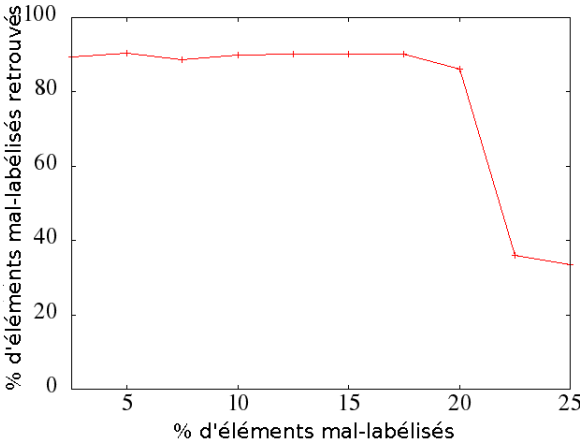


FIGURE 3.23 – Performances du filtre de bruit par "boosting" pour plusieurs pourcentages d'éléments mal-labélisés.

Les capacités de filtrage ont été testées sur la base de données INRIA [INRIA 2014] pour plusieurs pourcentages d'éléments mal-labélisés (Fig.3.23) ; Nous avons choisi de filtrer des images visible pour ce test, mais nous aurions très bien pu choisir de filtrer les images infrarouge. Le pourcentage d'éléments mal-labélisés a été augmenté par pas de 2,5%, à chaque fois nous avons filtré les

FIGURE 3.24 – Échantillon d'images d'entraînement positives contenant deux images mal-labélisées (5éme et 7éme images).

éléments d'entraînement avec les plus grands décomptes de bruit. Nous avons décidé de filtrer un nombre d'éléments d'entraînement égal au nombre d'éléments mal-labélisés introduit dans la base de données. Pour chaque test nous avons gardé le résultat moyen obtenu pour 10 cas différents : où chaque cas contient un même pourcentage d'éléments mal-labélisés différents. Pour "générer" des éléments mal-labélisés nous avons simplement échangé des éléments d'entraînement positifs pris au hasard avec des éléments d'entraînement négatifs également pris au hasard.

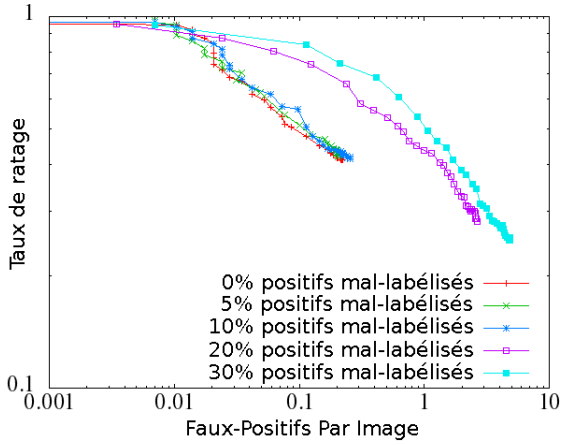


FIGURE 3.25 – Performances du détecteur ACF/SoftCascade entraîné avec la base de données INRIA pour laquelle on a substitué plusieurs pourcentage d'images d'entraînement positives par des images d'entraînement négatives.

Nous pouvons voir que le pourcentage d'éléments retrouvés est d'environ 90% pour un pourcentage d'éléments mal-labélisés allant de 0% à 20%. Le filtre devient inopérant pour un pourcentage d'éléments mal-labélisés supérieur à 20%.

Nous avons également étudié l'impact du nombre d'éléments d'entraînement positifs mal-labélisés sur les performances de détection (Fig.3.25). Nous avons utilisé la base de données INRIA pour ce test et le détecteur ACF/SoftCascade. Nous pouvons voir que les performances de détection se dégradent lentement pour un pourcentage d'éléments mal-labélisés allant de 0% à 10%. Les performances com-

mencent à se dégrader considérablement à partir de 10% et plus. À noter que nous n'avons jamais obtenu un pourcentage d'éléments mal-labélisés supérieur à 5% des éléments ajoutés après la première phase de filtrage par mesure du "caractère objet" infrarouge. Pour ce test, les éléments d'entraînement mal-labélisés ont été générés en substituant des images d'entraînement positives avec des cas négatifs nouvellement générés de manière aléatoire à partir d'images négatives pleine-résolution.

Chaîne de traitement de co-entraînement complète

Nous avons testé la capacité de notre approche de co-entraînement multimodale à améliorer le détecteur IR-ACF/SoftCascade et le détecteur ACF/SoftCascade après trois itérations (Fig.3.26 et Fig.3.27).

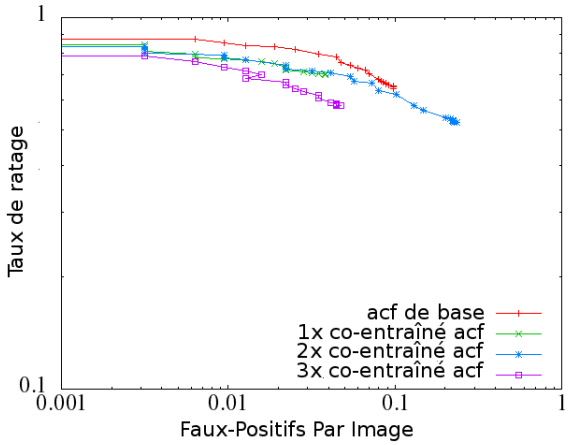


FIGURE 3.26 – Amélioration des performances de détection du détecteur ACF/SoftCascade pour trois itérations de co-entraînement multimodale.

Pour chaque itération n (Fig.3.19) 743 paires d'images de co-entraînement différentes ont été utilisées : pour l'itération $n = 1$ nous avons utilisé les paires d'images de la base de données CTAVIS-1 [9] [CTAVIS-1 2015], pour l'itération $n = 2$ nous avons utilisé les paires d'images de la base de données CTAVIS-2 [10] [CTAVIS-2 2015] et pour l'itération $n = 3$ nous avons utilisé les paires d'images de la base de données CTAVIS-3 [11] [CTAVIS-3 2015] (Fig.3.28).

Les tests ont été effectués sur une quatrième base de données de paires d'images synchronisées contenant 316 autres paires d'images annotées manuellement : la

9. *CoTraining Alpha Visible Infrarouge Synchronized dataset*, Base de données alpha contenant des images visible et infrarouge synchronisés pour le co-entraînement
10. *CoTraining Alpha Visible Infrarouge Synchronized Dataset*, Base de données alpha contenant des images visible et infrarouge synchronisés pour le co-entraînement numéro 2
11. *CoTraining Alpha Visible Infrarouge Synchronized Dataset*, Base de données alpha contenant des images visible et infrarouge synchronisés pour le co-entraînement numéro 3

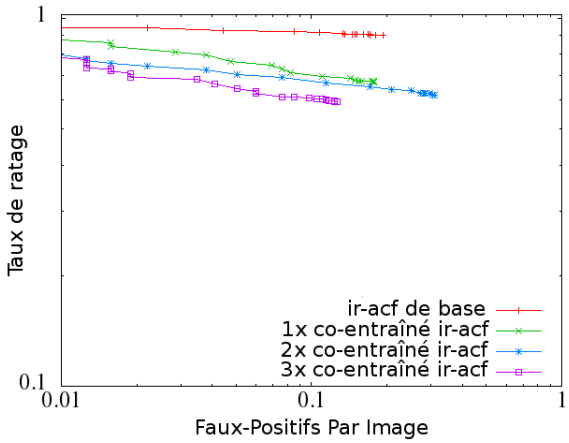


FIGURE 3.27 – Amélioration des performances de détection du détecteur IR-ACF/SoftCascade pour trois itérations de co-entraînement multimodale.

FIGURE 3.28 – Exemples de paires d'images visible / infrarouge de la base de données de co-entraînement CTAVIS-1 (1), CTAVIS-2 (2) et CTAVIS-3 (3). Chaque base de données contient 743 paires différentes de résolution 704x480. Un très grand nombre de personnes est présent dans les images.

base de données AVIS [12] (Fig.3.29) [AVIS 2015].

Les images sont similaires en résolution à celles utilisées pour le co-entraînement. À noter que la base de données AVIS que nous avons créée contient des cas beaucoup plus difficiles à traiter que la base de données OTCBVS [Davis 2005], qui est une base de données d'images visible/infrarouge synchronisées de référence. La base de données AVIS est constituée de paires d'images synchronisées qui ont été prises dans des endroits très différents. Les entraînements initiaux des détecteurs visible et infrarouge ont été fait en utilisant respectivement la base de données ATV [13] [ATV 2015] et la base de données ATI [14] [ATI 2015]

12. *Visible Infrarouge Synchronized Dataset*, Base de données de test contenant des images visible et infrarouge synchronisées
13. *Alpha Training Visible Dataset*, Base de données d'entraînement alpha contenant des images visible
14. *Alpha Training Infrared sataset*, Base de données d'entraînement alpha contenant des images

FIGURE 3.29 – Exemples de paires d'images visible / infrarouge annotées de la base de données de test AVIS. Cette base de données contient 316 paires d'images de scènes complexes.

(Fig.3.30).

(1) **(2)**

FIGURE 3.30 – Exemples d'images d'entraînement positives et négatives de la base de données ATV (1) et ATI (2). La base de données ATV contient 826 images positives et 5002 images négatives, la base de données ATI contient 996 images positives et 5640 images négatives.

Chaque itération a généré environ 300 nouvelles paires d'images d'entraînement positives (Fig.3.31). Après chaque itération les performances de détection du détecteur ACF/SoftCascade et du détecteur IR-ACF/SoftCascade sont améliorées, comme nous pouvons le voir Fig.3.27 et Fig.3.26. Pour le détecteur IR-ACF/SoftCascade, le taux de ratage évolue progressivement à chaque itération, passant de 0.95 à 0.75 puis de 0.70 jusqu'à 0.61 pour 0.1 faux-positifs par image (Fig.3.27). Pour le détecteur ACF/SoftCascade, le taux de ratage évolue progressivement à chaque itération, passant de 0.71 à 0.62 puis de 0.61 jusqu'à 0.5 pour 0.03 faux-positifs par image (Fig.3.26).

infrarouge

FIGURE 3.31 – Exemples de paires d'images d'entraînement extraites lors du co-entraînement multimodale.

3.5.3 Conclusion

Nous avons montré que les performances de détection du détecteur visible et du détecteur infrarouge sont améliorées après chaque itération de notre co-entraînement multimodale. Cette approche semi-supervisée permet de collecter autant de nouvelles images d'entraînement que souhaité or, le nombre et la diversité des images d'entraînement ont un impact important sur l'apprentissage et donc sur les performances de détection. La chaîne de traitement n'est pas dépendante d'un détecteur en particulier. Il est possible d'utiliser d'autres types de détecteurs tel que le HOG/SVM ou le Haar/Cascade par exemple.

La chaîne de traitement ne peut malheureusement pas être utilisée en ligne car les temps de calcul ne permettent pas une exécution temps réel. Trois étapes alourdissent considérablement les temps de calcul : le recalage local de chaque détection projetée, le calcul de la mesure du "caractère objet" thermique ainsi que le filtrage des éléments d'entraînement mal-labélisés. Après un grand nombre d'itérations de co-entraînement n ($n > 5$), les personnes dans les nouvelles images d'entraînement positives peuvent apparaître trop petites ou trop grandes par rapport à la taille de la fenêtre. Ce phénomène s'explique car nous réutilisons directement les détections (et les projetés de détections) comme nouvelles images d'entraînement ; la détection d'une personne correspond à un maximum local de score dont la position et la taille de la fenêtre par rapport à la personne peuvent être différentes de l'encadrement idéal (serré et bien centré) d'une image d'entraînement positive. Ce phénomène peut parasiter l'entraînement des détecteurs. Par conséquent, il n'est pas souhaitable d'itérer un trop grand nombre de fois le co-entraînement.

3.6 Nos approches de détection multimodales de personnes

Dans la Sec.3.4.3 nous avons exploité la complémentarité des modalités visible et infrarouge pour extraire de nouvelles données d'entraînement d'une manière

semi-supervisée avec pour but de renforcer les détecteurs. Dans la section présente, le spectre visible et le spectre infrarouge sont exploités pour permettre une détection plus robuste aux conditions environnementales.

Deux approches différentes de détection multimodale sont présentées dans la suite de cette section : 1) la première approche utilise l'information infrarouge pour réduire l'espace de recherche et donc réduire les temps de calcul, 2) la deuxième approche consiste à faire collaborer d'une manière rapide et flexible les détecteurs visible et infrarouge.

3.6.1 Réduction de l'espace de recherche dans l'infrarouge et détection dans le visible.

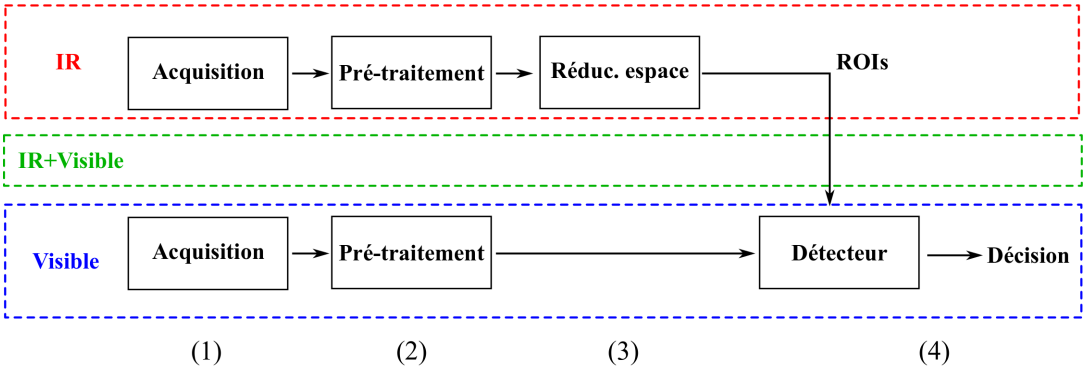


FIGURE 3.32 – Chaîne de traitement générale pour accélérer les temps de calcul à la détection en utilisant la modalité visible et la modalité infrarouge.

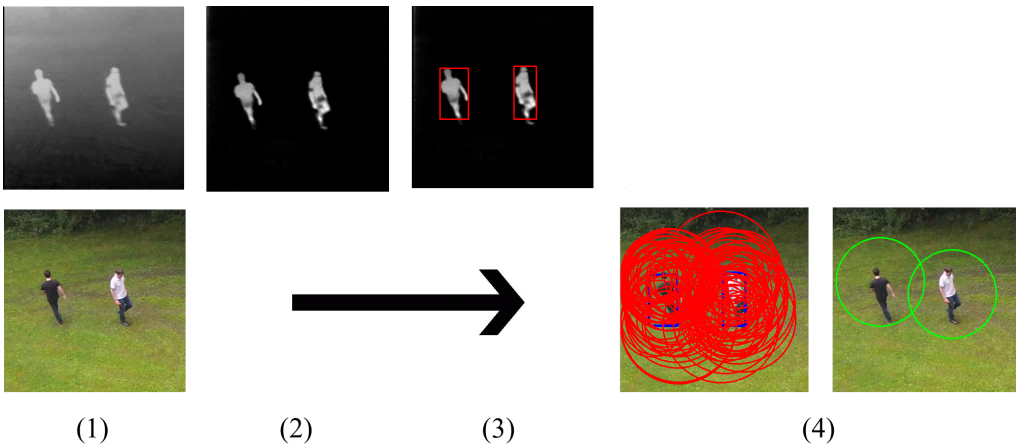


FIGURE 3.33 – Traitement étape par étape des images infrarouge et visible dans la chaîne de traitement.

Nous avons proposé une chaîne de traitement spécifique permettant d'accélérer les temps de calcul à la détection (Fig.3.32). Les images visible et infrarouge sont extraites durant la phase d'acquisition (puis synchronisées temporellement et spatialement), une étape de pré-traitement suit immédiatement, les images infrarouge sont ensuite analysées pour réduire l'espace de recherche, les régions d'intérêt (ROI) trouvées dans l'infrarouge sont analysées par un détecteur supervisé dans le visible.

Dans la suite de cette section nous détaillerons les étapes allant du pré-traitement à la détection (Fig.3.32). La Fig.3.33 est une illustration graphique des étapes de la chaîne de traitement.

Pré-traitement (2)

Il est préférable d'utiliser, dans la chaîne de traitement, la carte de saillance de l'image infrarouge plutôt que l'image infrarouge elle-même. Nous avons observé que cela facilitait considérablement la réduction de l'espace de recherche : grâce à cela, la réduction de l'espace de recherche est moins bruitée et plus robuste aux changements de luminosité. Nous avons décidé d'utiliser l'algorithme d'Achanta et al [Achanta 2010] qui est rapide et permet d'extraire des cartes de saillance de haute qualité. Nous avons légèrement modifié l'algorithme pour le cas infrarouge afin que seul le canal L de l'espace de couleur CIELAB soit traité (les canaux de couleurs a et b n'ayant pas de signification en niveau de gris).

Réduction de l'espace de recherche (3)

Nous avons utilisé l'algorithme de San-Biagio et al pour réduire l'espace de recherche par analyse de la carte de saillance infrarouge [San-Biagio 2012]. Cet algorithme est rapide et cible bien les zones chaudes de la scène qui ont des proportions humaines. Il fonctionne récursivement par extraction et raffinement de régions d'intérêt (ROI). Il s'arrête lorsque les ROIs de la scène ne changent plus de taille après deux itérations. La ROI initiale est l'image entière. Pour chaque itération les ROIs sont analysées de la manière suivante : le nombre de pixels avec une intensité supérieure à un seuil Thr_{etape} est gardé en mémoire pour chaque ligne de l'image. La même chose est effectuée pour chaque colonne de l'image. Les limites des nouvelles ROIs (trouvées à l'intérieur de la ROI traitée) sont trouvés en seuillant les compteurs calculées précédemment pour les lignes et les colonnes. Les seuils pour les lignes ($Thr_{pixel,ligne}$) et pour les colonnes ($Thr_{pixel,colonne}$) dépendent de la taille de l'image et de quelques paramètres [San-Biagio 2012]. Après chaque récursion Thr_{etape} est raffiné comme défini dans Equ.3.36 et Equ.3.37. Où $w1$, $w2$, $w3$ et Thr_{Sk} sont des paramètres. Thr_{ROI} est une valeur de seuil calculée pour chaque ROI.

$$Thr_{etape} = w1 \times Thr_{etape-1} + (1 - w1) \times Thr_{ROI} \tag{3.36}$$

$$\begin{aligned} Thr_{ROI} =& w2 \times maxNiveauIntensite(ROI) + \\ & w3 \times moyenneNiveauIntensite(ROI) + \\ & (1 - w2 - w3) \times Thr_{Sk} \end{aligned} \tag{3.37}$$

Détection (4)

Il s'agit de l'étape la plus importante de la chaîne de traitement. Les ROIs trouvées dans la carte de saillance infrarouge sont analysées dans le spectre visible. Cette étape est divisée en deux sous-étapes : 1) génération aléatoire de fenêtres de détection candidates pour couvrir au maximum la ROI et ses alentours pour détecter une éventuelle personne dans la zone étendue et 2) analyse de chaque fenêtre de détection candidate avec un détecteur de personne supervisé.

(1) Pour la première sous-étape, les fenêtres de détection candidates sont générées selon les trois règles suivantes : le centre de chaque fenêtre candidate est aléatoirement choisi à l'intérieur de la ROI, la taille de la fenêtre est aléatoirement choisie entre une échelle minimale et une échelle maximale et le nombre de fenêtres à générer pour chaque ROI dépend de la surface de la ROI.

(2) Pour la seconde étape, les détections candidates sont analysées par le détecteur de personnes PRD. Les résultats de détection sont ensuite fusionnés par une étape de NMS.

3.6.1.1 Expérimentations

Dans cette partie les performances de la chaîne de traitement multimodale sont évaluées en vue aérienne. Dans ce but, nous avons monté le système stéréoscopique hétérogène sur un quadri-rotors Pelican, comme montré Fig.3.34.

La base de données AerialTest2 [15] est utilisée pour tester et comparer les performances de notre chaîne de traitement avec d'autres approches de détection (Fig.3.35) [AerialTest12 2015]. Cette base de données contient des paires d'images visible / infrarouge synchronisées prises avec des points de vue complexes.

Trois aspects différents de la chaîne de traitement sont évalués : 1) l'amélioration du pouvoir de réduction grâce au calcul de la carte de saillance, 2) les performances globales de détection de notre chaîne par rapport à d'autres approches de détection et 3) les temps de calcul de notre chaîne de traitement.

15. *AerialTest2*, Base de données de test contenant des images aériennes infrarouge et visible

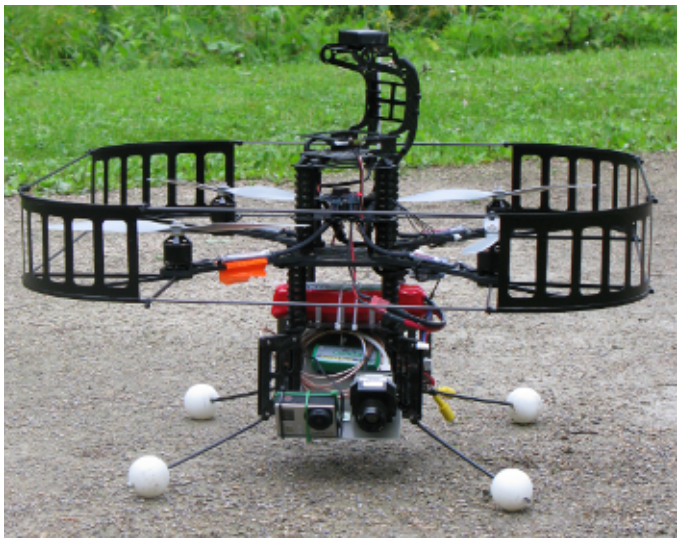


FIGURE 3.34 – Drone Pelican équipé du système d'acquisition stéréoscopique hétérogène visible / infrarouge

Efficacité de la réduction d'espace utilisant les images infrarouge.

Approche de réduction	Min (%)	Max (%)	Moyenne (%)	Écart-type (%)
San-Biagio	0	0,6185	0,05246	0,0811
Salliance+San-Biagio	0,001	0,1009	0,0266	0,0203

TABLE 3.3 – Comparaison des pouvoirs de réduction d'espace de recherche avec, et sans calcul de saillance sur l'infrarouge

Une température trop élevée dans la scène provoque une saturation des images infrarouges ; les images deviennent plus difficiles à analyser : la réduction de l'espace de recherche dans l'infrarouge en est affectée.

Si la réduction de l'espace de recherche est effectuée sur la carte de saillance de l'image infrarouge, alors la réduction de l'espace est plus robuste aux changements de température dans la scène et est plus grande. En effet, la moyenne du pourcentage de réduction est deux fois moins grande (passant de 0,05246% à 0,0266%), ce qui signifie que la réduction est plus grande (Tab.3.3). L'écart-type du pourcentage de réduction est environ quatre fois plus petit (passant de 0,0811% à 0,0203%), ce qui signifie que la réduction est plus robuste aux changements de température.

FIGURE 3.35 – Exemples d'images de la base de données de test aérienne AerialTest2. AerialTest2 est constituée de 141 paires d'images visible et infrarouge de test annotées et de résolution 640x480. Chaque paire contient 2 à 3 personnes prises pour des angles de vue complexes.

Performance de détection de la chaîne de traitement multimodale

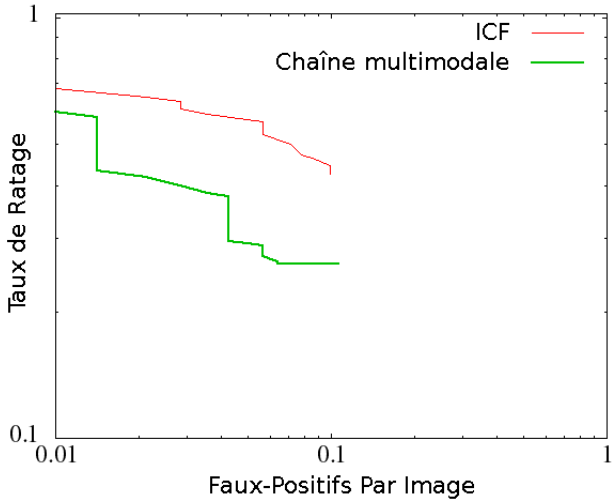


FIGURE 3.36 – Comparaison des performances globales de détection de la chaîne de traitement multimodale avec l'ICF sur la base de données de test AerialTest2

Nous avons comparé les performances globales de détection du détecteur de piétons ICF avec celles de notre chaîne de traitement multimodale sur la base de données de test annotée AerialTest2.

Nous pouvons remarquer que les performances globales de détection du détecteur ICF sont inférieures à celles de notre chaîne de traitement multimodale

(Fig.3.36). Bien évidemment, cela est dû au fait que notre chaîne de traitement utilise le détecteur supervisé PRD (qui est adapté à la vue aérienne). Cependant, nous pouvons également remarquer que le détecteur de piétons ICF n'est pas totalement inefficace car certaines paires d'images de AerialTest2 sont proches de la vue piétonne (Fig.2.37). En effet, AerialTest2 est constituée d'images prises à plus basse altitude et avec un angle d'élévation moins important que pour les images de AerialTest1.

Les paramètres guidant la génération de fenêtres de détection candidates sont : une échelle minimale de 0,5, une échelle maximale de 1,5 et une densité de fenêtres de détection par pixel au carré de 0,04.

FIGURE 3.37 – Comparaison qualitative des détections obtenues avec notre approche (colonne de gauche) et avec l'ICF (colonne de droite) sur la base de données AerialTest2.

Comparaison des temps de calcul

Méthode de détection	ICF	PRD	Notre approche
Temps de calcul	T	1,75×T	1,05×T

TABLE 3.4 – Comparaison des temps de calcul de l'ICF, du PRD et de notre chaîne de traitement multimodale sur la base de données AerialTest2

Les temps de calcul de notre chaîne multimodale se rapproche de ceux du détecteur de piétons ICF (Tab.3.4) : alors que le détecteur PRD est environ 1,75 fois plus lent que le détecteur ICF, notre chaîne est seulement 1,05 fois plus lente que l'ICF. Grâce à notre chaîne de traitement multimodale, la réduction de l'espace de recherche permet d'obtenir des temps de calcul comparables à ceux d'un détecteur de piétons.

3.6.2 Notre approche collaborative de détection multimodale

Dans cette section, nous proposons une approche collaborative dont le principe est de fusionner les scores de détection (Fig.3.38). Nous l'avons nommée : approche de détection multi-modalités ("Multiple Modalities Detection" ou M2D [16] en anglais). Avec le M2D, l'espace de recherche est balayé de manière optimisée et la détection s'adapte dynamiquement.

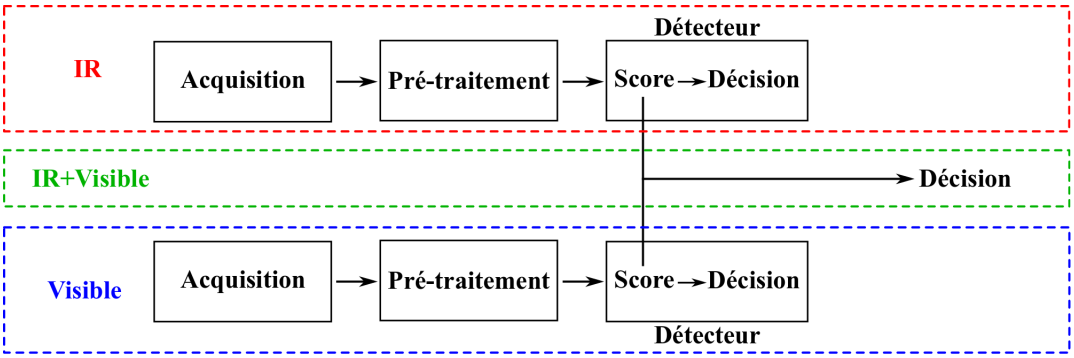


FIGURE 3.38 – Chaîne de traitement basée sur la fusion des scores de détections.

Dans cette section nous détaillerons le M2D : dans un premier temps nous parlerons de la manière dont est exploré l'espace de recherche et dans un second temps nous parlerons du caractère adaptatif de notre approche.

3.6.2.1 Exploration de l'espace bi-modalités

Nous considérons que la détection de personnes dans l'espace de recherche bi-modalités visible/infrarouge est un problème d'optimisation couplé. Cela revient à ne pas travailler sur deux espaces de recherche distincts mais un seul et même espace.

Résoudre un problème bi-objectifs revient à trouver les endroits de la scène où il y a à la fois un score élevé dans l'infrarouge et dans le visible par la recherche de compromis. Avec cette approche, il n'y a pas besoin de fusionner les résultats de détection, nous économisons donc du temps de calcul. Pour résoudre ce problème bi-objectifs nous avons basé notre approche sur l'algorithme d'optimisation par essaim de particules multi-objectifs de Coello et al publié en 2002 ("Multiple Objective Particle Swarm Optimization", en anglais ou MOPSO [17]) [Coello 2002]. Le MOPSO est basé sur l'algorithme PSO de Kennedy et al [Kennedy 1995].

Définissons f_1 et f_2 comme étant respectivement le détecteur visible et le détecteur infrarouge. $\forall d \in \{1,2\}$ $f_d(\vec{p_k})$ est la réponse du détecteur f_d pour la fenêtre

16. *Multiple Modalites Detection*, Approche de détection multi-modalités
17. *Multiple Objective Particle Swarm Optimization*, Essaim de particules multi-objectifs

de détection centrée sur la position de la particule $\vec{p_k}$. L'optimisation simultanée des réponses des deux détecteurs est un problème d'optimisation multi-objectifs comme suit :

$$\vec{f}(\vec{p_k}) = \{f_1(\vec{p_k}), f_2(\vec{p_k})\} \tag{3.38}$$

Ici, contrairement à un problème mono-objectif, l'optimisation du problème doit être considérée différemment : le but est de trouver des solutions de compromis pour deux objectifs plutôt que des solutions idéales pour un unique objectif comme, avec l'algorithme PSO, par exemple. En effet, pour un problème d'optimisation couplé, améliorer la réponse d'un détecteur peut se faire au détriment de la réponse de l'autre détecteur. Dans ce contexte la dominance de Pareto est largement utilisée pour trouver les meilleurs solutions de compromis [Coello 2002].

On dit que la particule $\vec{p_1}$ est Pareto dominée par la particule $\vec{p_2}$ si $\forall d \in \{1, 2\}$ $f_d(\vec{p_1})$ est pire que, ou égale à $f_d(\vec{p_2})$ et si $\exists d' \in \{1, 2\}$ tel que $f_{d'}(\vec{p_1})$ est pire que $f_{d'}(\vec{p_2})$. La Pareto dominance de $\vec{p_1}$ par $\vec{p_2}$ est aussi notée : $\vec{p_2} \succeq \vec{p_1}$. L'ensemble des solutions de compromis est l'ensemble optimal de Pareto P^* :

$$P^* = \{\vec{p_k} \in P / \nexists \vec{p_{k'}} \in P, \vec{p_{k'}} \succeq \vec{p_k}\} \tag{3.39}$$

Le front de Pareto PF^* est l'évaluation de P^* dans l'espace bidimentionel objectif (lignes reliant les points bleus dans Fig.3.41) :

$$PF^* = \{\vec{f}(\vec{p_k}) / \vec{p_k} \in P^*\} \tag{3.40}$$

Les solutions de compromis sont les particules non-Pareto dominées de l'essaim (points bleus dans Fig.3.41). Certaines particules non-Pareto dominées disparaissent aux itérations de mouvement de particules suivantes et d'autres sont plus stables : ce sont ces dernières que nous souhaitons identifier pour trouver les meilleurs candidats de détection.

Le M2D est découpé en deux algorithmes disjoints : le premier algorithme concerne l'exploration de l'espace de recherche bi-modalité et le second concerne le traitement des détections.

Algorithm 8: Algorithme principal du M2D

Data: Détecteurs $\vec{f}$ et l'espace de recherche S
Result: Détections D

1 $iteration = 0$
2 Initialisation aléatoire dans S d'un ensemble P de particules
3 Calculer P^*
4 **while** $iteration < max\_iterations$ **do**
5 **foreach** $\vec{p_k} \in P$ **do**
6 $\vec{g}$ = la particule non-Pareto dominée la plus proche de $\vec{p_k}$
7 Mettre à jour $\vec{v_k}$ et $\vec{p_k}$
8 Calculer $\vec{f}(\vec{p_k})$ dans S
9 Mettre à jour M_1 et M_2 (Equ.3.45 et Equ.3.46)
10 Mettre à jour $\vec{b_k}$ (avec la Pareto-dominance)
11 **end**
12 Appeler Algorithme "Test de convergence" (si il retourne $true$, aller à ligne 2)
13 Mettre à jour σ_{ref} (Equ.3.47)
14 $iteration = iteration + 1$
15 **end**

Algorithm 9: Test de convergence

1 Mettre à jour P^*
2 **foreach** $\vec{p_k} \in P^*$ **do**
3 **if** $\vec{p_k}$ *survie et contracte localement l'essaim* **then**
4 $ctr_k = ctr_k + 1$
5 **end**
6 **if** $ctr_k \geq min\_contractions$ **then**
7 **if** $m_1(\vec{p_k}) > mf_1(\sigma_{ref})$ *et* $m_2(\vec{p_k}) > mf_2(\sigma_{ref})$ **then**
8 $D = D \cup \{\vec{p_k}\}$
9 $S = S - \{ROI(\vec{p_k})\}$
10 $iteration = 0$
11 **end**
12 Retourner $true$
13 **end**
14 **end**
15 Retourner $false$

Les vecteurs vitesse et position $\vec{v_k}$ et $\vec{p_k}$ sont mis à jour de la même manière que pour l'algorithme PSO. Le vecteur "meilleure position" $\vec{b_k}$ de la particule k est mis à jour en utilisant la dominance de Pareto.

Le paramètre $max\_iterations$ correspond au nombre maximum d'itérations effectuées lorsqu'aucune nouvelle détection candidate n'est trouvée. Le paramètre $min\_contractions$ est le nombre minimum de survies combiné à une contraction locale nécessaire pour qu'une détection candidate soit considérée comme une détection.

Le critère de Pareto dominance est largement utilisé dans les deux algorithmes. Le M2D utilise un critère d'arrêt spécifique : le nombre de survies combinées à une contraction locale de l'essaim (ctr_k ligne 4 de Alg.9). On dit qu'une particule non-Pareto dominée a survécu si elle est toujours présente dans P^* après une itération de mouvement des particules. On dit qu'une particule non-Pareto dominée a commis une contraction locale de l'essaim si une autre particule non-Pareto dominée apparaît dans son voisinage proche après une itération de mouvement des particules (voir Fig.3.39) ; concrètement cela correspond à la découverte d'une autre solution de compromis dans le voisinage. Au début le voisinage de la particule non-Pareto dominée est l'espace de recherche entier. L'idée générale est de trouver les particules non-dominées ayant survécu et aidé à la contraction de l'essaim un nombre suffisant de fois (plus que $min\_contractions$, ligne 8 de Alg.9). Les détections candidates sont ensuite traitées dans le second algorithme pour être vérifiées. Concrètement, pour chaque détection candidate les pourcentages de classifieurs faibles passés pour le détecteur f_1 et le détecteur f_2 sont comparés à des pourcentages minimaux requis $mf_1(\sigma_{ref})$ et $mf_2(\sigma_{ref})$ (ligne 7 de Alg.9). Les pourcentages minimaux de classifieurs faibles à passer sont fonction de σ_{ref}. Une détection candidate $\vec{p_k}$ est considérée comme étant une vraie détection (gardée en mémoire dans D, ligne 8 de Alg.9) si les deux pourcentages sont au dessus de leurs minimaux respectifs donnés par les fonctions sigmoïdes mf_1 et mf_2 illustrées dans la Fig.3.40. Les sigmoïdes associées aux détecteurs f_1 et f_2 sont respectivement construites comme défini dans Equ.3.41 et dans Equ.3.42.

$$mf_1(\sigma_{ref}) = \frac{1}{1 + exp^{-5\times\sigma_{ref}}} \tag{3.41}$$

$$mf_2(\sigma_{ref}) = 1 - \frac{1}{1 + exp^{-5\times\sigma_{ref}}} \tag{3.42}$$

À la ligne 9 de Alg.9 nous supprimons localement la zone de l'espace de recherche correspondant à la détection. Concrètement, nous gardons en mémoire les coordonnées de la fenêtre de détection et nous ignorons la zone lors de la réinitialisation des particules dans l'espace et lors des itérations de mouvement des particules à l'aide d'un test d'intersection boîte contre boîte ("Axis Aligned Bounding Box test", en anglais).

Nous avons testé plusieurs fonctions (fonction affine, et gaussienne) et nous avons choisi d'utiliser la fonction sigmoïde pour jauger la contribution minimale de

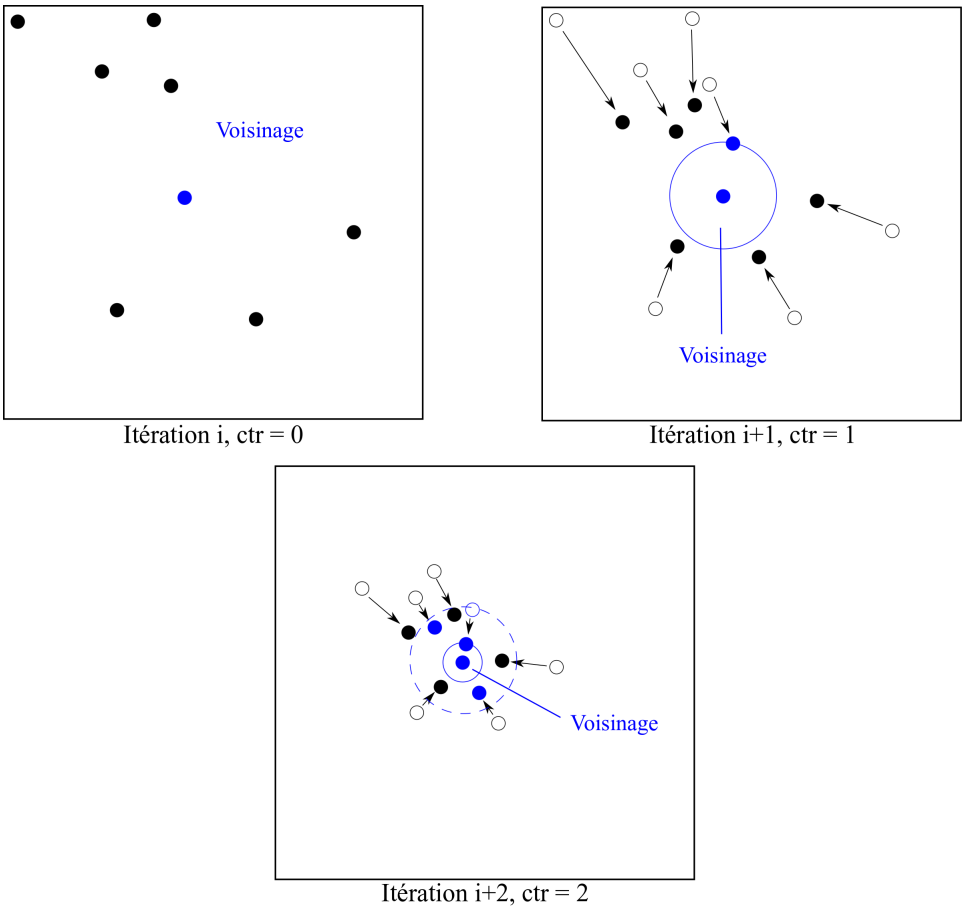


FIGURE 3.39 – Exemple d'une particule non-Pareto dominée survivant et contractant localement l'essaim deux fois de suite.

chaque détecteur en fonction de σ_{ref}. Cela pour plusieurs raisons : 1) pour $\sigma_{ref} = 0$ la fonction est égale à la moitié de sa valeur maximale, c'est-à-dire 0.5, ce qui permet une contribution égale des détecteurs visible et de l'infrarouge 2) la fonction tend progressivement vers 1 en $+\infty$ et tend progressivement vers 0 en $-\infty$ et 3) la fonction est symétrique par rapport au centre de symétrie $(0, 0.5)$. Nous verrons dans la partie suivante en quoi ces propriétés sont importantes dans notre cas.

3.6.2.2 Adaptation dynamique pour une détection robuste.

Le M2D s'adapte dynamiquement aux conditions d'acquisition (Tab.3.2) grâce à la valeur σ_{ref} qui est utilisée comme une référence pour chaque paire d'images acquises (Equ.3.47).

$$m_1(\vec{p_k}) = \% \text{ de classifieurs faibles passés avec } f_1 \text{ sur } \vec{p_k} \tag{3.43}$$

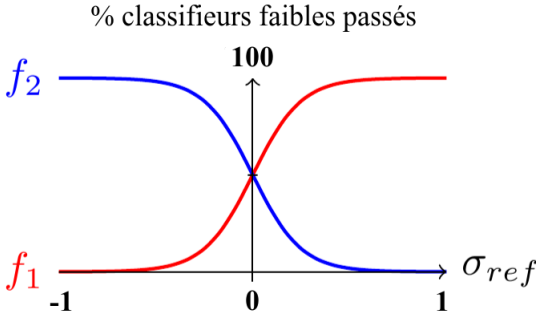


FIGURE 3.40 – Sigmoïdes représentant le pourcentage minimal de classifieurs faibles à passer pour chaque détecteur et pour toutes les valeurs de σ_{ref} variant entre -1 et 1.

$$m_2(\vec{p_k}) = \%\text{ de classifieurs faibles passés avec } f_2 \text{ sur } \vec{p_k} \tag{3.44}$$

$$M_1 = \max_{\vec{p_k} \in P}(m_1(\vec{p_k})) \tag{3.45}$$

$$M_2 = \max_{\vec{p_k} \in P}(m_2(\vec{p_k})) \tag{3.46}$$

$$\sigma_{ref} = \frac{M_1^2 - M_2^2}{M_1^2 + M_2^2} \tag{3.47}$$

Cette valeur est mise à jour à chaque itération de mouvement des particules (algorithme 1, ligne 13 de Alg.8), σ_{ref} converge rapidement vers une valeur stable après quelques itérations de mouvement. Nous nous sommes inspiré du travail de Mostaghim et al qui utilisèrent σ dans leurs travaux pour une toute autre tâche : explorer plus rapidement le front de Pareto [Mostaghim 2003]. À noter que la définition de σ_{ref} nécessite impérativement que les deux détecteurs utilisent des classifieurs de type SoftCascade.

La valeur σ_{ref} est liée à la capacité des détecteurs à traiter la scène. Lorsque les deux détecteurs ont des réponses équivalentes alors, les valeurs M_1 et M_2 ont des valeurs similaires (Equ.3.45 et Equ.3.46). Dans ce cas, σ_{ref} est très proche de zero (Fig.3.41.1). Lorsque qu'un détecteur ne fonctionne pas aussi bien qu'un autre (M_1 et M_2 très différents) alors la plupart des particules sont rassemblées près de l'axe du détecteur fonctionnant le mieux dans l'espace objectif bidimentionnel (Fig.3.41.2 ou Fig.3.41.3). Dans ce cas, la valeur de σ_{ref} est différente de 0 et son signe donne une indication sur le détecteur qui est en défaut. Si sa valeur est comprise entre 0 et 1 cela signifie que le détecteur infrarouge f_2 répond moins bien que le détecteur visible f_1 pour la scène donnée. Si σ_{ref} est compris entre 0 et -1 cela signifie que le détecteur visible f_1 répond moins bien que le détecteur infrarouge f_2 pour la scène donnée.

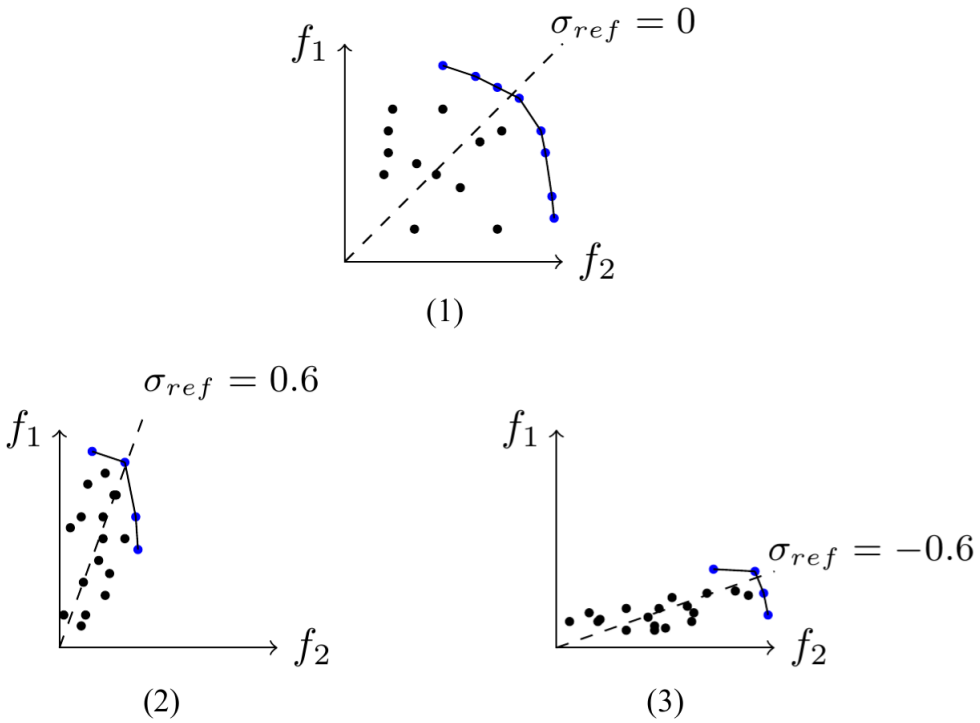


FIGURE 3.41 – Projections des particules dans l'espace objectif bidimmensionel <f_1,f_2>, en noir les particules Pareto dominées, en bleu les particules non-Pareto dominées.

La valeur σ_{ref} est utilisée pour moduler le nombre minimal de classifieurs faibles à passer pour chaque détecteur pour qu'une détection candidate soit validée comme étant une détection. Par exemple, lorsque que le détecteur f_2 et le détecteur f_1 ont des réponses équivalentes on demandera à ce que 50% minimum des classifieurs faibles des deux détecteurs soient passés. Si le détecteur f_2 est totalement inefficace, on demandera à ce que tous les classifieurs faibles du détecteur f_1 soient passés et vice-versa.

Cette approche permet une complémentarité des résultats obtenus sur l'infrarouge et sur le visible. Elle permet également de gérer de manière totalement dynamique l'inefficacité soudaine d'un des détecteurs et le dysfonctionnement soudain d'une des caméras dû aux conditions d'acquisition, par exemple (Tab.3.2).

Il est pertinent de se baser sur les valeurs M_1 et M_2 pour avoir une bonne estimation des capacités des détecteurs à traiter une scène donnée. En effet, M_1 et M_2 sont calculés à partir d'un très grand nombre de particules positionnées de manière uniformément aléatoire dans l'espace de recherche.

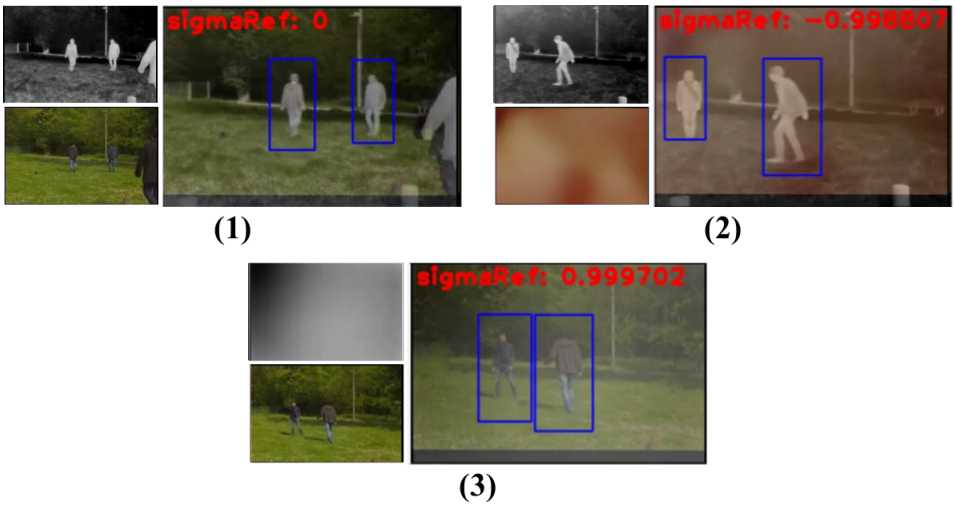


FIGURE 3.42 – Illustration de l'adaptation dynamique du M2D : 1) lorsque les deux capteurs fonctionnent normalement σ_{ref} est nul et les deux modalités sont explorées en même temps, 2) lorsque la caméra visible est obstruée σ_{ref} est proche de -1 alors on donne plus d'importance aux résultats du détecteur infrarouge et 3) lorsque la caméra infrarouge est obstruée σ_{ref} est proche de 1 alors on donne plus d'importance aux résultats du détecteur visible.

3.6.2.3 Expérimentations

L'approche de détection M2D a été testée pour trois scénarios distincts (Fig.3.43) : 1) un scénario où les caméras visible et infrarouge fonctionnent correctement, 2) un scénario où seule la caméra infrarouge fonctionne correctement et 3) un scénario où seule la caméra visible fonctionne correctement.

Les performances globales de détection obtenues dans le scénario (1) sont un compris entre les performances obtenues dans le scénario (2) et celles obtenues dans le scénario (3). En effet, pour un nombre de FPPI de 0,03 on observe que le M2D dans le scénario (1) a un Taux de Ratage de 0,51 qui est plus faible que le M2D dans le scénario (2) qui a un Taux de Ratage de 0,6, car le détecteur visible est fonctionnel dans ce scénario là. On observe également ce phénomène pour les autres valeurs. À noter que le détecteur visible est plus sensible que le détecteur infrarouge (sa courbe ROC associée est plus basse). Le compromis sur les performances de détection est fait au bénéfice d'une collaboration des détecteurs qui est rapide (voir Tab.3.6) et dynamiquement adaptative ; la détection continue si une des caméras (ou un des détecteurs associés) ne fonctionne pas. Nous avons simulé le dysfonctionnement des caméras en remplaçant les images de la base de données de test AVIS de la caméra défectueuse par des images noires, vide d'information. À noter que, étant donnée que l'approche M2D est non-déterministe nous avons lancé les tests plusieurs fois (10 fois) et nous avons gardé les résultats moyens pour

tracer les courbes.

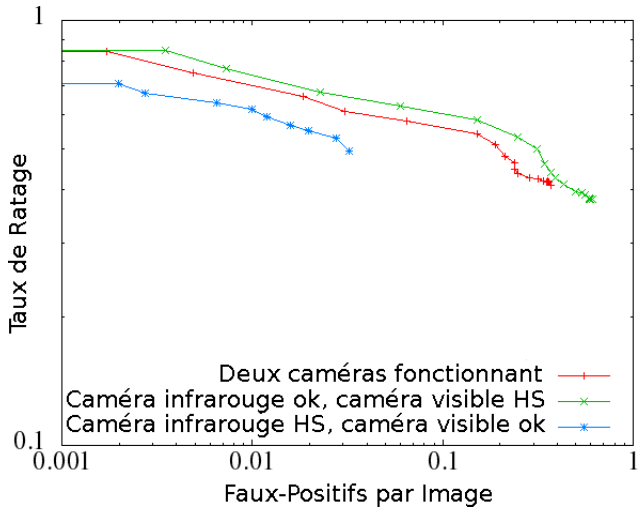


FIGURE 3.43 – Comparaison des performances globales de détection du M2D pour trois scénarios : caméras visible et infrarouge fonctionnant, caméra infrarouge fonctionnant et caméra visible ne fonctionnant pas et caméra infrarouge ne fonctionnant pas et caméra visible fonctionnant.

Scénarios	VIS ok IR ok	VIS HS IR ok	VIS ok IR HS
α moyen Taux de Ratage	0,014	0,012	0,004
α moyen FPPI	0,007	0,005	0,182

TABLE 3.5 – Écart-types α moyens pour le Taux de Ratage et le nombre de FPPI pour les trois scénarios.

Les écart-types moyens du Taux de Ratage et du nombre de FPPI des 10 tests effectués pour tracer les courbes de la Fig.3.43 montrent que notre approche non-déterministe est stable (Tab.3.5). Le Taux de Ratage ne varie au maximum que de 1,4% en moyenne et le nombre de FPPI varie au maximum que de 0,182 FFPI. Nous pouvons cependant constater que lorsque seul le détecteur visible est fonctionnel l'écart-type moyen du Taux de Ratage est de 0,004 (soit environ trois fois moins que pour les deux autres cas) et que l'écart-type moyen du nombre de FPPI est 36,4 fois plus important que pour les deux autres cas. Ces observations amènent à penser que notre détecteur visible est plus sensible que notre détecteur infrarouge.

La Fig.3.44 montre comment les performances de détection du M2D sur la base de données de test AVIS changent en fonction de la concentration de particules dans l'espace de recherche multimodal. Nous pouvons observé qu'au delà de 48 μ particules par pixel au carré les performances sont similaires. Augmenter

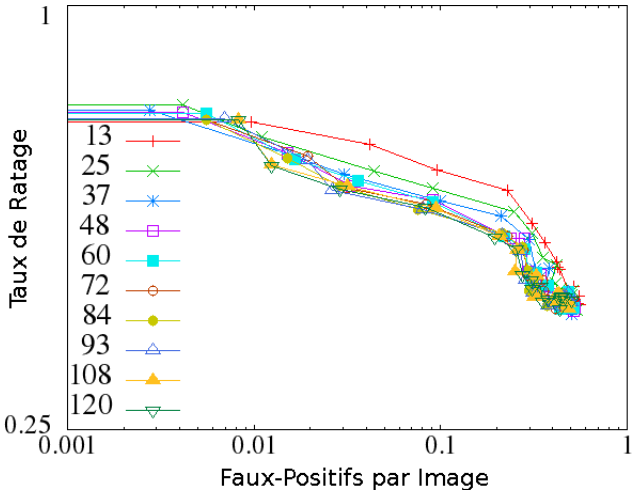


FIGURE 3.44 – Performances de détection de l'approche M2D en fonction de la concentration de particules dans l'espace de recherche (en μ particules par pixel au carré).

la concentration de particules dans l'espace fait accroître les temps de calcul. Par conséquent, choisir une concentration de 48 μ particules par pixel au carré est un compromis idéal entre performance de détection et temps de calcul.

Ratios de temps	ACF	IRACF	ACF+IRACF	M2D
ACF	1	0,96	1,96	**1,14**
ACF+IRACF	0,511	0,49	1	**0,58**

TABLE 3.6 – Comparaison des ratios de temps de calcul entre l'ACF, l'IR-ACF, l'ACF+IR-ACF et le M2D. Par exemple, le ratio des temps de calcul du détecteur IR-ACF/SoftCascade sur le détecteur ACF/SoftCascade est 0,96, cela signifie que le détecteur IR-ACF/SoftCascade est 4% plus rapide que le détecteur ACF/SoftCascade.

L'approche M2D avec une concentration de 48 μ particules par pixel carré est 0,58 fois plus rapide que l'approche ACF+IR-ACF consistant à balayer exhaustivement l'image visible et l'image infrarouge à chaque fois (Tab.3.6). À noter que pour l'approche ACF+IR-ACF nous n'avons pas ajouté le temps de calcul nécessaire à la fusion finale des résultats du ACF et du IR-ACF. Les tests ont été effectués sur la base de données de test AVIS.

Les paramètres choisis pour le M2D sont : un nombre de 100 particules, un paramètre $max\_iterations$ égal à 300 et un paramètre $min\_contractions$ égal à 5. Les pyramides d'images des détecteurs IR-ACF/SoftCascade et ACF/SoftCascade ont été définies toutes les deux sur deux octaves (16 niveaux) à partir de l'échelle

minimale 0,2 (allant donc jusqu'à l'échelle 0,8) ; les personnes étaient proches du système de vision dans la scène utilisée pour les tests.

3.6.3 Conclusion

Pour résumer :

1. Nous avons proposé une chaîne de co-entraînement permettant de générer autant de nouvelles données d'entraînement que souhaité de manière semi-supervisée. Les nouvelles données d'entraînement acquises permettent de renforcer l'entraînement des détecteurs de personnes visible et infrarouge. Quelques itérations de co-entraînement permettent de rendre plus performant les détecteurs. Notre approche de co-entraînement est originale car nous utilisons la mesure du "caractère objet" dans le spectre infrarouge.
2. Nous avons proposé une chaîne de traitement multimodale spécifique basée sur une réduction de l'espace de recherche et par conséquent, des temps de calcul. La chaîne de traitement multimodale permet une détection multi-vues de personnes avec des temps de calcul comparables à ceux d'une détection de personnes de type piéton.
3. Nous avons également proposé une approche collaborative de détection permettant une collaboration des détecteurs infrarouge et visible qui s'appelle le M2D. La recherche des solutions est guidée par les meilleurs compromis de résultats visible / infrarouge. L'approche M2D est capable de s'adapter dynamiquement aux dysfonctionnements d'un des capteurs ou d'un des détecteurs, sans intervention d'un utilisateur. Si un des détecteurs peine à détecter les personnes présentes dans une image, le deuxième détecteur agit comme un détecteur de secours pour localiser les personnes. À noter que le M2D ne nécessite aucune étape de fusion des données ni d'étape de NMS des détections. De plus, le M2D peut en théorie être utilisé avec un nombre quelconque de détecteurs (avec certaines adaptations).

Conclusion et perspectives

Dans ce mémoire de thèse nous avons présenté de nouvelles approches de détection de personnes à partir de vues aériennes. Ces travaux de thèse s'inscrivent dans le cadre du projet SEARCH dont le but est de développer des moyens rapides de recherche automatique de personnes disparues ou en difficultés. La mission est réalisée à l'aide d'une flottille de drones équipés chacun d'un système de vision. Dans ce cadre, nous avons développé des algorithmes originaux d'analyse d'images pour gérer la complexité des angles de vue du système de vision, les contraintes de temps de calcul inhérentes à une application embarquée ainsi que les conditions de luminosité changeantes (jour/nuit).

Dans le premier chapitre, nous avons détaillé le projet de recherche SEARCH financé par la région Picardie. Les spécificités propres au projet, ainsi que celles du vol aérien à basse altitude nous ont permis de définir un ensemble de contraintes à respecter pour réaliser un détecteur de personnes à partir de drones qui soit adapté.

Dans le second chapitre, nous avons fait une revue des travaux existant en détection automatique de piétons. Bien qu'utilisées dans un contexte différent du notre, les techniques de détection de piétons présentent de nombreux avantages qu'il nous a semblé intéressant d'exploiter et d'adapter à notre contexte. En effet, les détecteurs de piétons actuels sont rapides (le temps réel vidéo est atteint pour certaines configurations), robustes à des changements locaux de luminosité et fonctionnels pour plusieurs échelles (concrètement, les personnes peuvent être détectées pour plusieurs distances). Étape par étape, nous avons détaillé les processus d'apprentissage et de détection supervisé de personnes. Ainsi, nous avons montré comment construire un modèle mathématique de classification à partir d'images d'entraînement. Nous avons détaillé les approches les plus performantes et les plus référencées de l'état de l'art. Cela concerne, entre autre : le calcul des caractéristiques visuelles, l'apprentissage ainsi que la réduction de l'espace de recherche.

Dans le troisième chapitre, nous avons fait, dans un premier temps, un état de l'art des travaux existants en détection de personnes à partir de drones. Au vu des contraintes que nous avons définies au chapitre 1 nous avons exposé les limites des approches existantes pour la détection de personnes à partir de drones mais également les limites des approches de détection de piétons appliquées au cas aérien basse altitude. Partant d'un détecteur de piétons de référence, nous avons montré qu'il était possible de l'adapter aisément au cas aérien par un entraînement multi-élévations et par des réductions de l'espace de recherche dans le but de réduire les temps de calcul, nécessaire pour une bonne réactivité lors d'un vol à basse altitude. L'entraînement multi-élévations permet une amélioration significative des

performances de détection en vol, car cela permet de prendre en compte l'angle d'élévation lors de l'apprentissage. Dans un second temps, nous avons voulu aller plus loin en proposant une approche de détection robuste aux variations des angles de roulis et de tangage combinés du système de vision. Pour cela, nous avons proposé un algorithme d'apprentissage qui est en mesure d'apprendre et d'optimiser un très grand nombre de vues de personnes quelque soit l'angle de roulis et de tangage du système de vision. Nous avons adapté deux détecteurs de piétons parmi les plus performants et les plus rapides de l'état de l'art au cas aérien. Les tests indiquent que notre approche permet une détection de personnes au sol robuste aux variations d'angles du système de vision embarqué sur le drone. Les tests indiquent également que notre approche est rapide et multi-échelles. En définitive, notre approche combine les avantages de la détection de piétons modernes à une robustesse multi-vues.

Dans le quatrième chapitre, nous avons étendu la détection de personnes au spectre infrarouge lointain et long. Pour cela, nous avons conçu un système stéréoscopique hétérogène combinant caméras visible et infrarouge. Les axes optiques des deux caméras sont coplanaires et parallèles, permettant ainsi un recalage pixélique des objets de la scène à l'infini. Chaque paire d'images visible et infrarouge est préalablement synchronisée temporellement et spatialement. Dans un premier temps, nous avons proposé une approche semi-supervisée de co-entraînement multimodale. Notre but était de développer une approche de renforcement mutuel des détecteurs visible et infrarouge. Dans cet objectif, nous avons utilisé une mesure bas-niveau du caractère objet thermique ; cette mesure permettant de générer un premier ensemble de nouvelles images d'entraînement, un filtrage élimine les nouvelles images d'entraînement qui pourraient faire régresser les performances. Ainsi, nous avons montré qu'il était possible de faire se renforcer mutuellement un détecteur de personnes opérant dans le spectre visible avec un détecteur de personnes opérant dans le spectre infrarouge lointain et long. Dans un second temps, nous avons proposé une approche bimodale de détections. Les scores des détecteurs visible et infrarouge guident simultanément l'exploration de l'espace de recherche des solutions. Avec notre approche, les détections finales sont les meilleurs compromis de scores des détecteurs visible et infrarouge. Le compromis recherché entre le score du détecteur visible et celui du détecteur infrarouge est fonction des sensibilités de ceux-ci. Ainsi, si un détecteur est moins sensible qu'un autre pour une scène donnée, on sera plus exigeant avec le détecteur le plus sensible. Ce principe permet une adaptation dynamique de la sensibilité des détecteurs à l'environnement : si la nuit tombe, seul le détecteur infrarouge fonctionnera, s'il y a saturation d'infrarouge, seul le détecteur visible fonctionnera et dans les cas intermédiaires une contribution égale sera demandée aux deux détecteurs. À noter que cette approche est également rapide : bien que deux détecteurs soient exécutés en même temps, l'approche est un peu plus rapide que l'exécution d'un seul détecteur

avec une approche classique.

Perspectives

D'un point de vue scientifique les perspectives sont nombreuses : L'adaptation d'algorithmes de détection issus de la recherche sur les systèmes ADAS nous a permis dans cette thèse de proposer un détecteur de personnes en vue aérienne performant. Nous pensons que la plupart des approches développées pour la détection de piétons sont d'un grand intérêt pour la conception d'algorithmes de détection dans le cas aérien. Pour nous, l'avenir de la détection de personnes en vue aérienne passe par les avancées faites dans le domaine de la détection de piétons.

L'approche de co-entraînement multimodale est une réponse appropriée au manque d'images d'entraînement disponibles. Dans cette thèse, nous avons utilisé notre approche de co-entraînement multimodale pour générer suffisamment de données d'entraînement infrarouge et ainsi permettre un entraînement performant de notre détecteur de personnes opérant dans le spectre infrarouge. Cela nous a permis de contourner le problème du manque de données d'entraînement infrarouge disponibles. Nous pensons que ce principe peut être transposé à d'autres types de systèmes de vision. D'une manière générale, la disponibilité des données d'entraînement est un vrai problème, surtout depuis l'émergence de l'apprentissage profond, qui requière un nombre important de données d'entraînement pour être performant.

Avec le développement récent des caméras infrarouge de nouvelle génération les techniques issues de la vision par ordinateur dans le spectre visible vont de plus en plus être utilisées dans le spectre infrarouge. Parmi ces techniques, nous pensons que l'analyse du caractère objet, comme la mesure CAO proposée dans cette thèse, a un grand potentiel, notamment pour la localisation automatique d'êtres vivants dans la scène.

Dans cette thèse, nous avons proposé une approche de fusion différente de celles généralement envisagées dans la littérature : notre approche est basée sur les compromis de scores venant de détecteurs différents. Nous pensons que ce principe peut être appliqué pour d'autres systèmes de vision hétérogène que celui proposé dans cette thèse et pour détecter autre chose que des êtres humains. Nous pensons également qu'il serait intéressant d'étendre notre approche à un nombre quelconques de détecteurs opérant, ou non, sur des modalités différentes. L'avantage de notre approche de fusion des scores est que les temps de calcul sont moins impactés lorsque l'on ajoute un détecteur au système qu'avec une approche par fusion des détections, par exemple.

En termes d'applications, les perspectives sont nombreuses tant sur le plan civil que militaire. Les applications civiles : outre la surveillance des bords de mer pour

prévenir les noyades et la recherche de personnes disparues en forêt, notre détecteur de personnes au sol pourrait être utilisé pour la surveillance de bâtiments sensibles (centrales nucléaires, ambassades, entrepôts, etc.). Elle pourrait être également d'une grande aide après une catastrophe naturelle de grande ampleur, pour localiser les survivants. De plus, grâce à notre approche adaptative bi-modalités les personnes sont détectables 24h sur 24h et quelles que soient les conditions climatiques, ce qui rend plus flexible l'usage du drone pour la détection de personnes. Les applications militaires : la détection de personnes au sol peut être utilisée comme un système d'aide à la visée par les opérateurs du drone dans le but frapper avec plus de précision les cibles et pour limiter les dommages collatéraux.

Références bibliographiques

[Achanta 2009] Radhakrishna Achanta, Sheila Hemami et Francisco Estrada. *Frequency-tuned Salient Region Detection*. In Computer Vision and Pattern Recognition (CVPR), 2009. (Cité en pages 51 et 55.)

[Achanta 2010] Radhakrishna Achanta et Süsstrunk Sabine. *Saliency Detection Using Maximum Symmetric Surround*. In Internation Conference on Image Processing (ICIP), 2010. (Cité en pages 70 et 141.)

[AerialTest1 2015] AerialTest1. *Base de données de test en vue aérienne AerialTest1*. http://home.mis.u-picardie.fr/~p-blondel/papers/data/aerialtest1.zip, 2015. Accédé en : 2015-09. (Cité en pages 67, 70 et 72.)

[AerialTest12 2015] AerialTest12. *Base de données multimodale de test en vue aérienne AerialTest2*. http://home.mis.u-picardie.fr/~p-blondel/papers/data/aerialtest2.zip, 2015. Accédé en : 2015-09. (Cité en page 142.)

[Alexe 2010] Bogdan Alexe, Thomas Deselaers et Vittorio Ferrari. *What is an object ?* In Computer Vision and Pattern Recognition, 2010. (Cité en pages 55 et 56.)

[Andriluka 2010] M Andriluka, P Schnitzspan, J Meyer, S Kohlbrecher, K Petersen, O von Stryk, S Roth et B Schiele. *Vision based victim detection from unmanned aerial vehicles*. In Conference on Intelligent Robots and Systems (IROS), 2010. (Cité en page 59.)

[ATI 2015] ATI. *Base de données d'entraînement infrarouge ATI*. http://home.mis.u-picardie.fr/~p-blondel/papers/data/ATI.zip, 2015. Accédé en : 2015-09. (Cité en pages 123 et 137.)

[ATV 2015] ATV. *Base de données d'entraînement visible ATV*. http://home.mis.u-picardie.fr/~p-blondel/papers/data/ATV.zip, 2015. Accédé en : 2015-09. (Cité en pages 124 et 137.)

[AVIS 2015] AVIS. *Base de données multimodale de test AVIS*. http://home.mis.u-picardie.fr/~p-blondel/papers/data/AVIS.zip, 2015. Accédé en : 2015-09. (Cité en page 137.)

[Bardon 1998] JP. Bardon et B. Cassagne. Temerature de surface - measure par contact. Technique de l'Ingénieur - r2730, 1998. (Cité en pages 95 et 96.)

[Bebis 2000] George Bebis, Satishkumar Uthiram et Michael Georgiopoulos. *Face detection and verification using genetic search*. International Journal on Artificial Intelligence Tools, 2000. (Cité en page 43.)

[Bergeron 2004] Alain Bergeron, Hubert Jerominek, Jean Lacoursiere, Nichola Desnoyers, Christine Alain et Philips Laou. *Novel lightweight uncooled thermal weapon sight*. In Defense and Security, pages 402–411. International Society for Optics and Photonics, 2004. (Cité en page 100.)

[Bishop 2006] Christopher Bishop. Pattern recognition and machine learning. 2006. (Cité en pages 27 et 28.)

[Blum 1998] Avrim Blum et Tom Mitchell. *Combining Labeled and Unlabeled Data with Co-training*. In Conference on Computational Learning Theory, 1998. (Cité en page 120.)

[Bouguet] Bouguet. *Toolbox Mattlab de Jean-Yves Bouguet pour la calibration de caméras*. `http://www.vision.caltech.edu/bouguetj/calib_doc/`. Accédé en : 2015-09. (Cité en page 107.)

[Bourdev 2005] Lubomir Bourdev et Jonathan Brandt. *Robust Object Detection Via Soft Cascade*. In Conference on Computer Vision and Pattern Recognition, 2005. (Cité en pages 9, 32 et 33.)

[Brehar 2014] R. Brehar, C. Vancea et S. Nedevschi. *Pedestrian detection in infrared images using Aggregated Channel Features*. In Intelligent Computer Communication and Processing, 2014. (Cité en pages 112, 113 et 124.)

[Brits 2002] Riaan Brits, Andries P Engelbrecht et F Van den Bergh. *A niching particle swarm optimizer*. In Asia-Pacific conference on simulated evolution and learning, 2002. (Cité en page 44.)

[Bruhat 1968] G. Bruhat et A. Kastler. Thermodynamique. Masson et Cie, 1968. (Cité en pages 95 et 96.)

[Caccavale 2014] Fabrizio Caccavale, Gerardo Giglio, Giuseppe Muscio et Francesco Pierri. *Adaptive Control for UAVs Equipped with a Robotic Arm*. In IFAC World Congress, 2014. (Cité en page 1.)

[Cheng 2014] Ming-Ming Cheng, Ziming Zhang, Wen-Yan Lin et Philip Torr. *BING : Binarized normed gradients for objectness estimation at 300fps*. In Computer Vision and Pattern Recognition, 2014. (Cité en pages 55 et 56.)

[Cie 1931] Cie. *Commission Internationale de l'Eclairage Proceedings*. Rapport technique, Cambridge University Press, Cambridge, 1931. (Cité en page 18.)

[Coello 2002] Carlos A. Coello et M.S. Lechuga. *MOPSO : a proposal for multiple objective particle swarm optimization*. In Congress on Evolutionary Computation, 2002. (Cité en pages 146 et 147.)

[Cortes 1995] Corinna Cortes et Vladimir Vapnik. *Support-Vector Networks*. In Machine Learning, pages 273–297, 1995. (Cité en page 25.)

[CTAVIS-1 2015] CTAVIS-1. *Base de données multimodale de co-entraînement CTAVIS-1*. http://home.mis.u-picardie.fr/~p-blondel/papers/data/CTAVIS-1.zip, 2015. Accédé en : 2015-09. (Cité en page 136.)

[CTAVIS-2 2015] CTAVIS-2. *Base de données multimodale de co-entraînement CTAVIS-2*. http://home.mis.u-picardie.fr/~p-blondel/papers/data/CTAVIS-2.zip, 2015. Accédé en : 2015-09. (Cité en page 136.)

[CTAVIS-3 2015] CTAVIS-3. *Base de données multimodale de co-entraînement CTAVIS-3*. http://home.mis.u-picardie.fr/~p-blondel/papers/data/CTAVIS-3.zip, 2015. Accédé en : 2015-09. (Cité en page 136.)

[Dalal 2005] N Dalal et B Triggs. *Histograms of Oriented Gradients for Human Detection*. In Conference on Computer Vision and Pattern Recognition, 2005. (Cité en pages 12, 13, 22, 23, 61, 73, 74 et 78.)

[Dalal 2006] Navneet Dalal. *Finding people in images and videos*. PhD thesis, Institut National Polytechnique de Grenoble-INPG, 2006. (Cité en page 41.)

[Davis 2005] J.W. Davis et M.A. Keck. *A Two-Stage Template Approach to Person Detection in Thermal Imagery*. In Workshops on Application of Computer Vision, 2005. (Cité en page 137.)

[Derome 2014] M. Derome, A. Plyer, M. Sanfourche et G. Le Besnerais. *Real-Time Mobile Object Detection Using Stereo*. In International Conference on Control, Automation, Robotics and Vision, 2014. (Cité en page 47.)

[Dollár 2009a] P. Dollár, C. Wojek, B. Schiele et P. Perona. *Pedestrian detection : A benchmark*. In Conference on Computer Vision and Pattern Recognition, 2009. (Cité en pages 9, 12, 22, 23, 58, 61 et 78.)

[Dollár 2009b] Piotr Dollár, Zhuowen Tu, Pietro Perona et Serge Belongie. *Integral Channel Features*. In Proceedings of the British Machine Vision Conference, 2009. (Cité en pages 17, 18, 78, 86 et 115.)

[Dollár 2010] P. Dollár, Belongie S. et P. Perona. *The Fastest Pedestrian Detector in the West*. In British Machine Vision Conference, 2010. (Cité en pages 21, 22 et 61.)

[Dollár 2012] Piotr Dollár, Christian Wojek, Bernt Schiele et Pietro Perona. *Pedestrian detection : an evaluation of the state of the art*. Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 2012. (Cité en page 68.)

[Dollár 2014] P Dollár, R Appel, S Belongie et P Perona. *Fast Feature Pyramids for Object Detection Â´*. Transactions on pattern analysis and machine intelligence (TPAMI), pages 1–14, 2014. (Cité en pages 20, 21 et 124.)

[ElevationTest 2015] ElevationTest. *Base de données de pour tester la robustesse á l'élévation.* http://home.mis.u-picardie.fr/~p-blondel/papers/data/elevation_test.zip, 2015. Accédé en : 2015-09. (Cité en pages 66 et 87.)

[Elhamifar 2011] E. Elhamifar et R. Vidal. *Robust classification using structured sparse representation*. In Computer Vision and Pattern Recognition, 2011. (Cité en page 113.)

[E.Schapire 1999] Robert E.Schapire et Singer Yoram. *Improved Boosting Algorithms Using Confidence-rated Predictions*. Machine Learning, 1999. (Cité en page 81.)

[ETHZ 2014] ETHZ. *Base de donnée infrarouge proposée par l'ETHZ.* http://projects.asl.ethz.ch/datasets/doku.php?id=ir:iricra2014, 2014. Accédé en : 2015-09. (Cité en page 123.)

[Felzenszwalb 2005] Pedro F. Felzenszwalb et Daniel P. Huttenlocher. *Pictorial Structures for Object Recognition*. International Journal of Computer Vision, 2005. (Cité en page 61.)

[Felzenszwalb 2008] Pedro Felzenszwalb, David McAllester et Deva Ramanan. *A discriminatively trained, multiscale, deformable part model*. In Computer Vision and Pattern Recognition, 2008. (Cité en pages 37, 38 et 41.)

[Flir 2015] Flir. *Infrared camera - Flir Tau 2*. http://www.flir.com/cores/display/?id=54717, 2015. Accédé en : 2015-09. (Cité en page 98.)

[Freund 1996] Yoav Freund et Robert E. Schapire. *Experiments with a New Boosting Algorithm*. In Proceedings of the Thirteenth International Conference on Machine Learning (ICML 1996), 1996. (Cité en page 30.)

[Gaszczak 2011] Anna Gaszczak, Toby P Breckon et Jiwan Han. *Real-time People and Vehicle Detection from UAV Imagery*. In Intelligent Robots and Computer Vision, 2011. (Cité en page 59.)

[Gilmore 2009] E Thomas Gilmore, Preston D Frazier et MF Chouikha. *Improved Human Detection Using Image Fusion*. In IEEE Conference on Robotics and Automation, 2009. (Cité en page 119.)

[GMVRT1 2015] GMVRT1. *Base de données de d'entraînenemt réaliste multi-élévations GMVRT1.* http://home.mis.u-picardie.fr/~p-blondel/papers/data/gmvrt-v1.zip, 2015. Accédé en : 2015-09. (Cité en page 66.)

[GMVRT2 2015] GMVRT2. *Base de données de d'entraînenemt réaliste multi-élévations GMVRT2.* http://home.mis.u-picardie.fr/~p-blondel/papers/data/gmvrt-v2.zip, 2015. Accédé en : 2015-09. (Cité en page 86.)

[GMVST1 2015] GMVST1. *Base de données d'entraînment syntétique multi-élévations GMVST1*. http://home.mis.u-picardie.fr/~p-blondel/papers/data/gmvst, 2015. Accédé en : 2015-09. (Cité en page 62.)

[GMVST2 2015] GMVST2. *Base de données d'entraînement synthétique multi-élévations et multi-ratios GMVST2*. http://home.mis.u-picardie.fr/~p-blondel/papers/data/gmvst-v2, 2015. Accédé en : 2015-09. (Cité en page 62.)

[Gu 2009] Chunhui Gu, Jasmine J Lim, Pablo Arbeláez et Jagannath Malik. *Recognition using regions*. In Computer Vision and Pattern Recognition, 2009. (Cité en pages 55 et 56.)

[Hartley 2004] R. I. Hartley et A. Zisserman. Multiple view geometry in computer vision. Cambridge University Press, ISBN : 0521540518, second édition, 2004. (Cité en page 100.)

[Huang 2005] Chang Huang, Haizhou Ai, Yuan Li et Shihong Lao. *Vector Boosting for Rotation Invariant Multi-View Face Detection*. In Internation Conference on Computer Vision, 2005. (Cité en page 79.)

[INRIA 2014] INRIA. *Base de donnée proposée par INRIA*. http://pascal.inrialpes.fr/data/human/, 2014. Accédé en : 2015-09. (Cité en pages 76, 77 et 134.)

[Itti 1998] Laurent Itti, Christof Koch et Ernst Niebur. *A Model of Saliency-Based Visual Attention for Rapid Scene Analysis*. Transactions on pattern analysis and machine intelligence (TPAMI), 1998. (Cité en pages 48, 49 et 50.)

[Itti 2001] L Itti et C Koch. *Computational modelling of visual attention.* Nature reviews. Neuroscience, 2001. (Cité en page 49.)

[Jiang 2004] Lijun Jiang, Feng Tian, Lim Ee Shen, Shiqian Wu, Susu Yao, Zhongkang Lu et Lijun Xu. *Perceptual-based fusion of IR and visual images for human detection*. In Intelligent Multimedia, Video and Speech Processing, 2004. (Cité en page 117.)

[Katramados 2011] Ioannis Katramados et Toby Breckon. *Real-time visual saliency by division of gaussians*. In Internation Conference on Image Processing (ICIP), 2011. (Cité en pages 52, 55 et 70.)

[Kennedy 1995] J Kennedy et R Eberhart. *Particle swarm optimization*. In International Conference on Neural Networks (ICNN), 1995. (Cité en pages 43 et 146.)

[Kim 2005] Kyungnam Kim, Thanarat H. Chalidabhongse, David Harwood et Larry Davis. *Real-time Foreground-background Segmentation Using Codebook Model*. Real-Time Imaging, 2005. (Cité en page 47.)

[Kim 2006] Jun-Sik Kim et In So Kweon. *Estimating Intrinsic Parameters of Cameras using Two Arbitrary Rectangles*. In International Conference on Pattern Recongition, 2006. (Cité en page 107.)

[Kim 2008] Yong Sun Kim, Jae Hak Lee et Jong Beom Ra. *Multi-sensor Image Registration Based on Intensity and Edge Orientation Information*. Pattern Recognition, 2008. (Cité en pages 125, 126 et 127.)

[Koch 1985] C. Koch et S. Ullman. *Shifts in selective visual attention : towards the underlying neural circuitry.* Human neurobiology, 1985. (Cité en page 49.)

[Konigs 2012] A. Konigs et D. Schulz. *Evaluation of thermal imaging for people detection in outdoor scenarios*. In Safety, Security, and Rescue Robotics, 2012. (Cité en page 110.)

[Kovesi 2000] Peter Kovesi. *Phase congruency : A low-level image invariant*. Psychological Research, 2000. (Cité en pages 110, 111 et 112.)

[Krishnapuram 2004] Balaji Krishnapuram, David Williams, Ya Xue, Lawrence Carin, Mário Figueiredo et Alexander J Hartemink. *On semi-supervised classification*. In Advances in neural information processing systems, 2004. (Cité en page 23.)

[Krizhevsky 2012] Alex Krizhevsky, Ilya Sutskever et Geoffrey E. Hinton. *Imagenet classification with deep convolutional neural networks*. In Advances in Neural Information Processing Systems, 2012. (Cité en page 39.)

[Krotosky 2007] Stephen J. Krotosky et Mohan M. Trivedi. *Mutual information based registration of multimodal stereo videos for person tracking*. Computer Vision and Image Understanding, 2007. (Cité en page 106.)

[LeCun 1989] Y. LeCun, L. D. Jackel, B. Boser, J. S. Denker, H. P. Graf, I. Guyon, D. Henderson, R. E. Howard et W. Hubbard. *Handwritten Digit Recognition : Applications of Neural Net Chips and Automatic Learning*. IEEE Communication, 1989. (Cité en page 38.)

[Levin 2003] A. Levin, P. Viola et Y. Freund. *Unsupervised improvement of visual detectors using cotraining*. In International Conference on Computer Vision, 2003. (Cité en pages 121 et 122.)

[Li 2010] Xiaodong Li. *Niching without niching parameters : particle swarm optimization using a ring topology*. Evolutionary Computation, 2010. (Cité en page 44.)

[Liebelt 2008] Joerg Liebelt, Cordelia Schmid et Klaus Schertler. *Viewpoint-independent object class detection using 3d feature maps*. In Computer Vision and Pattern Recognition, 2008. (Cité en page 78.)

[Lienhart 2002] R. Lienhart et J. Maydt. *An extended set of Haar-like features for rapid object detection*. In Proceedings. International Conference on Image Processing, 2002. (Cité en page 15.)

[Lowe 1999] D.G. Lowe. *Object recognition from local scale-invariant features*. In International Conference on Computer Vision - Volume 2, 1999. (Cité en pages 12 et 14.)

[Lu 2013] Shijian Lu, Cheston Tan et Joo-Hwee Lim. *Robust and Efficient Saliency Modeling from Image Co-occurrence Histograms.* Transactions on pattern analysis and machine intelligence (TPAMI), 2013. (Cité en pages 53, 55 et 70.)

[Maes 1997] Frederik Maes, André Collignon, Dirk Vandermeulen, Guy Marchal et Paul Suetens. *Multimodality Image Registration by Maximization of Mutual Information*. Transactions on Medical Imaging, 1997. (Cité en page 126.)

[Mostaghim 2003] S. Mostaghim et J. Teich. *Strategies for finding good local guides in multi-objective particle swarm optimization (MOPSO)*. In Proceedings of Swarm Intelligence Symposium (SIS), 2003. (Cité en page 151.)

[Nelder 1965] John A Nelder et Roger Mead. *A simplex method for function minimization*. The computer journal, 1965. (Cité en page 127.)

[Olmeda 2012a] D. Olmeda, J.M. Armingol et A. de la Escalera. *Discrete features for rapid pedestrian detection in infrared images*. In Intelligent Robots and Systems, 2012. (Cité en page 114.)

[Olmeda 2012b] D. Olmeda, A. de la Escalera et J.M. Armingol. *Contrast invariant features for human detection in far infrared images*. In Intelligent Vehicles Symposium, 2012. (Cité en pages 110, 111 et 112.)

[Olmeda 2013] Daniel Olmeda. *Pedestrian detection in far infrared images*. PhD thesis, 2013. (Cité en page 112.)

[OOQP 2015] OOQP. *Librairie C++ OOQP pour l'optimisation de problèmes quadratiques.* `http://quadprog.sourceforge.net/`, 2015. Accédé en : 2015-09. (Cité en page 27.)

[Owechko 2004] Y. Owechko, S. Medasani et N. Srinivasa. *Classifier Swarms for Human Detection in Infrared Imagery*. In Conference on Computer Vision and Pattern Recognition Workshop (CVPRW), 2004. (Cité en pages 43, 44, 45 et 46.)

[Papageorgiou 1998] Constantine P Papageorgiou, Michael Oren et Tomaso Poggio. *A general framework for object detection*. In International Conference on Computer vision, 1998. (Cité en page 14.)

[Papageorgiou 2000] Constantine Papageorgiou et Tomaso Poggio. *A Trainable System for Object Detection*. International Journal of Computer Vision, 2000. (Cité en pages 9 et 22.)

[Portmann 2014] Jan Portmann, Simon Lynen, Maria Chli et Roland Siegwart. *People detection and tracking from aerial thermal views*. In Robotics and Automation (ICRA), 2014 IEEE International Conference on, 2014. (Cité en page 60.)

[Qi 2014] Bin Qi, V. John, Zheng Liu et S. Mita. *Use of Sparse Representation for Pedestrian Detection in Thermal Images*. In Computer Vision and Pattern Recognition Workshops, 2014. (Cité en page 113.)

[QuadProg++ 2015] QuadProg++. *Librairie C++ QuadProg++ pour l'optimisation de problèmes quadratiques*. `http://quadprog.sourceforge.net/`, 2015. Accédé en : 2015-09. (Cité en page 27.)

[Rapport Mikron Company] Rapport Mikron Company. *Table of Emissivity of Various*. Rapport technique. (Cité en page 96.)

[Reilly 2010] Vladimir Reilly, Berkan Solmaz et Mubarak Shah. *Geometric constraints for human detection in aerial imagery*. In European conference on Computer vision : Part VI, 2010. (Cité en page 59.)

[Rogalski 2002] Antoni Rogalski. *Infrared detectors : an overview*. Infrared Physics and Technology, 2002. (Cité en pages 97 et 98.)

[Roth 2010] Peter M. Roth, Christian Leistner, Armin Berger et Horst Bischof. *Multiple instance learning from multiple cameras*. In Computer Vision and Pattern Recognition Workshop, 2010. (Cité en pages 122 et 123.)

[RoulisTest 2015] RoulisTest. *Base de données de pour tester la robustesse au roulis*. `http://home.mis.u-picardie.fr/~p-blondel/papers/data/roulis_test.zip`, 2015. Accédé en : 2015-09. (Cité en page 87.)

[Rudakevych 2007] Pavlo Rudakevych et Brian Yamauchi. *A man portable hybrid UAV/UGV system*. In Defense and Security Symposium, 2007. (Cité en page 1.)

[Rudol 2008] Piotr Rudol et Patrick Doherty. *Human Body Detection and Geolocalization for UAV Search and Rescue Missions Using Color and Thermal Imagery*. In Aerospace Conference, 2008. (Cité en page 58.)

[Saif 2014] O. Saif, I. Fantoni et A. Zavala-Rio. *Flocking of Multiple Unmanned Aerial Vehicles by LQR Control*. In International Conference on Unmanned Aircraft Systems, 2014. (Cité en page 1.)

[Saisan 2005] Payam Saisan, Los Angeles, Swarup Medasani et Yuri Owechko. *Multi-View Classifier Swarms for Pedestrian Detection and Tracking*. In Conference on Computer Vision and Pattern Recognition Workshop (CVPRW), 2005. (Cité en pages 43, 44 et 46.)

[San-Biagio 2012] M. San-Biagio, M. Crocco et M. Cristani. *Recursive segmentation based on higher order statistics in thermal imaging pedestrian detection*. In Communications Control and Signal Processing, 2012. (Cité en page 141.)

[Santos 2013] Omar Santos, Hugo Romero, Sergio Salazar et Rogelio Lozano. *Real-time Stabilization of a Quadrotor UAV : Nonlinear Optimal and Suboptimal Control*. Journal of Intelligent and Robotic Systems, 2013. (Cité en page 1.)

[Sermanet 2012] Pierre Sermanet, Koray Kavukcuoglu, Soumith Chintala et Yann LeCun. *Pedestrian Detection with Unsupervised Multi-Stage Feature Learning*. In Computer Vision and Pattern Recognition, 2012. (Cité en page 39.)

[St-Laurent 2012] Louis St-Laurent. *Combinaison de caméras thermique et couleur pour la segmentation cibles/arrière-plan en environnement non contrôlé*. PhD thesis, Université Laval, 2012. (Cité en page 100.)

[Stauffer 1999] Chris Stauffer et W. E. L. Grimson. *Adaptive background mixture models for real-time tracking*. In Computer Vision and Pattern Recognition, 1999. (Cité en page 47.)

[Swets 1995] Daniel L Swets, Bill Punch et John Weng. *Genetic algorithms for object recognition in a complex scene*. In Image Processing, 1995. Proceedings., International Conference on, 1995. (Cité en page 43.)

[SyntheticAerialTest1 2015] SyntheticAerialTest1. *Base de données de test syntétique en vue aérienne SyntheticAerialTest1*. `http://home.mis.u-picardie.fr/~p-blondel/papers/data/scene_test.zip`, 2015. Accédé en : 2015-09. (Cité en page 63.)

[Thomas 2006] Alexander Thomas, Vittorio Ferrar, Bastian Leibe, Tinne Tuytelaars, Bernt Schiel et Luc Van Gool. *Towards multi-view object class detection*. In Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on, 2006. (Cité en page 79.)

[Toth 2003] D. Toth et T. Aach. *Detection and recognition of moving objects using statistical motion detection and Fourier descriptors*. In International Conference on Image Analysis and Processing, 2003. (Cité en pages 8 et 47.)

[Treisman 1980] Anne M Treisman et Garry Gelade. *A feature-integration theory of attention*. Cognitive psychology, 1980. (Cité en page 48.)

[Villamizar 2010] Michael Villamizar, Francesc Moreno-Noguer, Juan Andrade-Cetto et Alberto Sanfeliu. *Efficient rotation invariant object detection using boosted random ferns*. In Computer Vision and Pattern Recognition, 2010. (Cité en page 79.)

[Viola 2001] Paul Viola et Michael Jones. *Rapid Object Detection using a Boosted Cascade of Simple Features*. In Computer Vision and Pattern Recognition, 2001. (Cité en pages 9, 14, 22 et 32.)

[Wren 1997] Christopher Wren, Ali Azarbayejani, Trevor Darrell et Alex Pentland. *Pfinder : Real-Time Tracking of the Human Body*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997. (Cité en page 47.)

[Wu 2007] Bo Wu et Ram Nevatia. *Cluster Boosted Tree Classifier for Multi-View, Multi-Pose Object Detection*. In Computer Vision and Pattern Recognition, 2007. (Cité en page 79.)

[Xu 2014] Philippe Xu, Franck Davoine et Thierry Denoeux. *Evidential combination of pedestrian detectors*. In British Machine Vision Conference, 2014. (Cité en pages 116 et 117.)

[Yan 2007] Pingkun Yan, Saad M Khan et Mubarak Shah. *3d model based object class detection in an arbitrary view*. In International Conference on Computer Vision, 2007. (Cité en page 78.)

[Yang 2010] Shengxiang Yang et Changhe Li. *A clustering particle swarm optimizer for locating and tracking multiple optima in dynamic environments*. Evolutionary Computation, 2010. (Cité en page 44.)

[Zhang 2007a] Cha Zhang et Paul Viola. *Multiple-Instance Pruning For Learning Efficient Cascade Detectors*. In Neural Information Processing Systems, 2007. (Cité en page 33.)

[Zhang 2007b] Li Zhang, Bo Wu et Ram Nevatia. *Pedestrian detection in infrared images based on local shape features*. In Computer Vision and Pattern Recognition, 2007. (Cité en page 109.)

[Zhong 2005] Shi Zhong, Wei Tang et Taghi M Khoshgoftaar. *Boosted noise filters for identifying mislabeled data*. Rapport technique, 2005. (Cité en page 132.)

[Zhou 2005] Jianpeng Zhou et Jack Hoang. *Real Time Robust Human Detection and Tracking System*. In Computer Vision and Pattern Recognition, 2005. (Cité en pages 8 et 47.)

[Zitnick 2013] C Lawrence Zitnick et Piotre Dollár. *Edge Boxes : Locating Object Proposals from Edges*. In European Conference on Computer Vision (ECCV), 2013. (Cité en pages 55, 56, 128 et 129.)