



HAL
open science

Méthodes séquentielles de Monte Carlo pour le suivi d'objets multiples hétérogènes en données brutes de télémétrie laser

Élodie Vanpoeringhe

► **To cite this version:**

Élodie Vanpoeringhe. Méthodes séquentielles de Monte Carlo pour le suivi d'objets multiples hétérogènes en données brutes de télémétrie laser. Autre [cs.OH]. Université du Littoral Côte d'Opale, 2014. Français. NNT : 2014DUNK0346 . tel-00983352

HAL Id: tel-00983352

<https://theses.hal.science/tel-00983352>

Submitted on 25 Apr 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Thèse de doctorat

De l'Université du Littoral Côte d'Opale

Présentée par

Élodie Vanpoperinghe

Pour obtenir le grade de

Docteur de l'Université du Littoral Côte d'Opale

Domaine

Automatique, génie informatique, traitement du signal et de l'image

Méthodes séquentielles de Monte Carlo pour le suivi d'objets multiples hétérogènes en données brutes de télémétrie laser

Thèse soutenue le 27 janvier 2014 devant le jury composé de

Michèle ROMBAUT	Professeur des Universités	Université Joseph Fourier	Président
Régis LHERBIER	Maître de Conférences	Université du Littoral Côte d'Opale	Examinateur
Véronique CHERFAOUI	Maître de Conférences - HDR	Université de Technologie de Compiègne	Rapporteur
Yassine RUICHEK	Professeur des Universités	Université de Technologie de Belfort Montbéliard	Rapporteur
Jean-Charles NOYER	Professeur des Universités	Université du Littoral Côte d'Opale	Co-directeur
Martine WAHL	Chargé de Recherche	Institut Français des Sciences et Technologies des Transports, de l'Aménagement et des Réseaux	Co-directeur

À tous ceux que j'aime

Remerciement

Ce travail de thèse a été réalisé au laboratoire électronique, ondes et signaux pour les transports (LEOST du département COSYS de l'Ifsttar) dirigé successivement par Mme Marion Berbineau et M. Charles Tatkeu que je tiens à remercier. C'est avec bonne humeur et sympathie qu'ils m'ont accueillie au sein de leur équipe durant ma thèse.

Je tiens à remercier l'Ifsttar et la région Nord-Pas-De-Calais pour avoir financé ce travail.

Je tiens tout particulièrement à remercier M. Jean-Charles Noyer, mon directeur de thèse sans qui celle-ci n'aurait pu aboutir. Je suis reconnaissante du temps qu'il m'a accordé, ses qualités pédagogiques et scientifiques, son écoute, ses conseils, sa franchise et sympathie. J'ai beaucoup appris grâce à lui et je lui adresse ma gratitude pour cela.

J'adresse de chaleureux remerciements à mon encadrant de thèse Mme Martine Wahl, pour son attention de tout instant sur mes travaux, pour ses conseils et son écoute qui ont mené à la réussite de cette thèse.

Je souhaite remercier Dominique Gruyer pour m'avoir permis d'utiliser les données SiVIC qui ont été utilisées dans cette thèse. Je souhaite également remercier Sébastien Lefebvre pour m'avoir aidée dans l'utilisation de ce logiciel.

Je souhaite remercier l'ensemble des membres du jury Véronique CHERFAOUI, Yassine RUICHEK, Régis LHERBIER, Michèle ROMBAUT qui m'ont fait l'honneur de lire ma thèse et participer à ma soutenance. Je voudrais leur témoigner ma gratitude et les remercier de l'intérêt qu'ils ont porté à mon travail.

Comment ne pas remercier les membres du site de Villeneuve d'Ascq qui m'ont accueillie, conseillée et ont rendu agréable ces années : Isabelle, Corinne, Lidwine pour leurs attentions, leurs accompagnements administratifs et leurs échanges amicaux, Émilie, Mohamed qui m'ont supportée comme voisine de bureau. Manu qui a tout essayé pour avoir une JSS sans succès. Et enfin tous ceux que je ne cite pas mais avec qui j'ai eu l'honneur de discuter et d'échanger des idées.

Un merci tout particulier à ceux qui ont parsemé ces trois années de souvenirs inoubliables : Sonia, Aurélie, Stephen, Éloïdy, Paola, Camille, Wilfried, Cyril, Philippe, Greg, Manu et tant d'autres qui se reconnaîtront.

Sur un plan plus personnel, je tiens à remercier mes parents et grand-parents pour leur soutien sans faille depuis tant d'année. Merci à eux de toujours être là pour moi dans les bons comme les mauvais moments.

Enfin je terminerais par mes amis sans qui rien ne serait possible, sans qui ma vie ne serait pas la même. Pour tous les moments de détente, les week-ends, les rires et les larmes que nous avons partagées. Pour leur présence et leur soutien dans ce voyage qui se termine. Fanny, Anne-Laure, Daisy, Béatrice, Jimmy, Éloïdy, Aurélie et Florent, les mots ne suffisent pas pour exprimer ma gratitude et mon amour pour vous.

Table des matières

Liste des figures	xiii
Liste des tableaux	xv
Glossaire	xvii
Introduction	1
I Problématique, capteurs et architectures de traitement	3
I.1 Problématique	3
I.2 Capteurs et architectures	7
I.2.1 Capteurs	7
I.2.1.1 Système de vision par caméra	7
I.2.1.2 Radar	9
I.2.1.3 Lidar	10
I.2.2 Architectures de traitement de données	11
I.2.2.1 Architectures multicapteur classiques	11
I.2.2.2 Exemple d'une architecture autonome dans le domaine automobile	13
I.2.2.3 Architecture monocapteur pour lidar	13
I.3 Le télémètre laser à balayage : capteur lidar	15
I.3.1 Principe du télémètre laser à balayage	15
I.3.2 Exemple de mesures lidar issue de capteur monoplan et multiplan	17
I.3.3 Évolution des capteurs lidar	19
I.3.3.1 Capteur IBEO LD <i>Automotive</i>	20
I.3.3.2 Capteur IBEO LD-ML (<i>Ladar Digital MultiLayer</i>)	20
I.3.3.3 Capteur ALASCA (<i>Automotive Laser SCanner</i>)	21
I.4 Conclusion	22
II Problème de détection et suivi d'objets multiples	23
II.1 Architecture fonctionnelle classique du problème de détection et de suivi d'obstacles	23
II.2 Détection dans les données télémétriques	24
II.2.1 Partitionnement de données (<i>clustering</i>)	25
II.2.1.1 Algorithme de partitionnement	25
II.2.1.2 Seuil adaptatif de Dietmayer <i>et al.</i> (2001)	26
II.2.1.3 Seuil adaptatif de Santos <i>et al.</i> (2003)	26
II.2.1.4 Seuil adaptatif de Borges et Aldon (2004)	27
II.2.2 Segmentation de partitions	27
II.2.2.1 Approche incrémentale	28
II.2.2.2 Approche <i>Split and Merge</i>	28
II.2.2.3 Approche RANSAC (<i>Random Sample Consensus</i>)	30

II.2.2.4	Transformée de Hough	31
II.2.2.5	Approche de maximisation de l'espérance (EM, <i>Expectation-Maximization</i>)	33
II.2.3	Identification et classification d'objets	34
II.3	Modélisation des objets	34
II.4	Association de données	35
II.5	Méthode de suivi : filtrage	36
II.6	Filtre de Kalman et ses filtres dérivés	38
II.6.1	Filtre de Kalman	38
II.6.2	Filtre de Kalman étendu (EKF)	39
II.6.3	Filtre de Kalman sans parfum (UKF)	39
II.7	Filtrage particulaire	41
II.7.1	Principe	41
II.7.2	Échantillonnage d'importance	42
II.7.3	Problème de la dégénérescence des particules	43
II.7.3.1	Résolution de la dégénérescence par choix d'une densité d'importance optimale pour une application donnée	43
II.7.3.2	Résolution de la dégénérescence par rééchantillonnage des poids	44
II.8	Présentation de différents filtres particulières	46
II.8.1	Filtre à échantillonnage séquentiel par importance (SIS)	46
II.8.2	Filtre Bootstrap ou SIR	47
II.8.3	Filtre à rééchantillonnage adaptatif (SIR adaptatif)	48
II.8.4	Filtre particulaire auxiliaire (APF)	48
II.8.5	Filtre de Rao-Blackwell	50
II.9	Exemples d'application au suivi de cibles	51
II.10	Conclusion	54
III	Approche conjointe de détection et suivi	55
III.1	Introduction	55
III.2	Présentation du problème	56
III.3	Architecture fonctionnelle proposée	58
III.4	Module de surveillance	59
III.4.1	Détecteur d'objets	59
III.4.1.1	Restriction des mesures d'un balayage à une zone d'étude	60
III.4.1.2	Algorithme de partitionnement des mesures d'un balayage utilisant le seuil de Santos	61
III.4.1.3	Algorithme de segmentation basé sur la théorie de <i>Split and Merge</i>	62
III.4.1.4	Calcul du centre de gravité de la partition	64
III.4.1.5	Résultats de détection	67
III.4.2	Gestionnaire de pistes	68
III.4.3	Conclusion	70
III.5	Module de détection, association et suivi DAT	70
III.6	Approche conjointe de détection et suivi (ITPF)	70
III.6.1	Modélisation d'état du problème	70
III.6.1.1	Équation dynamique	71
III.6.1.2	Équation de mesure	71
III.6.2	Approche proposée	72
III.6.2.1	Initialisation	72
III.6.2.2	Évolution des particules	73
III.6.2.3	Pondération des particules : exploitation du modèle d'état	74
III.6.2.3.1	Calcul du poids sur la zone complète	75

III.6.2.3.2	Calcul du poids sur une restriction de l'espace d'observation . . .	75
III.6.2.4	Redistribution des particules	79
III.6.2.5	Prédiction de la partie linéaire (Kalman) du filtre de Rao-Blackwell	79
III.6.2.6	Estimation d'état	79
III.7	Résultats	80
III.7.1	Cas synthétique de suivi multivéhicule	80
III.7.1.1	Scénario et caractéristiques du capteur	80
III.7.1.2	Comparaison de l'approche proposée ITPF à une approche classique (CDG) sur les données du simulateur de la thèse	82
III.7.2	Résultats obtenus avec les données du logiciel SiVIC	92
III.7.2.1	Scénario et caractéristiques du capteur	92
III.7.2.2	Comparaison de l'approche proposée ITPF à une approche classique (CDG) sur des données du simulateur SiVIC	94
III.8	Conclusion	107
IV	Approche conjointe de détection et de suivi d'objets multiples hétérogènes par laser multiplans	109
IV.1	Introduction	109
IV.2	Détection et suivi d'objets multiples hétérogènes (approche VS-ITPF)	110
IV.2.1	Problématique du cas multi-objets	110
IV.2.2	Adaptation du module de surveillance au cas multi-objets hétérogènes	111
IV.2.3	Résumé des algorithmes du module de surveillance dans le cas multi-objets hétéro- gènes	115
IV.2.4	Adaptation du module DAT au cas multi-objets hétérogènes	115
IV.2.5	Résultats expérimentaux	116
IV.2.5.1	Scénario et caractéristiques du capteur pour le simulateur de la thèse	116
IV.2.5.2	Résultat de l'approche multi-objets pour les données du simulateur de la thèse	118
IV.2.5.3	Scénario et caractéristiques du capteur pour le simulateur SiVIC	125
IV.2.5.4	Résultat de l'approche VS-ITPF pour les données du simulateur SiVIC	126
IV.2.6	Conclusion	129
IV.3	Détection et suivi d'obstacles multiples homogènes avec un lidar multiplan (approche ML- ITPF)	130
IV.3.1	Problématique et mesures d'un capteur multiplan	130
IV.3.2	Notation : extension au cas multiplan	132
IV.3.3	Adaptation du détecteur d'objet du module de surveillance au cas multiplan	132
IV.3.4	L'approche ML-ITPF	134
IV.3.4.1	Modélisation du problème	134
IV.3.4.2	Filtrage de Rao-Blackwell	135
IV.3.4.3	Évolution des particules	135
IV.3.5	Résultats expérimentaux	137
IV.3.5.1	Scénario et caractéristiques du capteur	137
IV.3.5.2	Résultats de l'approche ML-ITPF	138
IV.3.6	Conclusion	148
	Conclusion	149
A	Simulateur de données	153
A.1	Le capteur lidar à balayage	154
A.1.1	Principe du générateur de données télémétriques	154
A.1.2	Modélisation d'un scénario routier	154

A.1.3	Modélisation de la trajectoire des véhicules cibles	156
A.1.4	Construction d'une image pour un balayage donné	156
A.1.4.1	Représentation d'un véhicule dans le repère cartésien du capteur	156
A.1.4.2	Représentation d'un véhicule dans le repère polaire du capteur	158
A.1.4.3	Calcul d'une image d'un véhicule dans le repère polaire du capteur	158
A.1.4.4	Calcul de l'image capteur d'une scène multivéhicule pour un balayage donné	158
A.1.5	Exemple de données obtenues à partir du simulateur	158
A.2	Le simulateur SiVIC	160
A.2.1	Présentation	160
A.2.2	Fonctionnalités du simulateur SiVIC	161
A.2.3	Exemple de données obtenues avec le simulateur SiVIC pour un lidar multiplan	161
Références bibliographiques		174

Table des figures

I.1	Nombre de tués sur les routes de 1950 à 2011 (Sécurité routièrre, 2012)	3
I.2	Évolution du nombre de tués sur les routes en fonction des nouvelles législations (Sécurité routièrre, 2012)	4
I.3	Architecture minimale de détection	5
I.4	Une camera avec objectif de type <i>Fisheye</i> à gauche et photo du ciel prise avec la caméra à droite	7
I.5	Système stéréoscopique expérimental (LEOST) à gauche et image d’une scène prise par un système stéréoscopique et sa carte de disparité à droite	8
I.6	Image d’un radar SRR extraite de (Bank, 2007)	9
I.7	Principe du lidar multicouche	10
I.8	Architecture de fusion centralisée extraite de (Waltz et Llinas, 1990)	11
I.9	Architecture de fusion autonome extraite de (Waltz et Llinas, 1990)	12
I.10	Architecture de fusion hybride extraite de (Waltz et Llinas, 1990)	12
I.11	Architecture du système de fusion de données du projet CARSENSE extraite de Wahl (2002)	13
I.12	Architecture de traitement monocapteur lidar extraite de (Kiehn <i>et al.</i> , 2002)	14
I.13	Architecture de traitement monocapteur lidar extraite de Jida (2008)	15
I.14	Principe du balayage horizontal (en haut) et vertical (en bas) extrait de (Besesty, 1999)	15
I.15	Schéma de fonctionnement du télémètre laser à balayage extrait de (Besesty, 1999)	16
I.16	Principe de mesure du temps de vol d’une onde émise par le télémètre laser à balayage extrait de (Besesty, 1999)	16
I.17	Image CCD de la scène et le balayage capteur associé vu par le lidar monoplan (en bas à gauche) et par le lidar multiplan (en bas à droite)	18
I.18	Agrandissement de la représentation de la scène vue par le lidar multiplan (figure I.18 à gauche) sur le véhicule en bas de la voie droite (à gauche) et sur le véhicule voie centrale (à droite)	18
I.19	Présentation des capteurs DENSO, SICK LD-MRS et Velodyne HDL-64E (de gauche à droite) extrait de (com, 2014)	19
I.20	Le capteur IBEO LD-ML (à gauche) et ses caractéristiques (à droite)	20
I.21	Le capteur IBEO ALASCA XT (à gauche) et ses caractéristiques (à droite)	21
II.1	Étapes classiques d’un algorithme de détection et de suivi	23
II.2	Seuil S_{NA} fixe d’un algorithme non adaptatif	25
II.3	Paramètres du seuil adaptatif S_{Santos} selon Santos <i>et al.</i> (2003)	26
II.4	Paramètres du seuil adaptatif S_{Borges} selon Borges et Aldon (2004)	27
II.5	Exemple de dispersion des <i>inliers</i> et <i>outliers</i>	30
II.6	Paramètres (Θ, ρ) tels que Θ est l’angle polaire de la normale à la droite (d) (Duda et Hart, 1971)	32
II.7	Biais de l’estimateur de Kalman étendu	37
II.8	Évolution des particules (figure extraite de Diginext (1996))	42

II.9 Rééchantillonnage des particules	45
II.10 Sélection des particules selon le vecteur U	46
III.1 Illustration d'un scénario de détection et de suivi	56
III.2 Représentation d'un véhicule dans le repère du capteur	57
III.3 Représentation des impacts lidar pour différentes orientations de véhicules dans le référentiel du capteur	57
III.4 Architecture fonctionnelle de l'approche ITPF proposée	58
III.5 Architecture fonctionnelle simplifiée du module de surveillance	59
III.6 Entités étudiées du détecteur d'objet	60
III.7 Zone d'étude d'une autoroute à trois voies : la zone est construite de façon à rejeter les points de bordures de routes	61
III.8 Point de rupture d'un ensemble de mesures dont on cherche les droites d'interpolation	63
III.9 Illustration des cinq configurations possibles d'une partition selon la position relative de l'objet au capteur	64
III.10 Estimation du centre de gravité des partitions détectées (données SiVIC)	67
III.11 Gestion des pistes d'une famille d'objet	68
III.12 Véhicule particulière reconstruit	74
III.13 Une région d'intérêt (ROI) de la zone d'étude considérée	76
III.14 Paramètres (θ_{min}^{ROI} , θ_{max}^{ROI} , ρ_{min}^{ROI} , ρ_{max}^{ROI}) d'une zone de suivi ou région d'intérêt (ROI)	77
III.15 Région ROI^+ d'une région d'intérêt ROI	77
III.16 Cas limite de la région ROI^+	78
III.17 Illustration des trajectoires relatives au véhicule porteur du capteur pour le scénario établi aux tableaux III.1 et III.2	82
III.18 Comparaison du nombre de pistes créées avec le nombre de véhicules pour les méthode ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse	83
III.19 Perception du lidar au temps $t = 5s$ (mesures en rouge) superposée aux modèles estimés des véhicules (rectangle en bleu) : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse	84
III.20 Agrandissement des véhicules de la scène figure III.19, au temps $t = 5s$, véhicule de la voie centrale en haut et de la voie de gauche en bas : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse	84
III.21 Perception du lidar au temps $t = 10s$ (mesures en rouge) superposée aux modèles estimés des véhicules (rectangle en bleu) : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse	85
III.22 Agrandissement du véhicule de la voie de droite, au temps $t = 10s$, de la figure III.21 : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse	85
III.23 Agrandissement du véhicule de la voie de gauche, au temps $t = 10s$, de la figure III.21 : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse	86
III.24 Agrandissement du véhicule de la voie centrale, au temps $t = 10s$, de la figure III.21 : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse	86
III.25 Courbes des Vitesses et accélérations de référence et estimées, du véhicule de la voie de droite en fonction du temps : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse	88
III.26 Courbes des Vitesses et accélérations de référence et estimées, du véhicule de la voie de gauche en fonction du temps : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse	89
III.27 Courbes des Vitesses et accélérations de référence et estimées, du véhicule de la voie centrale en fonction du temps : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse	90

III.28	Erreur sur la position du véhicule de la voie de droite en fonction du temps : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse	91
III.29	Erreur sur la position du véhicule de la voie de gauche en fonction du temps : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse	91
III.30	Erreur sur la position du véhicule cible de la voie centrale en fonction du temps : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse	92
III.31	Illustration des trajectoires des véhicules relativement au véhicule porteur du capteur pour le scénario établi au tableau III.4	93
III.32	Quatre images de la scène au temps $t = 0s, t = 2,5s, t = 5s$ et $t = 10s$	94
III.33	Comparaison du nombre de pistes créées avec le nombre de véhicules détectés pour les méthodes ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC . . .	95
III.34	Nombre de pistes créées avec le nombre de véhicules détectés pour la méthode ITPF appliquées aux données du simulateur SiVIC	96
III.35	Illustration de la perte d'un véhicule par masquage, image de la scène à $t = 6,3s, t = 6,5s$ et $t = 6,7s$	97
III.36	Image CCD de la scène, au temps $t = 5s$ et la même scène perçue par le laser à balayage (en rouge) superposée aux modèles estimés des véhicules (rectangle en bleu) : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC	99
III.37	Agrandissement du véhicule de la voie de droite haut, au temps $t = 5s$, de la figure III.36 : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC . . .	99
III.38	Agrandissement du véhicule de la voie centrale, au temps $t = 5s$, de la figure III.36 : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC	100
III.39	Agrandissement du véhicule de la voie de droite bas, au temps $t = 5s$, de la figure III.36 : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC . . .	100
III.40	Agrandissement du véhicule de la voie de gauche, au temps $t = 5s$, de la figure III.36 : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC . . .	101
III.41	Image CCD de la scène, au temps $t = 10s$ et la même scène perçue par le laser à balayage (en rouge) superposée aux modèles estimés des véhicules (rectangle en bleu) : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC	101
III.42	Agrandissement des véhicules de la scène figure III.41, au temps $t = 10s$, véhicule de la voie de gauche en haut, de la voie de droite au centre et de la voie centrale en bas : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC	102
III.43	Courbes de vitesse selon x en haut et y estimées du véhicule de la voie de droite en fonction du temps : approches ITPF à gauche et CDG à droite appliquées aux données SiVIC	104
III.44	Courbes d'accélération selon x en haut et y estimées du véhicule de la voie de droite en fonction du temps : approches ITPF à gauche et CDG à droite appliquées aux données SiVIC	105
III.45	Courbes de vitesse selon x en haut et y estimées du véhicule de la voie centrale en fonction du temps : approches ITPF à gauche et CDG à droite appliquées aux données SiVIC	106
III.46	Courbes d'accélération selon x en haut et y estimées du véhicule de la voie centrale en fonction du temps : approches ITPF à gauche et CDG à droite appliquées aux données SiVIC	107
IV.1	Calcul du nouveau centre de gravité après modification du modèle véhicule	115
IV.2	Illustration des trajectoires relatives au véhicule porteur pour le scénario (défini par les tableaux IV.1 et IV.2), constitué d'un camion et de deux voitures	118
IV.3	Comparaison du nombre de pistes actives avec le nombre de véhicules estimé par le détecteur au cours de la simulation	118
IV.4	Perception du lidar aux temps $t = 5s$ (figure haut gauche), $t = 7,5s$ (figure haut droite), $t = 11s$ (figure bas gauche) et $t = 14s$ (figure bas droite) superposée aux modèles de véhicules estimés par le filtre ITPF, pour un cas multi-objets	119

IV.5	Courbes des vitesses à gauche et accélération à droite en fonction du temps, du camion situé à droite (figure du haut), du véhicule central (figure du centre) et du véhicule de gauche (figure du bas), pour un cas multi-objets	121
IV.6	Longueur et largeur estimées des boîtes englobantes des véhicules de la simulation en fonction du temps, en haut le camion et en bas les deux voitures	122
IV.7	Courbes d'erreur sur la position en fonction du temps, du camion situé voie de droite (figure haut gauche), pour la voiture voie centrale (figure haut droite) et pour la voiture voie de gauche (figure du bas)	124
IV.8	Illustration des trajectoires relatives au véhicule porteur pour le scénario du simulateur SiVIC	125
IV.9	Quatre images de la scène au temps $t = 1s$, $t = 4s$, $t = 7,5s$ et $t = 10s$	126
IV.10	Scènes aux instants $t = 1s$ (figure haut gauche), $t = 4s$ (figure haut droite), $t = 7,5s$ (figure bas gauche) et $t = 10s$ (figure bas droite) et véhicules estimés par le filtre VS-ITPF	127
IV.11	Longueur et largeur estimées des véhicules en fonction du temps à gauche l'Espace et à droite la Mini	128
IV.12	Représentation de l'Espace avant à $t = 1s$ (à gauche) et après à $t = 4s$ (à droite) mise à jour des dimensions	128
IV.13	Architecture fonctionnelle de l'approche VS-ITPF modifiée	129
IV.14	Exemple de deux véhicules vus par un capteur lidar multiplan au temps $t = 5s$ (en bleu tir à 0° d'élévation, en rouge tir à $0,4^\circ$ d'élévation, en vert tir à $0,8^\circ$ d'élévation et en cyan tir à $1,2^\circ$ d'élévation)	131
IV.15	Nombre de véhicules détectés en fonction du temps pour les plans 1 (à gauche) et 3 (à droite)	132
IV.16	Illustration d'un véhicule vu par quatre plans capteur et la détection qui en découle	134
IV.17	Double restriction, ROI^+ d'une région d'intérêt ROI de la zone d'étude considérée	136
IV.18	Trajectoires relatives au véhicule porteur de l'ensemble des véhicules de la scène	138
IV.19	Quatre images de la scène au temps $t = 0s$, $t = 2,5s$, $t = 5s$ et $t = 10s$	138
IV.20	Comparaison du nombre de pistes créées avec le nombre de véhicule estimé par le détecteur présent au cours de la simulation, pour les méthodes ML-ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC	140
IV.21	Image CCD de la scène, au temps $t = 5s$ et la même scène perçue par le laser à balayage (en rouge) superposée aux modèles estimés des véhicules (rectangle en bleu) : approches ML-ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC, pour un cas multiplan	141
IV.22	Agrandissement du véhicule de la voie de droite haut, au temps $t = 5s$, de la figure IV.21 : approches ML-ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC, pour un cas multiplan	141
IV.23	Agrandissement du véhicule de la voie centrale, au temps $t = 5s$, de la figure IV.21 : approches ML-ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC, pour un cas multiplan	142
IV.24	Agrandissement du véhicule de la voie de gauche, au temps $t = 5s$, de la figure IV.21 : approches ML-ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC, pour un cas multiplan	142
IV.25	Image CCD de la scène, au temps $t = 10s$ et la même scène perçue par le laser à balayage (en rouge) superposée aux modèles estimés des véhicules (rectangle en bleu) : approches ML-ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC, pour un cas multiplan	143
IV.26	Agrandissement du véhicule de la voie de droite, au temps $t = 10s$, de la figure IV.25 : approches ML-ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC, pour un cas multiplan	143

IV.27 Agrandissement du véhicule de la voie centrale, au temps $t = 10s$, de la figure IV.25 : approches ML-ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC, pour un cas multiplan	144
IV.28 Agrandissement du véhicule de la voie de gauche, au temps $t = 10s$, de la figure IV.25 : approches ML-ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC, pour un cas multiplan	144
IV.29 Courbes de vitesse selon x en haut et y en bas estimées du véhicule de la voie de droite en fonction du temps : approches ML-ITPF à gauche et CDG à droite appliquées aux données SiVIC, pour un cas multiplan	145
IV.30 Courbes d'accélération selon x en haut et y en bas estimées du véhicule de la voie de droite en fonction du temps : approches ML-ITPF à gauche et CDG à droite appliquées aux données SiVIC, pour un cas multiplan	146
IV.31 Courbes de vitesse selon x en haut et y en bas estimées du véhicule de la voie de gauche en fonction du temps : approches ML-ITPF à gauche et CDG à droite appliquées aux données SiVIC, pour un cas multiplan	147
IV.32 Courbes d'accélération selon x en haut et y en bas estimées du véhicule de la voie de gauche en fonction du temps : approches ML-ITPF à gauche et CDG à droite appliquées aux données SiVIC, pour un cas multiplan	148
A.1 Modélisation de la route, du capteur et des véhicules cibles	155
A.2 Modèle de la voiture dans les repères cartésien et polaire du capteur	157
A.3 Données générées pour le scénario considéré (deux véhicules V1, V2 dont l'un change de voie)	160
A.4 Modélisation de la rotation du capteur lidar pour générer des données multiplans	162
A.5 Exemple de données générées par le simulateur SiVIC	163

Liste des tableaux

I.1	Comparaison des capteurs DENSO, SICK LD-MRS et Velodyne HDL-64E extrait de (com, 2014)	19
III.1	Paramètres de simulation du capteur positionné relativement à la route (données du simulateur de la thèse)	81
III.2	Paramètres de simulation initiaux des véhicules cibles relativement à la route (données du simulateur de la thèse)	81
III.3	Écart-type de l'erreur d'estimation des positions, vitesses et accélérations pour les véhicules cibles sur une période de convergence de les méthodes ITPF et CDG appliquées aux données du simulateur de la thèse	92
III.4	Paramètres de simulation du capteur positionné relativement à la route (données SiVIC) . . .	93
IV.1	Paramètres de simulation	117
IV.2	Paramètres de simulation initiaux des véhicules cibles relativement à la route et au véhicule porteur	117
IV.3	Écart-type de l'erreur d'estimation des positions, vitesses et accélérations pour les véhicules cibles sur une période de convergence de la méthode VS-ITPF appliquées aux données du simulateur de la thèse, dans un cas multi-objets	124
IV.4	Paramètres de simulation	125
IV.5	Paramètres du capteur positionné relativement à la route	137
A.1	Données paramétrables pour un scénario donné	155
A.2	Données paramétrables pour chaque sous-ensemble du scénario	156
A.3	Paramètres de simulation	158
A.4	Paramètres du véhicule V1	159
A.5	Paramètres du véhicule V2	159
A.6	Paramètres de simulation du véhicule capteur, du capteur et de la route pour un scénario donné associé au simulateur SiVIC	162

Glossaire

(i) une particule.

N le nombre de particules.

X_t le vecteur d'état de position, vitesse et accélération à l'instant t , $X_t = [x, y, v_x, v_y, a_x, a_y]$.

$X_t^{(i)}$ le vecteur d'état estimé à l'instant t pour la particule (i) .

Z_t le vecteur des mesures du balayage capteur de l'instant t , $Z_t = \{z_\theta; \theta \in \{\theta_{min}, \theta_{min} + \Delta\theta, \dots, \theta_{max}\}\}$.

t l'indice de l'échantillon de mesure.

$w_t^{(i)}$ le poids de la particule (i) à l'instant t .

(ρ, θ) les mesures en sortie du capteur.

$(\theta_{min}, \theta_{max})$ l'ouverture angulaire du capteur.

$Z_t^{(i)}$ le vecteur à l'instant t des échos reconstitués d'un véhicule à partir de la particule (i) .

$\Delta\theta$ la résolution angulaire du capteur.

Δt la période d'échantillonnage du capteur.

\mathcal{P}_t l'ensemble des partitions de Z_t , $\mathcal{P}_t = \left\{ \mathcal{P}(n); n \in [0, N_{max}], N_{max} \leq \frac{\theta_{max} - \theta_{min}}{\Delta\theta} + 1, \mathcal{P}(0) = \emptyset \right\}$.

ρ la distance de l'impact au capteur.

θ la résolution angulaire du capteur (pas de tir du capteur).

z_θ^+ la mesure de l'écho par le capteur à l'angle de tir suivant, $\theta^+ = \theta + \Delta\theta$.

z_θ^- la mesure de l'écho à l'angle de tir précédent, $\theta^- = \theta - \Delta\theta$.

z_θ la mesure d'un écho exprimée en coordonnées polaires, $z_\theta = (\rho_\theta, \theta)$, ou en coordonnées cartésiennes, $z_\theta = (x_\theta, y_\theta)$.

Introduction

Avec l'augmentation globale du trafic routier, la sécurité est devenue une problématique de plus en plus importante. Limiter les risques d'accidents et les dégâts corporels sont des soucis permanents tant chez les constructeurs automobiles que pour l'État. Ces dernières années, on a pu voir de nombreuses campagnes publicitaires et de nombreuses lois visant à limiter ces risques. Depuis les années 80, de nombreuses recherches sont effectuées dans le domaine de la sécurité routière et notamment sur les systèmes d'aide à la conduite dits « intelligents » tels que les ADAS (*Advance Driver Assistance System*). Ceux-ci exploitent les informations de l'environnement proche du véhicule afin de mettre en place des systèmes d'aide à la conduite. La difficulté réside dans le choix des systèmes de perception et des traitements associés. Les travaux effectués durant cette thèse se placent dans ce cadre général. On s'intéresse à un système coopératif de perception active distribué et interconnecté par réseaux de communication (véhicule à véhicule ou V2V). Ce type de système a pour but d'accroître le champ de vision de chacun des véhicules d'une scène par la prise en compte de données provenant des véhicules situés dans son voisinage. Pour cela, chaque véhicule est composé d'un système de perception local (pour la détection et le suivi des objets) et d'un système de communication (pour la réception et le transfert des données). Dans cette thèse, nous nous sommes intéressés, plus particulièrement, au système de perception local avec l'étude des problèmes de détection et de suivi multi-objet à partir de données télémétriques de type lidar à balayage. Ainsi, notre objectif est de surveiller en permanence l'environnement proche du véhicule, par une détection et un suivi des obstacles afin de détecter les situations potentiellement dangereuses.

L'étude générale du problème de détection et de suivi a attiré notre attention sur les problèmes de pertes d'informations et les cas de fausses alarmes. Les mesures issues des capteurs de type lidar sont de nature spatialement distribuées ce qui conduit généralement à l'utilisation d'une étape de prétraitement (composée, dans ce cas, de la détection et de l'association des objets de la scène). Cette phase permet d'extraire les véhicules présents dans la scène et de délivrer une information sur leur position et vitesse relative. Il est connu que cette étape de prétraitement peut engendrer des pertes d'informations menant à des cas de non détection ou à des données aberrantes. Par ailleurs, les non-linéarités liées à la transformation polaire-cartésien des mesures lidar ne permettent plus de préserver la statistique des bruits de mesure après traitement. La conclusion qui en découle est que seule une approche exploitant directement les données brutes permet de garantir l'optimalité des traitements (association, suivi, etc.). Pour cela, on s'est intéressé plus particulièrement aux méthodes de Monte Carlo séquentielles (également appelées filtrage particulaire) pour leur aptitude à traiter toute non-linéarité de modèle de manière naturelle.

Pour résoudre ce problème, nous avons développé une approche conjointe de détection et de suivi, que nous appellerons *Integrated Tracking Particle Filter* (ITPF). Parmi l'ensemble des méthodes particulières disponibles, nous nous sommes focalisés sur les approches de filtrage hybride de type Rao-Blackwell. Ces méthodes permettent d'exploiter les linéarités du modèle d'état de manière indépendante par un filtre conditionnellement linéaire de type Kalman, optimal dans ce contexte. Le traitement de la partie non-linéaire est dévolu au filtre particulaire. Le problème ainsi posé repose sur une exécution conjointe des phases de détection et de suivi des objets dans l'espace d'étude. Nous excluons de notre approche, tout recours à une phase de détection isolée, utilisée comme entrée de l'étape de suivi. L'information utilisée correspond donc à la mesure en sortie de capteur et non pas à une information potentiellement tronquée par un détecteur.

Comme tout filtre récursif, il est connu que la vitesse de convergence de la méthode peut dépendre de l'information dont on dispose *a priori* sur l'environnement. C'est pourquoi, nous mettons en place un module de surveillance dont le but est de fournir au filtre ITPF une information *a priori* sur les objets présents. Ce module agit alors comme un module de « veille » qui observe en permanence l'environnement du véhicule porteur et alimente en continu le filtre de détection/suivi.

Comme dans toute méthode de suivi, une place importante doit être laissée à la validation des approches proposées sur des données expérimentales. Nous avons donc choisi de les évaluer sur deux simulations dont les détails sont donnés en annexe :

- un premier simulateur dont l'objectif est d'évaluer les performances en estimation qui permet, par une connaissance exacte des paramètres dynamiques de tester les performances limites de la méthode ;
- le simulateur SiVIC développé par le LIVIC qui permet, par un rendu très proche de la réalité, d'évaluer la robustesse de nos approches vis-à-vis des géométries variables d'objets, de non détection et de fausse alarme.

Dans ce contexte, il est toujours difficile de pouvoir qualifier une nouvelle approche vis-à-vis de l'état de l'art sur le domaine. Nous avons donc choisi de comparer notre approche ITPF à une méthode classique basée sur un suivi du centre de gravité après une étape de détection, pour la même méthode de filtrage particulière. Cette méthode CDG (centre de gravité) est fondée sur la mise en œuvre séquentielle de la détection, de l'association et du suivi des centres de gravité des objets de la scène. Ceci permet également d'évaluer l'apport d'une approche conjointe de détection et suivi.

Cette thèse s'articule autour de quatre chapitres :

Le premier chapitre expose le contexte général applicatif de la thèse. Il met en avant les différentes difficultés liées aux choix du capteur et de l'architecture fonctionnelle à mettre en place. Ce chapitre permettra d'amorcer nos choix pour la résolution du problème.

Le second chapitre propose un état de l'art des techniques de détection et de suivi d'objets multiples à partir de mesures lidar. Nous nous sommes inspirés des différentes étapes de détection, association et suivi d'une architecture fonctionnelle de traitement classique pour présenter les différentes étapes. Ce chapitre permet d'exposer, d'une part, les techniques de détection et de suivi les plus couramment utilisées dans la littérature et, d'autre part, il permet de proposer une comparaison de quelques méthodes de filtrage (particulière) par rapport au filtre de Kalman (plus classiquement utilisé pour le suivi de véhicule). Ce dernier point permettra de détailler les raisons de notre choix de méthode de filtrage.

Maintenant que nous avons parcouru l'ensemble du problème, le chapitre trois propose de décrire notre méthode conjointe de détection et de suivi multi-objet (ITPF). Cette méthode a la particularité de ne pas utiliser de phase de détection ni d'association, puisque la détection et le suivi sont réalisées par le filtre particulière mis en place. La méthode présentée propose d'utiliser, tout au long des traitements, la mesure brute en sortie du capteur. On s'intéressera, dans ce chapitre, au cas simplifié d'une détection et d'un suivi multi-objet de forme et taille semblable sur des données lidar monoplan (c'est-à-dire que le capteur lidar balaye la scène sur un seul plan vertical durant toute la simulation).

Les capteurs lidar monoplans ayant montré, sur route, leur vulnérabilité au tangage et roulis, le chapitre quatre se propose de traiter le cas de capteur lidar multiplan. Afin de traiter le problème dans sa globalité, nous choisissons également d'accroître notre champ de recherche à l'étude d'objet de différentes natures (tels que camion, piéton...) dans ce chapitre. De même, la possibilité d'obtenir plusieurs plans verticaux de balayage va modifier les structures de détection et de filtrage afin de prendre en compte les informations des différents plans de balayage. L'intégration des différents plans de mesures va ainsi être discutée dans un second temps.

Enfin, nous concluons sur les apports et perspectives de nos travaux.

Chapitre I

Problématique, capteurs et architectures de traitement

I.1 Problématique

Les accidents de la route sont la première cause de mortalité chez les 15 – 29 ans et restent importants dans les autres classes d'âges. Le parc automobile français, constitué de l'ensemble des véhicules immatriculés en France, est passé de 4,5 millions en 1955 contre plus de 38 millions de véhicules au premier janvier 2012. Selon les chiffres de la sécurité routière (figure I.1), issus de l'observatoire national interministériel de sécurité routière (ONISR), nous sommes passés d'environ 18000 tués sur les routes dans les années 70 à 3963 personnes tuées en 2011.

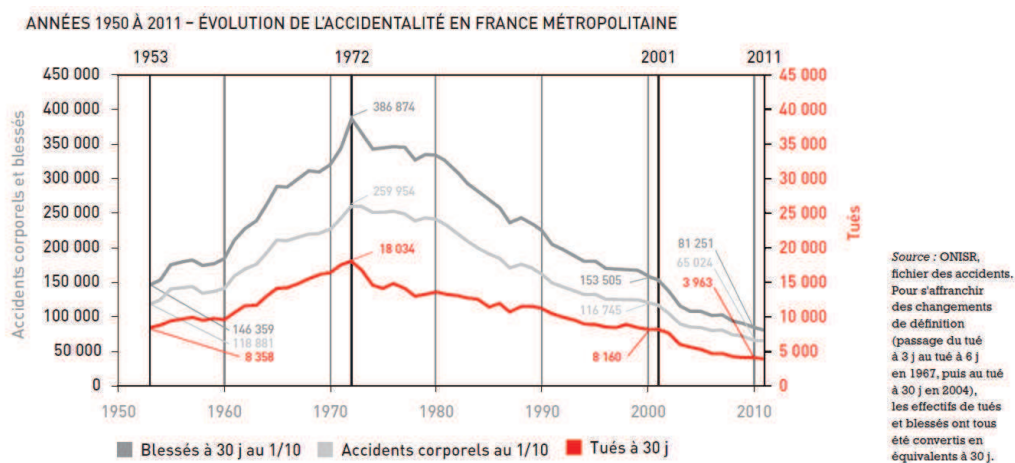


FIGURE I.1. Nombre de tués sur les routes de 1950 à 2011 (Sécurité-routière, 2012)

Il existe plusieurs causes et facteurs de risque d'accident dont nous faisons une liste non-exhaustive :

- comportement des usagers (non respect du code, ingestion de substances influençant le comportement, téléphone...),
- fatigue, baisse de vigilance et somnolence,
- vitesse,
- infrastructures routières,
- véhicules automobiles.

Bien qu'il soit impossible d'agir sur l'ensemble de ses facteurs, de nombreuses mesures sont mises en

place pour en limiter les risques. Par exemple, dans les années 70 où le taux de mortalité sur les routes atteint un pic, le port obligatoire de la ceinture de sécurité et l'apparition des limitations de vitesse sur routes et autoroutes ont initié une baisse du nombre de tués sur les routes (figure I.2 issue de l'observatoire national interministériel de sécurité routière (ONISR)). Par la suite, de nouvelles législations relatives au code de la route apparaissent. Leur mise en place permet la poursuite de la baisse de mortalité sur les routes. Les plus récentes (2012) concernent l'interdiction des avertisseurs radar et le durcissement des sanctions contre l'utilisation du téléphone portable au volant.

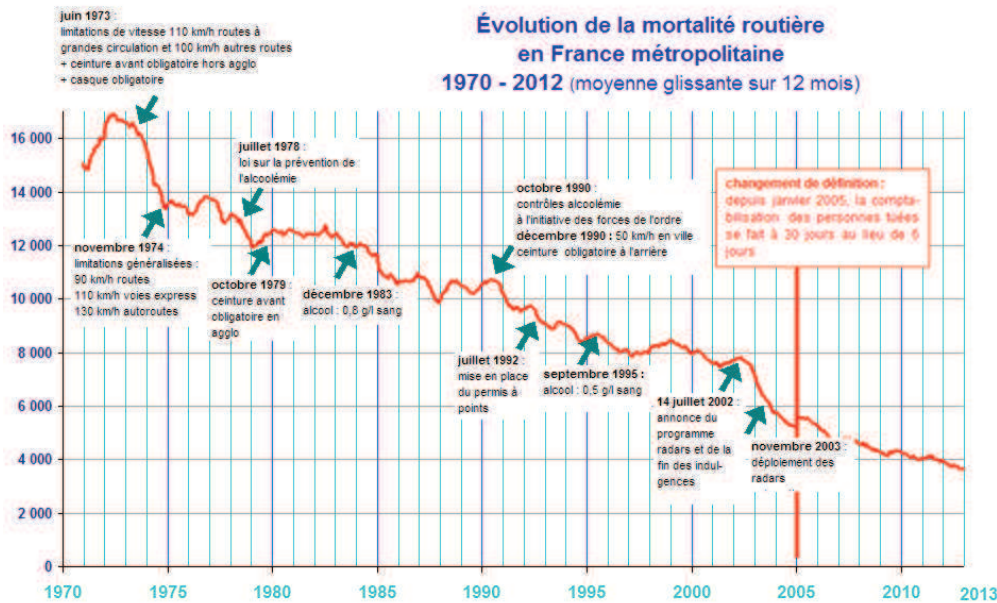


FIGURE I.2. Évolution du nombre de tués sur les routes en fonction des nouvelles législations (Sécurité-routière, 2012)

Dans le même temps, les constructeurs automobiles conçoivent des véhicules de plus en plus sûrs. Ils mettent en place des systèmes de sécurité dits actifs ou passifs :

- les systèmes passifs n'interviennent pas directement sur la conduite. Ils permettent de sécuriser les passagers ou les usagers de la route. Citons par exemple, la ceinture de sécurité, les airbags (coussins gonflables de sécurité) ou également les études sur l'optimisation des déformations de la structure du véhicule.
- les systèmes actifs eux interviennent directement sur la conduite. Ils permettent d'aider le conducteur. L'aide au freinage ABS (*Anti Blockier System*), par exemple, est apparu pour la première fois en 1966 (Wikipedia, 2013). Cependant, ces systèmes ne permettent pas d'anticiper des situations à risque mais seulement de répondre à un problème survenu. Idéalement, un système de freinage va s'activer lorsque la pédale de frein est activée. Par contre, les systèmes d'aide à la conduite dits « intelligents » sont capables d'interagir avec le conducteur sans s'y substituer. C'est ce genre de système qui est étudié dans cette thèse.

Comme nous venons de le voir, la sécurité routière est, de nos jours, très importante. Limiter les risques d'accidents est un souci permanent tant chez les constructeurs automobiles que pour l'État. Actuellement, les recherches sont centrées sur les systèmes d'aide à la conduite dits « intelligents » tels que les ADAS (*Advanced Driver Assistance Systems*). Les ADAS exploitent les informations de l'environnement proche du véhicule afin de mettre en place des systèmes d'aide à la conduite. Les recherches portent sur des aspects perceptifs comme la détection de signalisation, la détection d'obstacle, les systèmes de conduite autonome.

La détection et le suivi d'obstacle s'avèrent prometteurs pour anticiper les collisions, principales causes d'accidents corporels.

Depuis les années 90, de nombreux travaux ont été menés sur ce sujet. La difficulté réside ici dans le choix du système de perception, d'une méthode de détection et de suivi. Dans la suite, nous faisons une rapide étude des techniques d'exploitation des informations de l'environnement qui se basent sur trois principales étapes dont l'architecture est présentée en figure I.3.



FIGURE I.3. Architecture minimale de détection

Les travaux portés par les chercheurs diffèrent en particulier par le système de perception utilisé : radar (Tan *et al.*, 2007; Bank, 2007), lidar (Takagi *et al.*, 2006) et image (Tango *et al.*, 2008; Richter *et al.*, 2008). Dans la suite, nous présentons des exemples utilisant l'un ou l'autre de ces capteurs. Dans ce cadre, une thématique importante est la détection d'obstacle au voisinage du véhicule.

Dans notre contexte d'étude autoroutier, nous considérerons comme l'a fait (Perrollaz, 2008) qu'un obstacle est un objet de taille significative susceptible de couper la route de notre véhicule dans un laps de temps court, compte tenu du temps de réaction du conducteur et des vitesses mises en jeu.

Le problème peut être étudié selon deux points de vue, que cherche-t-on à détecter, et quels outils de détection peut-on mettre en œuvre ? Dans le premier cas, nous cherchons à déterminer des obstacles potentiels au voisinage du véhicule. Le type d'obstacle diffère selon l'environnement dans lequel nous nous trouvons. Par exemple, sur autoroute, les principaux objets à suivre sont des véhicules motorisés (tels que véhicule, camion...), par contre, à un carrefour, il faut également prendre en compte les piétons, les cyclistes, etc. Citons les travaux de (Koller *et al.*, 1992) qui proposent une détection des véhicules dans des séquences d'images et les travaux de (Sato *et al.*, 2010; Ogawa *et al.*, 2011) qui travaillent à partir de données lidar, pour la détection de piétons.

Après avoir décrit quel type d'objet nous cherchons à détecter, le deuxième aspect de la détection concerne les outils. Comment détecter ces objets dans des séquences d'images ou de données ? Pour cela, plusieurs méthodes existent pouvant être utilisées à la fois dans le cas de séquences d'images et dans le cas de séquences de données (radar, lidar). Deux types d'algorithmes de détection existent :

- les algorithmes de partitionnement qui consistent à regrouper les mesures qui partagent un caractère commun. Citons, par exemple, l'algorithme de *K-means*, méthode dont le but est de diviser des observations en K partitions dans lesquelles chaque observation appartient à la partition avec la moyenne la plus proche, ou encore les algorithmes reposant sur la comparaison de critère de distance entre deux données successives telles que les seuils adaptatifs de Santos (Santos *et al.*, 2003), de Borges (Borges et Aldon, 2000, 2004) et de Dietmayer (Dietmayer *et al.*, 2001).
- les algorithmes de segmentation qui consistent à retrouver des formes (droites, rectangles, etc) dans des ensembles de données. Leur but est d'extraire les contours des objets. Citons quelques exemples d'algorithmes. L'algorithme *Split and merge* (Ray et Ray, 1995) permet de réaliser des approximations polygonales de lignes par une succession de division de ces lignes en deux, puis par une phase de fusion. La transformée de Hough (Duda et Hart, 1971) cherche à déterminer les paramètres géométriques d'une forme donnée au moyen de vote. L'approche de maximisation de l'espérance (Dempster *et al.*, 1977) repose sur la recherche du maximum de vraisemblance ou du maximum *a posteriori* pour estimer les paramètres de modèle statistique. L'algorithme de Ransac, méthode destinée à estimer les

paramètres d'un modèle mathématique à partir d'un ensemble de données observées contenant un grand nombre de valeurs aberrantes.

Il reste une étape à étudier, le suivi des objets détectés. Cette étape est indispensable afin de déterminer si un objet est un futur obstacle ou non. Là encore, différentes méthodes existent. En effet, au début des années 90, dans le contexte automobile, c'est le filtre de Kalman (KF) qui est utilisé pour le suivi des objets car il a prouvé son efficacité dans de nombreux domaines applicatifs. Différentes méthodes seront alors mises en œuvre comme l'utilisation de plusieurs capteurs (Baig *et al.*, 2011; Darms *et al.*, 2008), l'utilisation des lignes de la route qui permettent d'améliorer les précisions de modèle (Tan *et al.*, 2007). Cependant, ce filtre ne permet pas l'étude de système non-linéaire. Des extensions du filtre de Kalman prenant en compte ce type de systèmes existent : le filtre de Kalman étendu (EKF) permettant de linéariser les systèmes non-linéaires et le filtre de Kalman sans parfum (UKF) qui est une extension de la transformation sans parfum (méthode pour calculer les statistiques d'une variable aléatoire issue d'une transformation non-linéaire). De même que pour le filtre de Kalman, ces filtres sont exploités. Depuis les années 2000, une nouvelle catégorie de filtres, reposant directement sur l'estimation de la densité de probabilité conditionnelle, solution du problème de filtrage, fait son apparition. Ces filtres, appelés filtres particuliers ou méthodes de Monte-Carlo séquentielles ont été introduits en 1992. Contrairement au filtre de Kalman et ses extensions, ces derniers sont applicables quelles que soient les non-linéarités de modèle.

Une autre difficulté réside dans l'environnement dans lequel nous nous trouvons. En effet, la technique mise en œuvre pour suivre un véhicule de jour sera différente de celle de nuit, compte tenu des différences de visibilité. Le jour, les véhicules sont entièrement visibles (s'ils ne sont pas occultés par les autres), par contre de nuit, les véhicules sont partiellement visibles (sans pour autant être occultés). Cela implique des méthodes de suivi variées. Considérons l'exemple de O'Malley *et al.* (2011), où les auteurs partent du constat que, la nuit, les premiers éléments visibles des véhicules sont ses phares. C'est pourquoi, ils proposent un système de traitement des images vidéo pour détecter et suivre les véhicules de nuit à partir de leurs phares. Pour le suivi de jour, il existe plusieurs méthodes, par exemple, celle de Tan *et al.* (2007), qui met en place un système de vision monoculaire guidé par radar combiné à la détection et au suivi de la signalisation horizontale de la route afin de détecter les véhicules et prédire leur intention de changer de voie. Leur but est de mettre en place des systèmes d'aide à la conduite et de prévention de collision.

On peut constater le même phénomène avec le suivi des piétons. Olmeda *et al.* (2009) proposent de détecter et suivre, de nuit, des piétons à partir d'une caméra infrarouge avec laquelle ils peuvent extraire les caractéristiques de distribution de la chaleur du corps humain. Dans les travaux de Masoud et Papanikolopoulos (2001), les auteurs s'intéressent à la modélisation et au suivi des piétons, de jour, aux intersections de type carrefour. Pour cela, ils présentent un système temps réel pour le suivi de piétons dans des séquences d'images acquises par caméra fixe.

Compte tenu du nombre important de recherches sur le sujet, il est évident que nous n'avons pas balayé l'intégralité des techniques et méthodes utilisées pour la détection et le suivi d'obstacles sur route. Cependant, nous avons mis en évidence que le choix d'un capteur, d'un outil et plus généralement d'une méthode de détection et de suivi est déterminé par le système que nous souhaitons étudier. Toute la difficulté réside donc dans les choix réalisés pour la mise en place de notre approche de détection et de suivi multicible. Quel capteur est le plus approprié ? De quels outils de détection et de suivi avons-nous besoin ? Et enfin quelle méthode allons-nous mettre en place ? Le paragraphe suivant sera consacré à l'étude des capteurs utilisés dans le domaine automobile et aux architectures fonctionnelles existantes.

I.2 Capteurs et architectures

La section précédente a mis en évidence la complexité des systèmes de suivi multi-objets, liés aux contraintes du domaine autoroutier et à l'environnement. Dans cette section, nous cherchons des solutions à ces contraintes en étudiant les différents capteurs utilisés et également les différentes architectures de traitement qui existent, pour la détection d'objets sur route. Nous étudierons dans un premier temps les différents capteurs, puis dans un second temps les architectures existantes.

I.2.1 Capteurs

I.2.1.1 Système de vision par caméra

Les systèmes de vision par caméra sont utilisés dans les systèmes d'aide à la conduite. Suivant le système mis en place, ils sont composés d'une ou plusieurs caméras. Elles permettent d'accéder à une grande quantité d'informations sur un champ de vue large (les données maximales peuvent atteindre 180° d'angle de vue sur environ 90m). L'image résultante peut être en couleur, en niveau de gris ou encore une image infrarouge, cela dépend de la caméra utilisée. L'un des inconvénients de la caméra pour ce type d'application est qu'elle ne fournit pas directement l'information de mesure des distances. Pour cela, elle doit faire appel à des algorithmes de traitement d'image plus ou moins complexes pour réaliser la détection. Dans la littérature, différents systèmes sont mis en place à partir de caméra.

- Les systèmes de vision par caméra monoculaire se distinguent généralement par le type d'objectif qui les composent. Nous pouvons distinguer deux types d'objectif, ceux de petit angle de vue, celui-ci correspond généralement à un objectif de longue focale et ceux de grand angle de vue, qui peuvent atteindre jusqu'à 180° dans la diagonale. Dans cette dernière catégorie, il existe des objectifs de type *Fisheye* (figure I.4 gauche).



FIGURE I.4. Une camera avec objectif de type *Fisheye* à gauche et photo du ciel prise avec la caméra à droite

L'intérêt premier de ces caméras est leur grand angle de champ, cependant, celui-ci cause une distorsion de l'image. La figure I.4 de droite montre une photo prise par la caméra *Fisheye* appartenant au LEOST, c'est une vue du ciel qui présente des distorsions (le centre est grossi par rapport aux côtés de l'image). L'utilisation de ce type de caméra implique donc de mettre en place des traitements particuliers afin de réduire ces effets de champ.

Dans le contexte routier, la plupart des algorithmes de détection visent à trouver dans l'image un modèle d'objet connu, le plus souvent voiture ou piéton. Selon (Perrollaz, 2008), deux étapes sont nécessaires pour la détection. La première consiste à extraire des primitives pouvant appartenir à des obstacles (par exemple, utiliser les contours images des objets, utiliser les symétries des objets à suivre, les textures et les ombres, l'utilisation de capteur particulier comme les caméras infrarouges permettent de distinguer efficacement les sources de chaleur), et la deuxième cherche à vérifier que celles-ci correspondent effectivement aux modèles d'obstacles recherchés.

- Les systèmes stéréoscopiques sont généralement constitués de deux caméras identiques fixées sur un support rigide. Elles sont alignées, et ont leur objectif parallèle, pointant vers la même direction, sans pour autant converger. La figure I.5 de gauche montre un système stéréoscopique appartenant au LEOST. Il est composé de deux caméras positionnées sur un support rigide à une distance connue l'un de l'autre. Le système est tel que les deux caméras sont déclenchées au même instant. Ceci implique une synchronisation importante d'autant plus si le ou les objets à suivre sont mobiles. L'utilisation de ce genre de système permet d'obtenir des informations de profondeur de la scène. Il est également possible de faire de la vision stéréoscopique avec une unique caméra en utilisant deux instants successifs et la donnée temps, les traitements des images sont ensuite similaires.



FIGURE I.5. Système stéréoscopique expérimental (LEOST) à gauche et image d'une scène prise par un système stéréoscopique et sa carte de disparité à droite

La stéréovision permet d'obtenir une image 3D de la scène à partir des deux images prises au même instant. Pour cela, il existe plusieurs techniques décrites dans (Perrollaz, 2008) que nous présentons rapidement :

Les approches par appariement qui correspondent à la mise en correspondance des informations issues des différentes images du système stéréoscopique. Cette méthode correspond à l'algorithme de *stereo matching* qui cherche à définir les pixels en correspondance, c'est-à-dire, ceux formés par un seul point de l'objet observé sur chacune des images. Cela permet de construire une image ou carte de disparité, la segmentation de celle-ci permet de réaliser la détection des objets. La figure I.5 de droite, prise avec le matériel du LEOST, montre une scène prise sur autoroute à gauche, et à droite sa carte de disparité associée. La carte de disparité correspond à une image de profondeur, on l'interprète ainsi, lorsque les couleurs tendent vers le rouge foncé, cela signifie que l'on est proche du véhicule porteur, à l'inverse lorsque les couleurs tendent vers le bleu foncé, cela signifie que l'on est loin du véhicule porteur.

Les approches par rectification homographique qui consistent à appliquer une transformation homographique à l'une des images, ou une transformation perspective inverse aux deux, pour que l'apparence de la route soit semblable dans les deux images. Les pixels n'appartenant pas au plan de la route sont alors extraits par la différence entre les deux images, associés à un seuillage. Ces pixels sont alors segmentés et correspondent aux objets.

Les systèmes de vision par caméra sont intéressants car ils permettent d'obtenir des informations sur une distance pouvant atteindre les 90 mètres. En dépit de leurs hautes performances en vision de la scène, on distingue quatre types de contraintes relatives à l'information transmises par une caméra :

- les problèmes de synchronisation des différentes caméras entre elles,
- les temps de transfert,
- les temps de traitement,
- les problèmes de stockage.

Les trois dernières sont consécutives à la quantité d'information à traiter. En effet, la quantité d'information contenue dans une image est importante (2048x1944 pixels) suivant la résolution de l'objectif, le temps de transfert, le temps de traitement et les problèmes de stockage d'une image dépendent de la quantité d'information qu'elle contient. Il est possible de réduire cette quantité d'information en réduisant la taille de l'image

cependant, cette manipulation implique une perte de précision qui, si elle est trop prononcée, implique une perte de précision au niveau des traitements. Une autre possibilité pour "réduire" la quantité d'information à traiter est de choisir de ne pas traiter toutes les images en provenance de la caméra. Cela permet de stocker une quantité d'information moins importante mais également de réduire le coût de traitement.

I.2.1.2 Radar

Le radar est un système utilisant les ondes radio afin de détecter la présence, la position et la vitesse d'objets. Pour cela, il émet une onde produite par un oscillateur radio. C'est la réflexion de cette onde sur un objet qui sera analysée à son retour. Le temps de vol aller et retour mesuré permet de localiser l'objet détecté. Il existe différentes façons d'émettre ces ondes. Les plus utilisées sont :

- les radars à impulsions, où le radar émet périodiquement une impulsion et attend son retour. Le temps de vol aller et retour de l'onde entre l'antenne et la cible est alors mesuré.
- les radars à émission continue qui émettent continuellement à partir d'une antenne et reçoivent à l'aide d'une seconde.

L'analyse du signal réfléchi offre la possibilité de localiser et d'identifier l'objet responsable de la réflexion, ainsi que le calcul de sa vitesse de déplacement grâce à l'effet Doppler (dans le cas des radars cohérents). Ainsi, le radar peut détecter des objets ayant une large gamme de propriétés réfléchissantes alors que les autres types de signaux, tels que le son ou la lumière visible revenant de ces objets, seraient trop faibles pour être détectés. De plus, les ondes radio peuvent se propager avec une faible atténuation à travers l'air et divers obstacles tels les nuages, le brouillard ou la fumée, qui absorbent rapidement un signal lumineux. Cela rend possible la détection et le pistage dans des conditions qui paralysent les autres technologies.

Dans le domaine automobile, citons à titre d'exemple les travaux de Bank (2007) qui introduisent un algorithme de traitement du signal Radar pour le développement d'une fonction d'alerte de collision latérale. Pour cela, ils utilisent un capteur radar SRR (figure I.6) de fréquence 24,125GHz pour la détection d'objets de 0,2m à 30m avec une résolution en distance de 0,15m. Ils utilisent un filtre de Kalman pour réaliser le suivi multicapteur, multicible. Pour appliquer ce filtre, un modèle de vitesse constante est choisi. Enfin, l'approche globale du plus proche voisin est utilisée pour associer les nouveaux objets avec des pistes existantes.

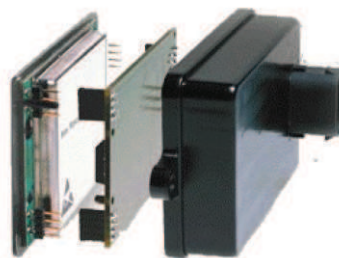


FIGURE I.6. Image d'un radar SRR extraite de (Bank, 2007)

Skutek *et al.* (2003) mettent en place un système anticollision pour voiture, fonctionnant sous toutes les conditions météorologiques. Il est basé sur quatre radars courte portée de 24GHz et un système de traitement du signal associé. Ce dernier fournit l'information sur une situation de collision potentielle. Le système est décrit en quatre étapes, d'abord, les mesures passent dans un filtre d'entrée qui crée un masque et filtre la distance entre objet et radar. Ensuite, la vitesse relative des objets est estimée. Puis, les auteurs estiment les distances critiques entre deux véhicules et enfin, ils classifient les objets par rapport à cette distance.

Perrollaz (2008) montre que le radar présente quelques limites pour les applications automobiles. En premier lieu, l'angle de vue du capteur est faible, environ 10 degrés pour un radar longue portée ($\sim 120m$) et

70 degrés pour les radars courtes portées ($\sim 30\text{m}$). Cette faible résolution angulaire ne permet pas toujours de distinguer deux objets situés côte à côte par exemple. En second lieu, la présence de nombreux échos parasites dans la chaussée ou sur les bords de la route provoque un grand nombre de fausses détections. Pour limiter ces fausses détections, les objets situés en milieu de voie et immobiles par rapport au véhicule porteur sont filtrés, par comparaison de leur vitesse avec celle du porteur. Enfin, ce type de capteur ne permet pas la détection fiable des piétons car ceux-ci créent peu d'échos.

I.2.1.3 Lidar

Un télémètre laser permet de réaliser une mesure précise entre le capteur et un obstacle. Ce système de mesure, également appelé LIDAR (*Light Detection And Ranging*) est conçu à l'aide de lasers. C'est un système utilisant la lumière afin d'estimer la distance d'un objet. Pour cela, il émet une onde lumineuse, la distance à un objet ou à une surface est donnée par le délai entre le temps d'émission de l'impulsion et le temps de détection du signal réfléchi. Contrairement au radar, le lidar permet de détecter tout type d'objets (aussi bien véhicule que piéton) à partir du moment où celui-ci renvoie tout ou une partie de la lumière incidente. Les mesures transmises par le lidar correspondent à une distance et un angle.

À la différence du radar, le capteur lidar émet une onde très focalisée et non sur un cône d'ouverture angulaire donnée. Dans le domaine autoroutier, ce sont les capteurs télémétriques à balayage qui sont généralement utilisés car ils permettent de balayer la scène par envoi de multiple rayons. L'orientation du rayon est permise par un miroir rotatif, commandé par un moteur électrique. Cette technique permet d'orienter le rayon dans un grand nombre de direction, ainsi un angle de vue large peut être étudié et ce avec une grande précision angulaire.

En raison de sa directivité plus grande, le capteur lidar est plus sensible aux mouvements du véhicule que les autres capteurs. Le tangage de celui-ci peut provoquer une désorientation du rayon capteur et ainsi toucher la route émettant un écho factice. L'utilisation d'un capteur télémétrique multicouche permet de remédier partiellement à cette difficulté en donnant une vision 3D de la scène. Ce type de capteur émet plusieurs niveaux de tir comme le montre la figure I.7. Des traitements et un paramétrage supplémentaire sont donc indispensables.

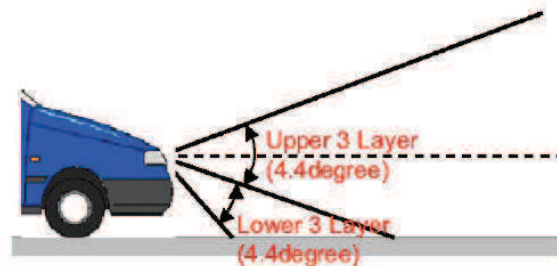


FIGURE I.7. Principe du lidar multicouche

Takagi *et al.* (2006) présentent un algorithme de reconnaissance de l'environnement routier à partir d'un capteur lidar. Pour cela, ils utilisent un capteur DENSO multiplan à trois nappes pour détecter les véhicules et les signalisations horizontales jusqu'à 120m. Dans cet article, la fréquence est de 10Hz, l'élévation du rayon est de $(-4^\circ, 4^\circ)$ et son azimut est de $(-18^\circ, 18^\circ)$ avec une résolution angulaire de $0,08^\circ$. Les auteurs utilisent un algorithme de détection en trois étapes, reconnaissance des objets, reconnaissance des signalisations horizontales et reconnaissance de l'environnement routier par l'intégration des deux premières étapes.

Nous choisissons de nous focaliser sur le capteur lidar pour résoudre notre problème de détection et de suivi d'obstacle. En effet, ce capteur permet d'obtenir une grande quantité d'informations concernant la

scène, et ceci avec une faible demande en mémoire. Ceci a notamment un impact sur les temps de transfert et les problèmes de stockage. La capacité du capteur à détecter tous types d'objets et cela sur une distance pouvant atteindre 120m sur une ouverture angulaire de $(-18^\circ, 18^\circ)$ (exemple de (Takagi *et al.*, 2006)) correspond à nos besoins. Un dernier aspect est la capacité des fournisseurs à gérer les difficultés liées au domaine automobile, avec notamment la mise en place d'un balayage horizontal et d'un balayage vertical pour pallier les difficultés liées au contexte routier (déplacement sur la route, tangage). La section I.3 concernant le lidar réalise une étude plus poussée des capteurs lidar avec notamment un tour d'horizon des différents capteurs du marché.

I.2.2 Architectures de traitement de données

Dans cette section, nous cherchons à caractériser les différentes architectures de traitement des systèmes de détection et de suivi d'obstacles. Ces systèmes sont représentés par une suite de tâches liées, qui effectuées les unes après les autres, permettent d'obtenir l'objet suivi ou à suivre à partir des données reçues en entrée du système.

L'architecture de traitement d'un problème de détection et de suivi d'objets varie suivant plusieurs points : le nombre de capteurs utilisés, l'utilisation d'une tâche de fusion de données, le nombre d'objets à suivre. Nous présentons ici les architectures permettant la détection et le suivi d'un ou plusieurs objets à partir d'un ou plusieurs capteurs. Les cas multicapteur et monocapteur sont étudiés séparément.

I.2.2.1 Architectures multicapteur classiques

La présence de plusieurs capteurs implique la présence d'une étape de fusion de données. Dans les livres (Waltz et Llinas, 1990; Hall, 1992; Hall et Llinas, 1997), les auteurs présentent trois types distincts d'architecture basés sur la fusion des données que nous présentons ci-après :

- architecture de fusion centralisée (figure I.8). Dans cette approche, les données brutes de chaque capteur sont transmises à une unité de traitement composée successivement d'une étape de fusion centralisée qui réalise l'association des données, suivie par une étape de corrélation, une étape de filtrage qui réalise l'estimation dynamique et enfin une étape de classification des objets ciblés. Ce type d'architecture permet d'obtenir à la fois une estimation du vecteur d'état de chaque objet mais également une classification des objets suivis.

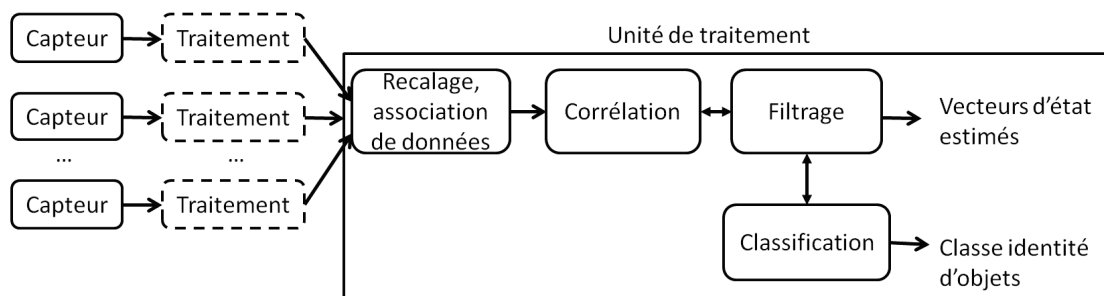


FIGURE I.8. Architecture de fusion centralisée extraite de (Waltz et Llinas, 1990)

- architecture de fusion autonome (figure I.9). Les données brutes de chaque capteur sont localement prétraitées pour générer un vecteur d'état et classifier les objets. Les informations résultantes sont transmises à l'unité de traitement, identique à celle de l'architecture de fusion centralisée, pour le calcul de l'estimation d'état.

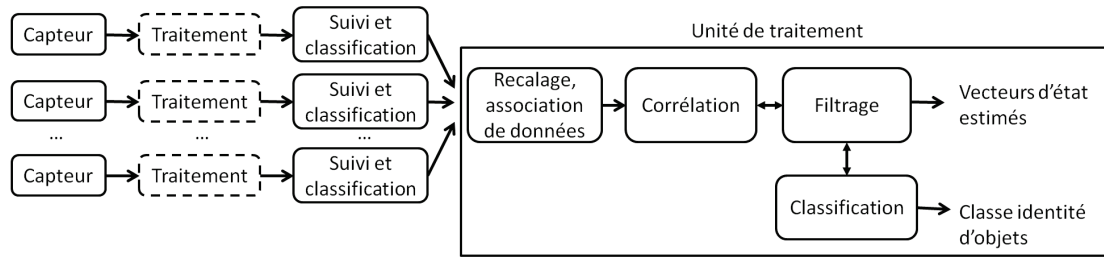


FIGURE I.9. Architecture de fusion autonome extraite de (Waltz et Llinas, 1990)

- architecture de fusion hybride (figure I.10). Cette architecture est une combinaison des architectures centralisées et autonomes. On a donc deux étapes distinctes. La première partie de l'architecture est identique à celle de fusion autonome. Ensuite, une deuxième partie récupère les informations brutes en sortie des capteurs et les transmet à une unité centrale qui effectue la détection suivant le même schéma que l'architecture de fusion centralisée. Les informations résultantes sont transmises à l'unité de filtrage qui calcule l'estimation des vecteurs d'états et classe les objets.

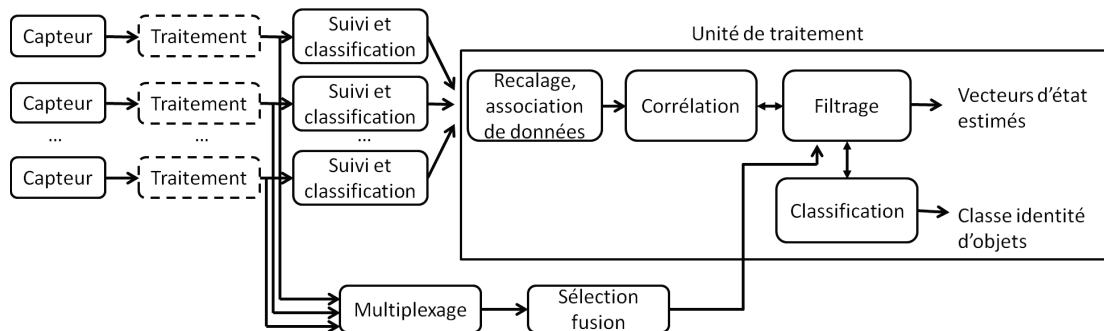


FIGURE I.10. Architecture de fusion hybride extraite de (Waltz et Llinas, 1990)

Dans ces trois cas, on utilise les données brutes de chacun des capteurs. Il existe également une autre alternative qui ajoute une étape de prétraitement local associé à chaque capteur (en pointillés sur les figures). Cette étape permet d'extraire des données brutes les vecteurs caractéristiques pour chaque objet détecté. C'est cette étape de prétraitement qui diffère suivant le capteur utilisé.

Dans ce type d'architecture, l'abondance de capteurs peut vite devenir problématique et ce pour différentes raisons. L'utilisation de capteurs de types différents implique des prétraitements de données locaux afin de fusionner des informations de même type, ce qui peut augmenter le coût de calcul. Par contre, cette multiplicité des informations est un avantage majeur qui permet la redondance des informations. Ainsi, un objet peut être vu par plusieurs capteurs. Cela a pour avantage de limiter les fausses alarmes, non détection et autres. Ceci implique également que la défaillance d'un capteur peut être compensé par les autres ce qui n'est pas le cas dans les architectures monocapteurs.

Quand plusieurs capteurs coexistent, un autre souci est la synchronisation des données. En effet, il est possible que les informations émises par les différents capteurs ne proviennent pas des mêmes instants rendant nécessaire l'utilisation d'une étape de synchronisation.

I.2.2.2 Exemple d'une architecture autonome dans le domaine automobile

Le projet européen CARSENSE (Wahl, 2002) a conçu un démonstrateur de fusion de données multi-capteurs pour détecter les objets devant le véhicule porteur. Son architecture comporte (figure I.11) deux capteurs extéroceptifs autonomes dotés de capacité locale de suivi (laser scanner) et de traitement d'images (unité vidéo) et des capteurs proprioceptifs fournissant au système de fusion de données et aux capteurs extéroceptifs les informations d'angle de braquage, de vitesse, de vitesse de lacet et d'accélération latérale du véhicule. Dans cette architecture, l'unité de fusion est en charge de la gestion centralisée du bus capteur auquel sont connectés les unités vidéo et laser scanner. Elle transfère périodiquement du bus véhicule au bus capteur, les informations proprioceptives nécessaires aux capteurs extéroceptifs. Elle réalise également la fusion des informations du laser scanner et de l'unité vidéo sur un matériel spécifiquement conçu pour accueillir les algorithmes de fusion de données.

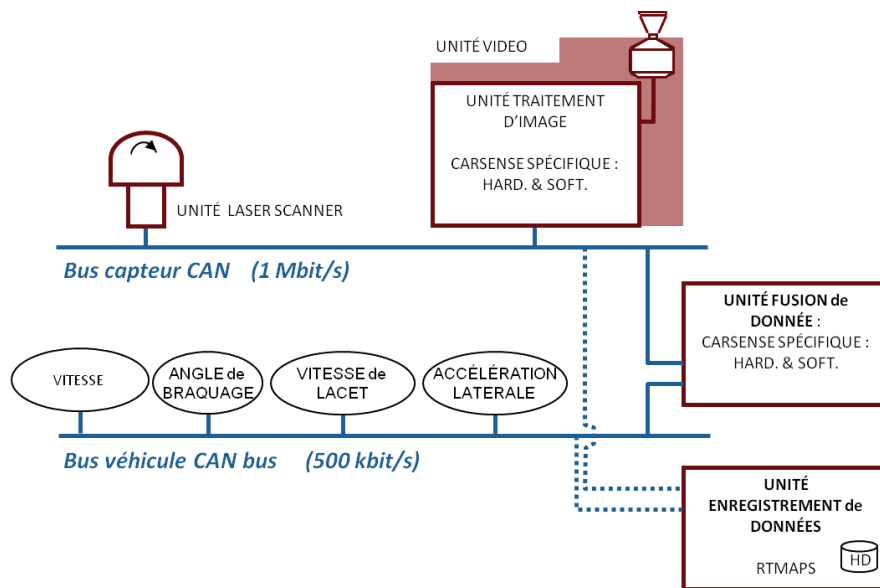


FIGURE I.11. Architecture du système de fusion de données du projet CARSENSE extraite de Wahl (2002)

I.2.2.3 Architecture monocapteur pour lidar

Dans le cas d'une détection et d'un suivi monocapteur, l'architecture de traitement est simplifiée quant au nombre de capteurs mais reste similaire au niveau des étapes de traitement. Prenons l'exemple monocapteur (figure I.12) du projet CARSENCE (Kiehn *et al.*, 2002). L'objectif est le suivi des objets, via des informations de haut niveau, obtenues à partir d'un laser à balayage. Les données brutes sont segmentées puis transmises à un module d'affectation, correspondant à la fusion. Celui-ci associe les données provenant de la segmentation à celles de la prédiction des objets (de l'instant précédent). Les informations résultantes sont transmises à un module de classification des objets puis à un module de mise à jour, il en ressort l'objet suivi.

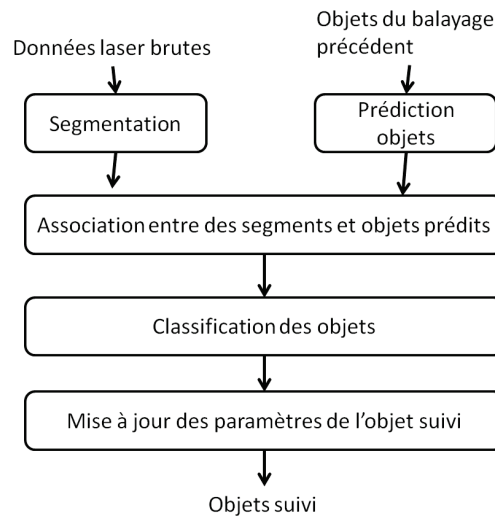


FIGURE I.12. Architecture de traitement monocapteur lidar extraite de (Kiehn *et al.*, 2002)

Si maintenant nous observons le choix d'architecture monocapteur de Jida (2008) présenté en figure I.13. On remarque que l'architecture est simplifiée au niveau des capteurs, *a contrario* elle se complique fortement au niveau de l'unité de traitement. Dans cette thèse (Jida, 2008), l'auteur cherche à estimer les paramètres dynamiques des objets observés sur une scène afin de mettre en place un système d'aide au conducteur. Pour cela, il part du constat que la nature du suivi multi-objets implique d'associer, efficacement et à chaque instant, les mesures disponibles à chaque objet présent dans la scène pour permettre une estimation de leurs paramètres dynamiques. Dans un premier temps, les paramètres prédéfinis des objets sont extraits par trois phases :

- une phase d'acquisition des signaux qui utilise un capteur rotatif laser,
- une phase de prétraitement qui met en forme les signaux sous forme de mesures,
- une phase de détection des objets qui identifie des groupes de mesures correspondant à des objets à suivre selon différents critères et connaissances, *a priori*, de la forme des obstacles.

Dans un second temps, il met en relation les objets détectés avec les données précédentes. Cela correspond à l'étape d'association. Il s'en suit une étape d'estimation, de suivi et enfin de prédiction correspondant notamment aux différentes phases d'un filtre prédictif de Kalman et à la gestion du suivi des objets dans le temps. Des pistes sont créées (une piste est un ensemble de variables caractérisant les objets suivis), d'autres sont maintenues et mises à jour en fonction des apparitions et disparitions des objets détectés devant le véhicule, enfin certaines sont détruites lorsqu'aucune nouvelle observation ne leur a été affectée depuis un temps prédéfini.

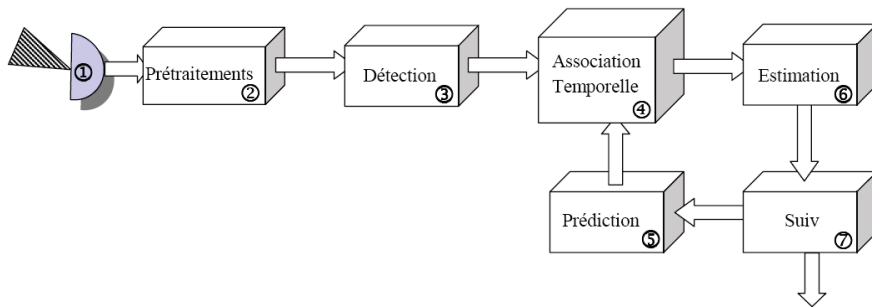


FIGURE I.13. Architecture de traitement monocapteur lidar extraite de Jida (2008)

I.3 Le télémètre laser à balayage : capteur lidar

I.3.1 Principe du télémètre laser à balayage

La technologie laser est directive, elle permet d'envoyer un signal lumineux dans une direction donnée. Dans les systèmes embarqués, la directivité du signal lumineux a pour conséquence le décrochage régulier de l'obstacle suivi, qui entre et sort régulièrement de la ligne de tir. Pour pallier cette directivité, les concepteurs de capteur télémétrique ont, à partir des années 2000, augmenté l'ouverture angulaire horizontale des capteurs (Fuerstenberg *et al.*, 2001c; Velodyne, 2013; Ogawa et Takagi, 2006). Ainsi le signal balaye la scène (figure I.14 haut) sur l'ouverture angulaire. Ce balayage est rendu possible par l'utilisation d'un prisme rotatif couvrant 360° . La résolution angulaire varie en général de $0,25^\circ$ à 1° et dépend de la fréquence du balayage. La portée du laser dépend de la taille et de la réflectivité de l'objet. Par exemple, pour un objet de forte réflectivité, la portée peut atteindre 250m à l'inverse pour un objet peu réflectif (5% de réflectivité) la portée atteint les 40m (Fuerstenberg et Willhoeft, 2001a,b; Fuerstenberg *et al.*, 2001c).

Ces capteurs sont également sensibles au roulis et au tangage du véhicule, dans ce cas, le signal lumineux change d'inclinaison, cela a pour conséquence le décrochage de l'obstacle suivi, menant comme précédemment à des pertes du signal. Dans les années 2000 – 2002 la société IBEO a conçu, à partir d'une première expérience (capteur laser scanner IBEO LD), une nouvelle génération de capteurs capables de mesurer les distances sur différents plans verticaux de balayage (IBEO LDML). Ceci correspond au balayage vertical (figure I.14 bas) de la scène et permet d'être moins sensible au tangage du véhicule (Fuerstenberg *et al.*, 2002a,b, 2003). Le capteur balaye ainsi la scène suivant deux plans, un vertical et un horizontal.

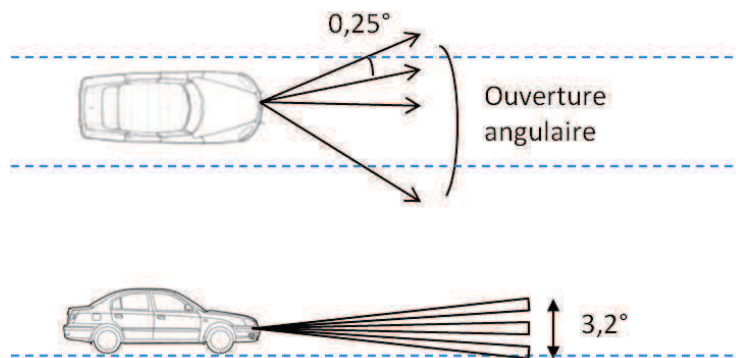


FIGURE I.14. Principe du balayage horizontal (en haut) et vertical (en bas) extraite de (Besesty, 1999)

Dans notre contexte d'étude, c'est la technologie laser à balayage qui est donc utilisée. Le laser émet une onde lumineuse par son émetteur laser et la reçoit via son récepteur comme le montre la figure I.15. La distance à la cible étant proportionnelle à l'intervalle de temps entre transmission et réception de l'impulsion, c'est le temps de vol aller et retour de l'émetteur au récepteur en passant par la cible qui donne la distance à l'objet.

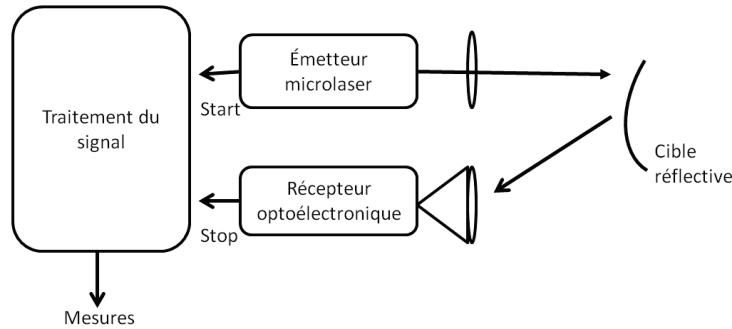


FIGURE I.15. Schéma de fonctionnement du télémètre laser à balayage extrait de (Besesty, 1999)

Selon (Besesty, 1999), il existe deux méthodes pour mesurer les temps de parcours :

- un faisceau lumineux est modulé, dans ce cas, c'est la notion de retard de phase qui est importante pour la mesure. Cette méthode est le plus souvent employé pour mesurer des distances courtes,
- une impulsion optique courte est générée, le système mesure alors le temps mis par la lumière pour revenir au récepteur. Cette méthode est la plus souvent utilisée pour déterminer des grandes distances.

La distance correspond alors à :

$$D = \frac{c}{2}(T_{stop} - T_{start}) \quad (I.1)$$

où D est la distance du capteur à la cible, c est la vitesse de la lumière, T_{start} est le temps de départ du signal et T_{stop} celui de fin d'émission (comme le montre la figure I.16).

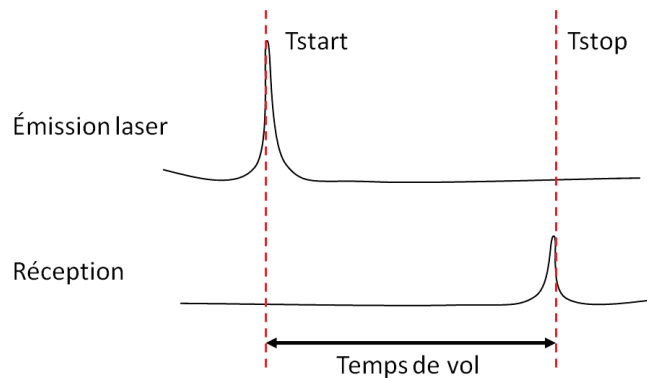


FIGURE I.16. Principe de mesure du temps de vol d'une onde émise par le télémètre laser à balayage extrait de (Besesty, 1999)

I.3.2 Exemple de mesures lidar issue de capteur monoplan et multiplan

Le capteur lidar est généralement muni d'un système interne de traitement qui permet de transformer le signal présenté ci-dessus en échos (ρ, θ) . Ce sont ces échos qui sont délivrés par le lidar et que nous étudions, c'est pourquoi nous les présentons, dans cette section.

Les mesures télémétriques d'un lidar décrivent la scène par un ensemble de points, appelés impacts ou échos, correspondant à la scène qu'il balaye. Dans cette scène, les véhicules présents sont représentés par des ensembles d'impacts correspondant à la partie visible de leurs contours. Ces ensembles de points contiennent un nombre variable d'impacts suivant l'orientation et la distance de l'objet. Ils peuvent être séparés en segment contour du véhicule représentant un ou deux côtés du véhicule (Fuerstenberg *et al.*, 2003).

Les figures présentées ci-après sont issues d'une simulation, d'un capteur de type télémètre laser à balayage, avec le logiciel SiVIC (Livic, 2011). Le véhicule porteur du capteur est situé voie centrale. Les figures I.17 du bas présentent deux types de mesures, converties dans le repère cartésien. La première (à gauche) représente les mesures de la scène prises par un capteur monoplan et l'autre (à droite) celles prises par un capteur multiplan. La scène représentée par les deux figures est celle présentée en figure I.17 du haut où quatre véhicules, situés à des distances différentes, évoluent sur les trois voies de la route.

Sur la figure I.17 de gauche, le capteur lidar monoplan visualise les données de la route sur une ouverture angulaire de 180° avec une résolution angulaire de $0,25^\circ$. La portée du capteur est de 10m à 180m (les éléments considérés à l'infini sont à 180m). La fréquence d'échantillonnage est de 20Hz. Le capteur lidar multiplan (à droite) visualise les données de la scène avec les mêmes caractéristiques que le capteur monoplan. Les tirs capteurs sont effectués suivant une élévation de 0° en bleu, $0,4^\circ$ en rouge, $0,8^\circ$ en vert et $1,2^\circ$ en cyan.

Sur la figure I.18 de gauche, un agrandissement, du véhicule situé en bas à droite de la scène vue par le capteur lidar multiplan (figure I.17 à droite), permet de montrer les différents plans déterminant un objet. Ce véhicule est proche du véhicule porteur du capteur, l'élévation du tir permet de voir tous les plans, par contre pour le véhicule situé voie centrale, seuls les plans d'élévation 0° et $0,4^\circ$ sont visibles (figure I.18 de droite). Cette disparition des informations constitue la principale difficulté de l'étude de ces données capteurs multiplans.

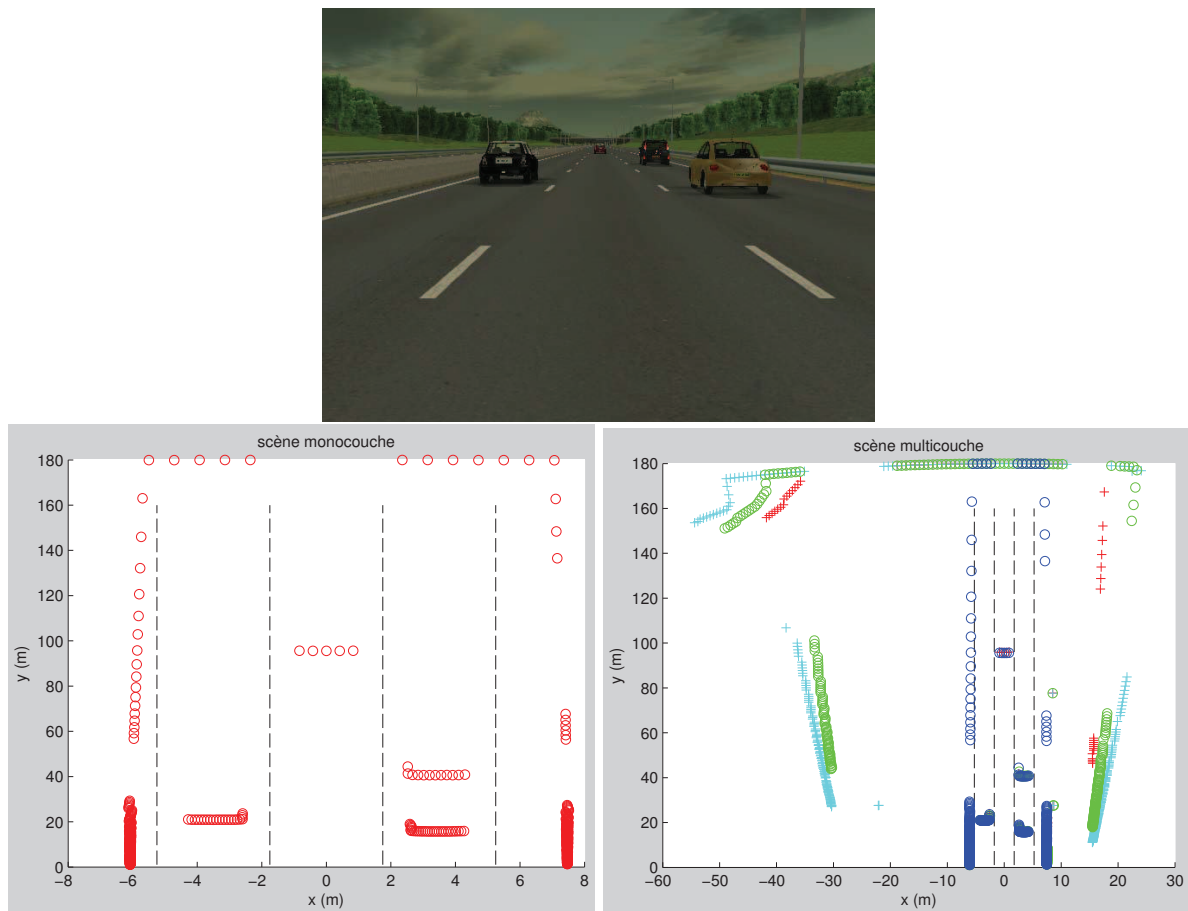


FIGURE I.17. Image CCD de la scène et le balayage capteur associé vu par le lidar monoplan (en bas à gauche) et par le lidar multiplan (en bas à droite)

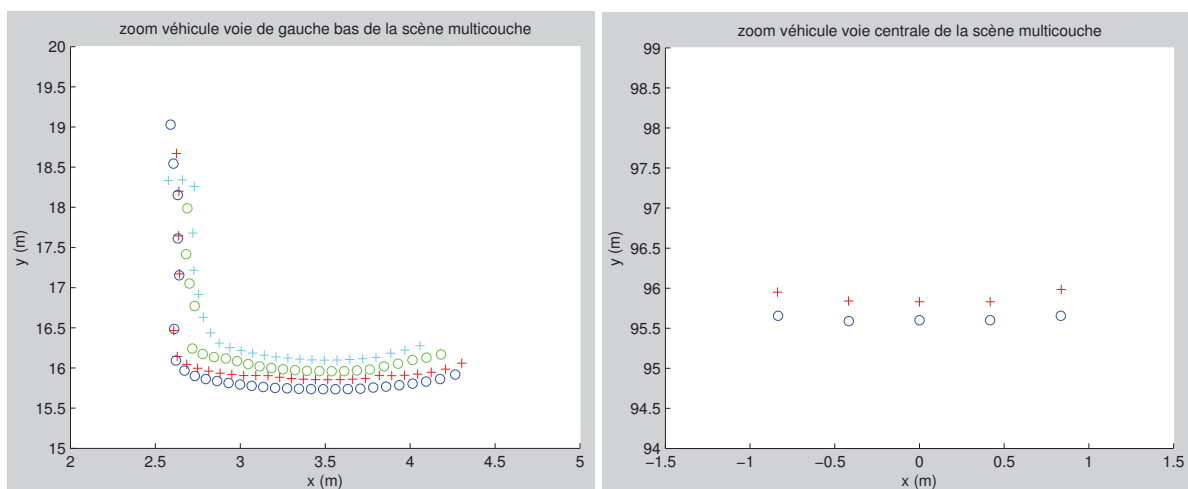


FIGURE I.18. Agrandissement de la représentation de la scène vue par le lidar multiplan (figure I.18 à gauche) sur le véhicule en bas de la voie droite (à gauche) et sur le véhicule voie centrale (à droite)

I.3.3 Évolution des capteurs lidar

Après avoir vu le fonctionnement des capteurs lidar, nous nous focalisons sur leur évolution, dans le domaine automobile, en décrivant quelques exemples de capteur. Bien que similaires dans leur fonctionnement, ils diffèrent par leurs caractéristiques et leurs évolutions. Différents fournisseurs réalisent et commercialisent des capteurs de type lidar, tel que DENSO, IBEO, SICK et Velodyne.

Dans la littérature, beaucoup d'articles font références aux capteurs de la gamme IBEO. Par contre pour les fournisseurs DENSO, SICK et velodyne, peu d'articles y font référence et nous trouvons peu d'informations concernant les capteurs qu'ils réalisent. (com, 2014) donne tout de même une comparaison rapide des capteurs DENSO, SICK LD-MRS et Velodyne HDL-64E présentés en figure I.19 et tableau I.1.



FIGURE I.19. Présentation des capteurs DENSO, SICK LD-MRS et Velodyne HDL-64E (de gauche à droite) extrait de (com, 2014)

	DENSO	SICK LD-MRS	Velodyne HDL-64E
Distance	120m	> 50m pour 10% de réflectivité	50m pour 10% de réflectivité, 120m pour 80% de réflectivité
Précision en distance	< 10m	< 100mm	< 20mm
Angle d'ouverture horizontale	36°	110° (2 plans), 85° (4 plans)	360°
Angle d'ouverture verticale	8°	1,6° (2 plans) 3,2° (4 plans)	26,8°
Fréquence de balayage	10Hz	12,5Hz, 50Hz	5 – 15Hz
Résolution angulaire	0,08°	0,25° (à 12,5Hz) et 0,5° (à 50Hz)	0,09°
Nombre de plan	6	2 ou 4	64

Tableau I.1. Comparaison des capteurs DENSO, SICK LD-MRS et Velodyne HDL-64E extrait de (com, 2014)

Nous choisissons d'étudier les capteurs de la gamme IBEO et leurs évolutions. La société IBEO, créée en 1998 et située à Hambourg en Allemagne devient « IBEO Automobile Sensor » en devenant filiale de la société SICK. Son activité est concentrée sur les lasers à balayage pour l'automobile.

I.3.3.1 Capteur IBEO LD Automotive

Le premier capteur réalisé et commercialisé par IBEO, est un capteur laser à balayage « IBEO LD Automotive » à forte résolution. Le système mis en place est généralement muni d'un capteur et d'un ordinateur pour le traitement interne des données. Le champ de vision de ce capteur est ouvert à 270°, il peut détecter un véhicule à partir de 3m jusqu'à 250m suivant la réflectivité de l'objet (40m pour un objet à 5% de réflectivité) avec une précision de $\pm 5\text{cm}$. Sa résolution varie de 0,25° à 1° pour une fréquence allant de 10Hz à 40Hz (Fuerstenberg et Willhoeft, 2001a,b; Fuerstenberg *et al.*, 2001c).

Fuerstenberg et Willhoeft (2001a) partent du constat que la classification des objets détectés est importante pour estimer la dangerosité des obstacles. Ils constatent, par ailleurs, qu'il est intéressant d'obtenir des informations sur le risque potentiel des objets environnants afin de protéger les piétons en cas de collision. Les auteurs présentent, dans cet article, le laser « IBEO LD Automotive » et leurs algorithmes de détection, suivi et classification. Les mesures brutes, en sortie du capteur, sont divisées en groupes représentant un même objet. Un filtre de Kalman est utilisé pour estimer la vitesse longitudinale et latérale de l'objet. Enfin, les objets sont classés suivant les données.

Ce capteur a également été utilisé dans le projet CHAMELEON (« Pre-crash application all around the vehicle »). Ce projet portait sur le développement d'une plateforme sensorielle de détection de collision imminente, autour du véhicule et pour tout type de scénario (Fuerstenberg *et al.*, 2001d; Chameleon, 2013). Dans ce projet, le capteur « IBEO LD Automotive » a été utilisé pour la détection, le suivi et la classification des objets (voiture, piéton, ...).

I.3.3.2 Capteur IBEO LD-ML (*Ladar Digital MultiLayer*)

(Fuerstenberg *et al.*, 2002a,b, 2003) partent du constat que le capteur « IBEO LD Automotive » est sensible aux mouvements du véhicule, perturbant ainsi la détection. La société IBEO présentent alors une nouvelle génération du capteur IBEO (décrit dans ces articles) permettant de pallier ce problème : le capteur « IBEO LD-ML » (*Ladar Digital MultiLayer*). Celui-ci possède des caractéristiques similaires à celle du capteur « IBEO LD Automotive ». Elles sont présentées en figure I.20.



Caractéristiques techniques :

- Fréquence de balayage 10Hz à 40Hz
- Angle d'ouverture horizontale : jusqu'à 270°
- Angle d'ouverture verticale : 3,2° (4 plans de balayage)
- Distance : jusqu'à 120m (généralement de 0,3m à 65m)
- Résolution en distance $\pm 5\text{cm}$
- Résolution angulaire de 0,25° (à 10Hz) à 1° (à 40Hz)
- Compensation automatique du tangage : oui
- Performance au temps : robuste à la pluie

FIGURE I.20. Le capteur IBEO LD-ML (à gauche) et ses caractéristiques (à droite)

Ce capteur a la capacité de détecter deux distances avec un unique tir laser. Cela se produit si le faisceau frappe une cible à faible réflectivité et transmission élevée ou des cibles de petites tailles tels que le verre ou les gouttes de pluie. Dans ce cas, une partie du faisceau est déviée vers le capteur, déclenchant la première mesure. Par la suite, le faisceau restant frappe une seconde cible, et l'écho de cette cible déclenche la seconde mesure. Ainsi pour un angle donné, le capteur peut recevoir jusqu'à huit valeurs de distance en prenant en compte le balayage vertical. Avec un capteur monocible, la réception du faisceau dévié arrêterait la mesure, et l'objet resterait caché. Concrètement, le capteur « IBEO LD-ML » attend la pulsation du second écho pour

générer la mesure d'un objet. À moins que trop de lumière soit absorbée par la distorsion, ceci permet au laser à balayage de voir à travers la pluie (Fuerstenberg *et al.*, 2002a,b).

Le capteur laser « IBEO LD-ML » a été développé pour supporter une large gamme d'application automobile, incluant le freinage d'urgence, le « *Stop and Go* », la reconnaissance de piéton et le « *Pre-Crash* » (Fuerstenberg *et al.*, 2002a).

Une application de ce type de capteur est donnée dans (Gidel *et al.*, 2008). Les auteurs de cet article présentent une méthode de détection, d'identification et de suivi de piétons à partir d'un unique capteur laser quatre plans. Pour cela, les points du balayage sont regroupés en différentes classes géométriques, dont le but est de filtrer le fond de l'image (murs, voiture, ...). Les auteurs utilisent ensuite une méthode exploitant la technique de fenêtrage de Parzen afin d'isoler les piétons. Enfin, ils utilisent un filtre à particules pour caractériser la trajectoire du piéton.

I.3.3.3 Capteur ALASCA (*Automotive Laser SCAnner*)

Le successeur du capteur « IBEO LD-ML » est l'ALASCA (*Automotive Laser SCAnner*). La version suivante est l'ALASCA XT. Ces deux capteurs IBEO comptent jusqu'à quatre réflexions par impulsion laser contrairement au capteur « IBEO LD-ML » qui n'en comptait que deux. Ces capteurs ont la capacité de déterminer si la mesure reçue provient d'un signal retourné par un réflecteur (type plaque d'immatriculation, ...) ou d'une autre surface à moindre pouvoir réfléchissant. En effet, la réflexion totale ou diffuse d'un objet crée des différences dans le rayonnement de l'impulsion de l'écho. Ces disparités se situent au niveau du récepteur du capteur et peuvent être mesurées grâce au rayonnement qui provient de la distance et de l'angle d'incidence de la cible. Cette principale évolution permet d'améliorer le pistage des objets et leur classification (Fuerstenberg *et al.*, 2004).

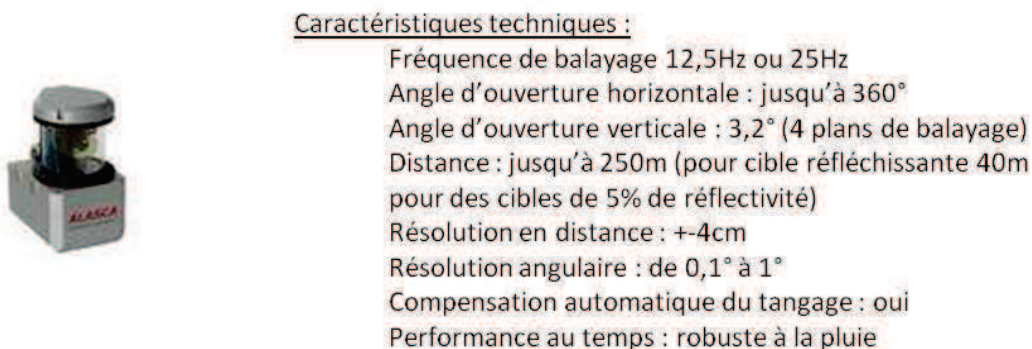


FIGURE I.21. Le capteur IBEO ALASCA XT (à gauche) et ses caractéristiques (à droite)

Ce capteur est notamment utilisé dans le projet INTERSAFE qui a débuté en 2004 et s'est terminé en 2007. Il portait sur la création d'une approche Européenne pour augmenter la sécurité aux intersections (Rössler et Fürstenberg, 2013). L'objectif de celui-ci était d'augmenter la sécurité et réduire (à long terme, éliminer) les collisions fatales aux intersections (Fuerstenberg, 2005a,b). La méthode est basée sur deux approches parallèles. La première utilise un capteur et les infrastructures de communication de véhicule. La communication est bidirectionnelle entre le véhicule et les infrastructures. La seconde utilise un simulateur de conduite qui analyse les situations potentiellement dangereuses.

I.4 Conclusion

De nos jours, la sécurité routière est de plus en plus importante. Limiter les risques d'accidents est un souci permanent tant chez les constructeurs automobiles que pour les nations. Aujourd'hui, les recherches sont notamment axées sur les systèmes d'aide à la conduite permettant à la fois de prédire une situation potentiellement dangereuse et d'aider le conducteur à en limiter les conséquences, voire les neutraliser.

Ce chapitre a notamment permis de mettre en évidence les contraintes qu'implique la mise en œuvre d'un système de détection et de suivi d'obstacle mobile sur route. Le type d'application mis en œuvre est fonction de l'environnement dans lequel nous nous trouvons, mais également du capteur choisi et des outils.

Nous avons vu que le choix d'un système monocapteur ou multicapteur est complexe et est fortement dépendant du résultat recherché. Dans notre cas, nous cherchons à mettre en place un système proche du temps réel. L'étude a montré que la multiplicité de capteur peut vite devenir lourde à gérer au niveau mémoire, temps de transfert des données et temps de traitement. Compte tenu de ces difficultés, nous choisissons de mettre en place un système monocapteur plus proche de nos besoins, limitant les coûts de traitement.

Une rapide étude des capteurs automobiles a permis de choisir le capteur lidar comme notre source de données. En effet, sa capacité à détecter tous types d'objets et ce sur une distance pouvant atteindre 120m pour un balayage de la scène pouvant atteindre 160° répond à nos besoins. De plus, ce capteur permet d'obtenir une grande quantité d'informations concernant la scène et notamment les distances capteur/objet sans nécessité de traitement. Cependant, une difficulté de ce type de mesure est leur nature spatialement distribué dans le cas monoplan. Dans le cas multiplan, la difficulté réside dans la disparition d'informations (due à une élévation du tir trop élevé) sur certain plan.

La section suivante est dédiée à l'état de l'art des techniques de détection et de suivi d'objets multiples à partir de données de type lidar.

Chapitre II

Problème de détection et suivi d'objets multiples

II.1 Architecture fonctionnelle classique du problème de détection et de suivi d'obstacles

Les mesures télémétriques d'un lidar décrivent les véhicules par un ensemble de points correspondant à la partie visible de leurs contours. Une difficulté de ce type de données est leur nature : l'ensemble de mesures décrivant un véhicule est spatialement corrélé. C'est pourquoi une étape préliminaire d'agrégation avant le processus de suivi est généralement nécessaire, comme l'illustre la figure II.1.

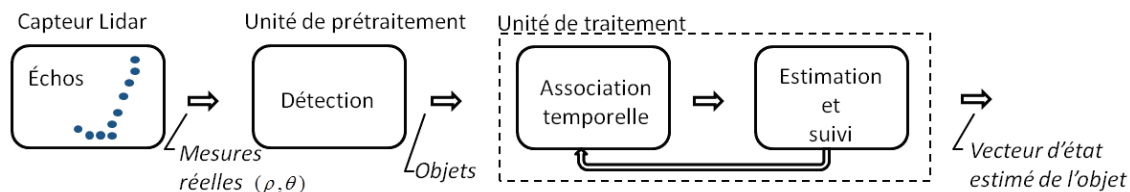


FIGURE II.1. Étapes classiques d'un algorithme de détection et de suivi

Ainsi, dans un premier temps, les mesures brutes, en sortie du capteur physique, sont utilisées pour détecter les objets présents dans le champ de vision du capteur. Un module de détection prend en compte des attributs caractéristiques du capteur et de son environnement pour extraire, des mesures du capteur, les paramètres d'objets d'intérêt (Nashashibi et Bargetin, 2008).

Dans un deuxième temps, une étape d'association temporelle associe les objets en sortie du module de détection à ceux en sortie du module de suivi en vue de mettre à jour le module d'estimation et de suivi. Le module d'association temporelle crée et maintient une liste des objets en cours de suivi. Ceci lui permet de mettre en relation les objets qui viennent d'être détectés avec ceux de sa liste. Il associe des objets détectés aux estimés d'objets présents dans la liste et en identifie d'autres comme des objets venant d'apparaître dans le champ de vision. Enfin, il élimine, de cette liste, les objets identifiés comme étant disparus du champ de vision du capteur. Plusieurs méthodes ont été proposées pour réaliser cette étape d'association. On peut citer, parmi les plus populaires, les approches de suivi à hypothèses multiples, *Multiple Hypothesis Tracking* (MHT), et celles d'association probabiliste conjointe de données ou *Joint Probabilistic Data Association* (JPDA). Les méthodes MHT envisagent, pour un horizon temporel donné, toutes les associations possibles entre les mesures et les pistes (Reid, 1979). JPDA modélise le problème d'association de manière probabiliste (Bar-Shalom et Fortmann, 1988; Bank, 2007).

Dans un troisième temps, une étape de filtrage rafraîchit l'estimation des paramètres des objets suivis à partir des informations d'associations de la liste d'objets, précédemment mises à jours. Elle initie celle des objets nouvellement détectés à partir des informations de création. Différents filtres peuvent être appliqués selon la nature de la modélisation du système. Si le système est décrit par un modèle linéaire ou qui a été rendu linéaire, un filtre de Kalman (Bank, 2007; Sato *et al.*, 2010) ou un filtre de Kalman étendu (Tango *et al.*, 2008) sera choisi. S'il est décrit par un modèle non-linéaire, un filtre de Kalman sans parfum (Julier et Uhlmann, 2004; Richter *et al.*, 2008), une approche à base de grille (Homm *et al.*, 2010) ou encore une approche de Monte Carlo séquentielle sera utilisée. Les approches de Monte Carlo séquentielles, également connues sous le nom de filtrage particulaire (Doucet *et al.*, 2000a), ont également été plus récemment mises en œuvre dans des algorithmes de suivi pour des applications routières (Idler *et al.*, 2006; Thuy et Leon, 2009).

Lorsque le système de détection et de suivi comporte plusieurs capteurs de détection d'objets, une étape de fusion peut être ajoutée entre les capteurs physiques et le module de détection (fusion signal), après les modules de détection des capteurs physiques (fusion d'objets), ou encore après les modules d'estimation et de suivi des capteurs logiques (fusion de pistes) (Hall, 1992; Liggins *et al.*, 1997; Herpel *et al.*, 2008; Nashashibi et Bargetin, 2008; Baig *et al.*, 2011).

II.2 Détection dans les données télémétriques

Les lidars automobiles délivrent un ensemble de mesures (ρ, θ) à chaque nouvelle rotation de leur laser (balayage capteur t). Ces mesures sont séparées d'une distance angulaire fixe $\Delta\theta$ (le pas de tir). Le capteur télémétrique calcule chaque distance ρ_θ (distance délivrée à l'angle θ) en divisant la vitesse de la lumière par la mesure du temps qui s'est écoulée entre la date d'émission du signal laser et celle de la réception du signal réfléchi correspondant (temps de propagation). Chaque ensemble de mesures délivré est un agglomérat d'échos pouvant être représentatif de différents objets détectés, de pertes de signal et de non détection.

Détecter un objet parmi ces mesures nécessitent donc de les regrouper en une information objet. Pour cela, généralement on distingue trois étapes (Jida, 2008) :

- le partitionnement des données d'un balayage en groupes de mesures (*clustering*) ;
- la segmentation de chacune des partitions trouvées, afin d'en extraire des contours d'objets ;
- l'identification et la classification des objets issus de l'étape de segmentation.

Nous considérons, dans la suite de ce document, que chaque tour de rotation du lidar correspond à notre fréquence d'échantillonnage, Δt , d'une image capteur. t est l'indice ou numéro du balayage (tour de rotation), par conséquent, $t.\Delta t$ est la date du balayage. Nous utilisons, aussi, les expressions des mesures d'un même échantillon suivantes :

- Z_t l'ensemble des mesures d'un balayage t ;
- $Z_t = \{z_\theta; \theta \in \{\theta_{min}, \theta_{min} + \Delta\theta, \dots, \theta_{max}\}\}$;
- \mathcal{P}_t l'ensemble des partitions de Z_t ;
- N_{max} le nombre maximal d'objets d'une image à t d'au moins un point
- $\mathcal{P}_t = \left\{ \mathcal{P}(n); n \in [0, N_{max}], N_{max} \leq \frac{\theta_{max} - \theta_{min}}{\Delta\theta} + 1, \mathcal{P}(0) = \emptyset \right\}$;
- $\mathcal{P}(n) = \{z_\theta; \theta \in \{\theta_a, \theta_a + \Delta\theta, \dots, \theta_{a+f}\}\}$, f le nombre de points de $\mathcal{P}(n)$ et $\mathcal{P}(1) \cap \dots \cap \mathcal{P}(n) = \emptyset$;
- $z_\theta = (\rho_\theta, \theta)$ lorsque la mesure est exprimée en coordonnées polaires ;
- $z_\theta = (x_\theta, y_\theta)$ lorsqu'elle l'est en coordonnées cartésiennes ;
- z_θ^- indique la mesure de l'impact à l'angle de tir précédent ($\theta^- = \theta - \Delta\theta$) ;
- z_θ^+ indique celle de l'impact à l'angle de tir suivant ($\theta^+ = \theta + \Delta\theta$) ;
- $\Delta\theta$ est la résolution angulaire du lidar, c'est-à-dire le pas de tir durant le balayage.

D'autre part, le passage du repère polaire au repère cartésien se fait classiquement selon le système

d'équations suivant :

$$\begin{cases} x_\theta = \rho_\theta \cos \theta \\ y_\theta = \rho_\theta \sin \theta \end{cases} \quad (\text{II.1})$$

II.2.1 Partitionnement de données (*clustering*)

L'étape de partitionnement de données regroupe, au sein de partitions, les mesures qui partagent un caractère commun. Dans le cas d'un échantillon de mesures lidar, l'étape de partitionnement consiste à regrouper les mesures correspondant à un même objet. Les algorithmes de partitionnement reposent alors sur la comparaison d'un critère de distance avec un critère seuil. Jida (2008) part du constat que les données lidar sont naturellement ordonnées, lors du balayage de la scène, pour éliminer un certain nombre d'algorithmes tels que le *K-Means* basés sur le calcul de centroïdes selon un processus itératif. Il s'intéresse alors aux approches de partitionnement reposant sur la comparaison de critère de distance entre deux mesures successives. Nous nous plaçons dans cette continuité et reprenons ci-après la description de la méthode générale de l'algorithme de partitionnement, puis nous nous intéressons aux déclinaisons adaptatives de Dietmayer *et al.* (2001); Santos *et al.* (2003); Borges et Aldon (2004).

II.2.1.1 Algorithme de partitionnement

L'algorithme de partitionnement consiste en la comparaison, mesure après mesure d'un balayage, de la distance $d(z_{\theta-}, z_\theta)$ entre deux mesures successives à une valeur de distance seuil. Si cette distance $d(z_{\theta-}, z_\theta)$ est inférieure à la valeur seuil, alors la mesure z_θ au pas de tir courant est affectée à la même partition que la précédente $z_{\theta-}$. Si ce n'est pas le cas, elle est affectée à un nouveau groupe. La distance la plus utilisée est la distance euclidienne définie comme suit :

- $d(z_{\theta-}, z_\theta) = \sqrt{\rho_{\theta-}^2 + \rho_\theta^2 - 2(\rho_{\theta-})(\rho_\theta) \cos(\Delta\theta)}$ lorsqu'elle est déclinée en coordonnées polaires ;
- $d(z_{\theta-}, z_\theta) = \sqrt{(x_{\theta-} - x_\theta)^2 + (y_{\theta-} - y_\theta)^2}$ lorsqu'elle l'est en coordonnées cartésiennes.

La détermination de la valeur du seuil est la principale difficulté des algorithmes de partitionnement. Le seuil peut être fixe (algorithme non adaptatif) ou calculé à chaque itération (algorithme adaptatif). Une valeur de seuil fixe ne tient cependant pas compte des grandes différences de distances qui peuvent être calculées entre deux points d'un même objet selon l'orientation et la distance au lidar de cet objet. Ceci est illustré en figure II.2. Ainsi, on préférera l'utilisation d'une distance seuil adaptative, qui est calculée à chaque comparaison entre deux mesures $z_{\theta-}$ et z_θ consécutives. Dietmayer *et al.* (2001); Santos *et al.* (2003) et Borges et Aldon (2004) proposent différents seuils adaptatifs que nous présentons maintenant.

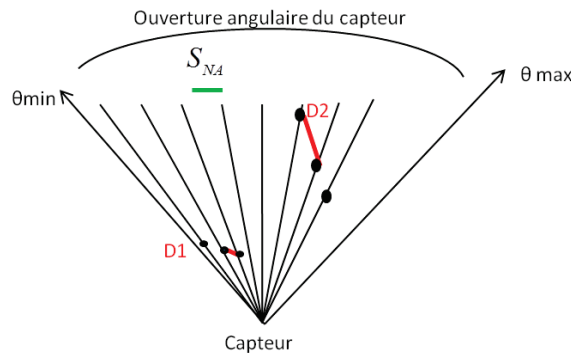


FIGURE II.2. Seuil S_{NA} fixe d'un algorithme non adaptatif

II.2.1.2 Seuil adaptatif de Dietmayer *et al.* (2001)

Dietmayer *et al.* (2001) définissent une distance seuil qui ne dépend que de la distance entre la surface d'impact et le repère du télémètre. Sa formulation est la suivante :

$$S_{Dietmayer} = C_{Dietmayer} \min\{\rho_{\theta^-}, \rho_{\theta}\} + C_{bruit}$$

avec

- $C_{Dietmayer} = \sqrt{2(1 - \cos(\Delta\theta))}$ le paramètre de raffinement du partitionnement ;
- C_{bruit} une constante traduisant l'influence du bruit en distance des impacts sur le calcul du seuil, classiquement $C_{bruit} = 3\sigma_{\rho}$ où σ_{ρ} est une donnée constructeur de l'ordre de quelques centimètres correspondant à l'erreur expérimentale sur la mesure de la distance ρ (Jida, 2008).

Comme le remarque Jida (2008), le calcul du paramètre $C_{Dietmayer}$ prend pour hypothèse que la surface est perpendiculaire à la bissectrice de la droite passant par les points z_{θ^-} à z_{θ} . Ainsi, ni l'orientation ni l'inclinaison de la surface d'impact ne sont considérées et le calcul du seuil est donc très fortement dépendant de la distance du capteur à la surface d'impact. Il devient alors difficile de séparer les groupes de mesures lorsque les impacts sont très éloignés du capteur.

II.2.1.3 Seuil adaptatif de Santos *et al.* (2003)

L'approche de Santos *et al.* (2003) reprend celle de Dietmayer *et al.* (2001). Mais, ils y introduisent une valeur d'angle constante, β , permettant de prendre en compte l'inclinaison maximale de la surface (figure II.3). β réduit la dépendance du partitionnement, précédemment constatée dans l'approche de Dietmayer *et al.* (2001), à la distance entre l'objet et le lidar. La valeur de β est à choisir expérimentalement. Si β est choisi trop grand, il sera plus difficile de séparer deux objets proches, à l'inverse s'il est choisi trop petit, deux mesures successives provenant d'un même objet risquent d'être séparées (Santos *et al.*, 2003).

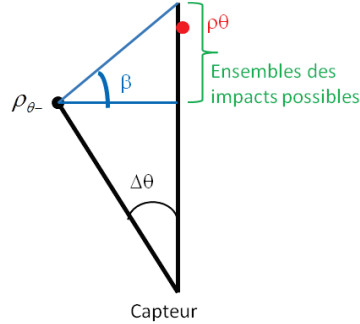


FIGURE II.3. Paramètres du seuil adaptatif S_{Santos} selon Santos *et al.* (2003)

La distance seuil devient alors :

$$S_{Santos} = \frac{C_{Dietmayer}}{\cot(\beta) \cos(\Delta\theta/2) - \sin(\Delta\theta/2)} \min\{\rho_{\theta^-}, \rho_{\theta}\} + C_{bruit}$$

avec β angle d'inclinaison maximal de la surface d'impact, fixé expérimentalement, généralement autour de 60° (Jida, 2008).

II.2.1.4 Seuil adaptatif de Borges et Aldon (2004)

Borges et Aldon (2004) prennent en compte, comme Santos *et al.* (2003), l'orientation maximale de la surface entre les impacts et le capteur télémétrique pour son calcul de la distance seuil. Cependant, ils n'utilisent pas la mesure de distance z_{θ} de l'angle de tir courant, mais seulement celle $z_{\theta-}$ de l'angle de tir précédent. Ils font en outre intervenir l'angle λ non orienté mais construit, dans le sens trigonométrique, au niveau du point d'impact sur la surface et à partir de « la droite de tir de l'impact $z_{\theta-}$ » (figure II.4). Cet angle λ joue, en quelque sorte, le rôle « d'angle limite à la surface en deçà duquel le capteur télémétrique ne pourra délivrer de mesures », car, en deçà, la surface et le faisceau se rapproche du parallélisme et l'angle de réflexion du signal empêche alors le retour de l'écho vers le capteur. L'angle limite de tir à la surface est fixée par les constructeurs de lidar aux alentours de 10° . L'angle λ peut être légèrement supérieur à cette valeur, jusqu'à 25° selon Jida (2008), mais un λ trop grand peut rendre impossible la séparation en partition. Ce paramètre est donc à régler expérimentalement.

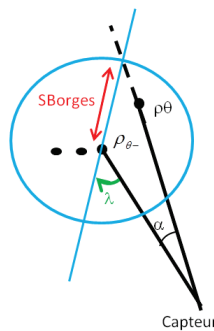


FIGURE II.4. Paramètres du seuil adaptatif S_{Borges} selon Borges et Aldon (2004)

La formulation de la valeur de la distance seuil selon l'approche de Borges et Aldon (2004) est la suivante :

$$S_{Borges} = (\rho_{\theta-}) \frac{\sin(\Delta\theta)}{\sin(\lambda - \Delta\theta)} + C_{bruit}$$

avec λ angle limite en deçà duquel le lidar ne délivre plus de mesure (perte du signal).

II.2.2 Segmentation de partitions

Les algorithmes de partitionnement de mesures ont eu pour objectif de regrouper les mesures partageant une ou plusieurs caractéristiques communes. Ceux de segmentation ont pour finalité de retrouver des formes dans ces ensembles de mesures. Afin d'identifier des contours d'objets, nous nous intéressons aux algorithmes d'extraction de ligne dans des données laser 2D et nous nous appuyons en particulier sur les travaux de Nguyen *et al.* (2005, 2007). Nous présentons ainsi ci-après les cinq méthodes de segmentation répertoriées : l'approche incrémentale, l'approche par décomposition et fusion (*Split and merge*), l'approche RANSAC (*Random Sample Consensus*), l'approche par transformée de Hough et, enfin, l'approche par maximisation de l'espérance (*EM, Expectation-Maximization*).

II.2.2.1 Approche incrémentale

Les approches incrémentales de segmentation sont apparues dans les années 70. Elles consistent en la recherche de forme, dans un ensemble ordonné \mathcal{P} de mesures. À partir des premiers points constituant la forme, elles vérifient pour chaque nouvelle mesure son appartenance à la forme en cours de construction selon un critère de décision. Le choix de ces critères repose sur des conditions simplifiées. Un exemple de critère de décision dans le cas de l'extraction d'une droite est le respect d'une valeur seuil maximale par la distance orthogonale entre la mesure considérée et la droite. Cette approche est fondée sur le calcul successif des paramètres de la droite en ajoutant le point z_{θ^+} .

1. Soit un ensemble \mathcal{P} de N points $\mathcal{P} = \{z_{\theta_a}, \dots, z_{\theta_{a+N}}\}$
2. Construire une droite (d) entre les deux premiers points
3. Ajouter le point suivant z_{θ^+} au modèle de droite courant (d) et réestimer les paramètres de la droite (d)
4. Si les conditions de la droite sont satisfaites, retour à l'étape 3,
5. Sinon
 - Éliminer le dernier point z_{θ^+} de la droite (d)
 - Réestimer les paramètres de la droite (d)
 - Mémoriser la droite (d)
 - Retour à l'étape 2 avec comme modèle de droite (d) ce dernier point z_{θ^+} et le suivant.

L'algorithme incrémental de segmentation est simple à mettre en œuvre. Les critères de décision sont facilement modifiables et il ne nécessite pas que l'acquisition des données soit terminée pour être exécuté. Cependant, avoir un seuil fixe est un inconvénient. Le choisir trop petit implique un morcellement des droites détectées à une distance très éloignée de la position du capteur. À l'inverse, un seuil trop grand conduit à des associations aberrantes de points pour des droites détectées à une distance proche du capteur.

II.2.2.2 Approche *Split and Merge*

L'approche *Split and Merge* est probablement la plus populaire des méthodes d'extraction de droites. C'est un algorithme itératif qui provient du traitement d'images, il est ainsi capable de traiter les points d'échantillonnage d'un balayage laser qui sont peu nombreux par rapport aux pixels d'une image. Il est plus rapide qu'un algorithme de groupement (Einsele, 1997). Dans la littérature, on retrouve cette technique de segmentation dans beaucoup de travaux, notamment dans la robotique, sur des données télémétriques de laser 2D (Einsele, 1997; Borges et Aldon, 2004; Siadat *et al.*, 1997; Zhang et Ghosh, 2000).

Les algorithmes *Split and Merge* permettent de réaliser des approximations polygonales de lignes par une succession de division de ces lignes en deux (*Split*), puis par une phase de fusion (*Merge*), selon un ou plusieurs critères de validation. Selon Ray et Ray (1995), le problème fondamental de ces techniques est la segmentation initiale.

Par exemple, la technique *Split and Merge* de Pavlidis et Horowitz (1974) (cité dans (Ray et Ray, 1995)) cherche à adapter à un contour des lignes droites dont la définition provient d'une première segmentation. Elle calcule pour cela l'erreur quadratique des points du contour à la droite. Si l'erreur est trop grande, la ligne est divisée ; deux lignes seront fusionnées si elle est trop petite.

Dans (Ray et Ray, 1995), des polygones insensibles à l'orientation et à l'échelle de courbes chaînées sont générés en utilisant trois critères de validation. Le premier est la distance (non perpendiculaire) d'un point k d'une courbe à un segment de droite ij divisée par la longueur du segment l_{ij} afin de la rendre indépendante à l'échelle (normalisation), critère d_{kij}/l_{ij} . Les segments de courbes sont séparés aux points k pour lesquels d_{kij}/l_{ij} atteint un maximum. Le second, le ratio de la longueur de l'arc à la longueur du segment de ligne

(a_{ij}/l_{ij}) , comparé à une valeur seuil, permet de vérifier que les points k de la courbes compris entre i et j sont colinéaires. En cas de dépassement du seuil, un troisième critère, la somme des d_{kij}/l_{ij} pour les points k de la courbe variant entre $(i + 1)$ et $(j - 1)$, sert à trancher la décision de colinéarité ou non colinéarité. En fonction de la valeur de ces trois critères, des lignes sont tout d'abord divisées (phase *Split*). Des lignes seront par la suite fusionnées selon ces critères (phase *Merge*).

La phase de division peut utiliser la méthode IEPF (*Iterative-End-Point-Fit*) (Borges et Aldon, 2000). Celle-ci permet, si le critère de validation est satisfait, la division des points d'un ensemble ordonné \mathcal{P} en deux sous-ensembles à partir du point de rupture de l'ensemble. Le point candidat au titre de point de rupture de l'ensemble \mathcal{P} est le point pour lequel la distance à la ligne, construite à partir des deux points extrêmes de \mathcal{P} , est maximale (par exemple distance orthogonale à la droite, distance de Mahalanobis...). Si en ce point, la valeur de la distance est supérieure à une distance seuil, l'ensemble est divisé en deux sous-ensembles en ce point de rupture. De façon récursive, ces sous-ensembles vont eux-même être divisés tant que le critère de validation est satisfait.

Borges et Aldon (2000) présentent un algorithme SMF (*Split and Merge Fuzzy*) dans une liste de droites, selon un schéma proche de la méthode IEPF. Par contre, ils utilisent un critère de validation basé sur la dispersion des points supports (i.e. points à l'origine de la définition des droites), aussi bien au moment de la phase de division que celle de fusion. Les droites de la liste sont, chacune, définies par leurs paramètres en coordonnées polaires, leur centre de gravité en coordonnées cartésiennes et un vecteur comportant les indices des points supports. Une droite sera séparée en deux droites lorsque la valeur de dispersion maximale trouvée en un point support est supérieure à une valeur de dispersion maximale donnée. Lors de la phase de fusion, pour toute ligne de la liste, on cherche à savoir si deux lignes sont candidates à la fusion. Ce sont les deux lignes les plus proches en termes de distance entre leur centre de gravité et celui de la ligne initiale. Parmi les deux possibilités, la ligne fusionnée résultante sera celle dont la dispersion est la plus petite, si toutefois elle satisfait le critère de dispersion.

Un algorithme *Split and Merge* basé sur une technique de division IEPF peut être schématisé ainsi :

1. Soit un ensemble \mathcal{P} de N points, $\mathcal{P} = \{z_{\theta_a}, \dots, z_{\theta_{a+N}}\}$
2. Placer \mathcal{P} dans la liste L
3. Créer une droite (d), selon la méthode choisie, à partir des données de \mathcal{P}
4. Chercher le point z_{θ_R} pour lequel la distance choisie à la droite $d(z_{\theta_R}, (d))$ est maximale
5. Si $d(z_{\theta_R}, (d))$ est inférieure à un seuil, retour à l'étape 3
6. Sinon
 - Diviser l'ensemble \mathcal{P} au niveau de z_{θ_R} en \mathcal{P}_a et \mathcal{P}_b
 - Remplacer \mathcal{P} par \mathcal{P}_a et \mathcal{P}_b dans L
 - Retourner à l'étape 3
7. quand tous les sous-ensembles de L ont été étudiés, fusionner les segments colinéaires selon le ou les critères de validation choisis.

L'approche *Split and Merge* est simple à mettre en œuvre. Cependant, cet algorithme récursif ne peut être exécuté que sur un ensemble de point déterminé. Ceci implique que le traitement ne peut commencer qu'après la fin de la phase d'acquisition des mesures. Comme pour l'algorithme incrémental, le choix du seuil est une difficulté. L'article de (Fortin *et al.*, 2012) montre d'ailleurs cette difficulté et propose une nouvelle méthode qui définit un critère de détection de segments de lignes dépendant uniquement des paramètres intrinsèques du capteur.

II.2.2.3 Approche RANSAC (*Random Sample Consensus*)

L'algorithme RANSAC, publié pour la première fois par Fischler et Bolles (1981), est une méthode itérative destinée à estimer les paramètres d'un modèle mathématique à partir d'un ensemble de données observées contenant un grand nombre de valeurs aberrantes (appelées *outliers*).

L'algorithme RANSAC repose sur deux hypothèses fortes :

- les *inliers* sont les données dont la distribution peut être expliquée par un ensemble de paramètres du modèle considéré ;
- les *outliers* sont des données qui ne correspondent pas au modèle choisi.

RANSAC suppose qu'étant donné un ensemble petit d'*inliers*, il existe une procédure qui permet d'estimer les paramètres du modèle de façon à expliquer de manière optimale ces données. L'algorithme cherche donc à estimer le modèle par ses *inliers* en éliminant ses *outliers*, comme le montre la figure II.5.

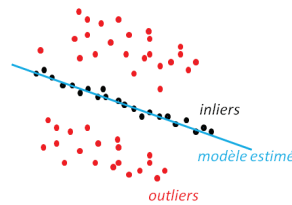


FIGURE II.5. Exemple de dispersion des *inliers* et *outliers*

À l'inverse des techniques conventionnelles qui, à partir d'un grand ensemble de données \mathcal{P} , éliminent les points aberrants, l'algorithme de Ransac utilise le plus petit ensemble de données initiales possible et l'accroît en associant les données les plus compatibles avec la modélisation antérieurement réalisée à partir de l'ensemble des données initiales (Fischler et Bolles, 1981).

L'algorithme consiste en deux étapes répétées de manière itérative :

- un ensemble minimal de n points est sélectionné de manière aléatoire dans l'ensemble des données d'entrée \mathcal{P} , à partir duquel les paramètres du modèle sont estimés ;
- chacun des éléments de l'ensemble de données \mathcal{P} est testé pour vérifier s'il correspond au modèle.

L'algorithme de Ransac s'arrête lorsque la probabilité de trouver un meilleur sous-ensemble est inférieure à un seuil (Zuliani, 2012).

Nous considérons maintenant l'exemple de Jida (2008) qui cherche à estimer une droite parmi un ensemble de points (figure II.5). Deux points ($z_{\theta a}, z_{\theta b}$) sont choisis aléatoirement parmi l'ensemble des données à traiter. Les paramètres de la droite (d) passant par ces points sont estimés. La distance de chacun des autres points $z_{\theta j}$ à la droite (d) est alors calculée. Un point sera associé à (d) si sa distance à cette droite est inférieure à un seuil fixé. Ces points sont appelés *inliers* candidats. Si le nombre des *inliers* candidats est suffisant, alors les paramètres de la droite sont réestimés en les prenant en compte. Ces *inliers* candidats sont ensuite éliminés du groupe de points initial. On réitère la procédure jusqu'à ce qu'il n'y ait plus suffisamment de points dans le groupe ou jusqu'au nombre maximum d'itération. L'algorithme est résumé ci-dessous :

- Soit un ensemble \mathcal{P} de N points $\mathcal{P} = \{z_{\theta_a}, \dots, z_{\theta_{a+N}}\}$
- Répéter :
 - Choisir aléatoirement deux points $(z_{\theta_a}, z_{\theta_b})$
 - Créer une droite (d) entre ces deux points et estimer ses paramètres
 - Calculer les distances des autres points à la droite $d(z_{\theta_j}, (d)) \forall z_{\theta_j} \in \mathcal{P} - \{z_{\theta_a}, z_{\theta_b}\}$
 - Construire l'ensemble des *inliers* candidats de distance inférieure à un seuil
 - S'il y a un nombre suffisant d'*inliers* candidats
 - Réestimer les paramètres de la droite (d) en prenant en compte les *inliers* candidats
 - Mémoriser la droite (d)
 - Éliminer les *inliers* candidats de l'ensemble \mathcal{P}
- Tant que le maximum de N itérations n'est pas atteint ou que le nombre de points de \mathcal{P} soit insuffisant.

La méthode RANSAC est populaire en vision par ordinateur. Simple à mettre en œuvre, elle oblige cependant d'attendre la fin de l'acquisition pour démarrer les traitements. Elle permet d'estimer les paramètres avec un degré élevé de précision, même si une quantité importante d'*outliers* est présente. Dans notre cas, un des problèmes est le caractère aléatoire de la recherche des droites, on peut obtenir des droites qui n'ont pas d'existence réel en termes d'obstacles présents dans le champ de vision du capteur. Cet algorithme nécessite également de fixer un seuil spécifique au problème traité. Enfin, l'approche RANSAC est une approche à modèle unique. C'est-à-dire lorsque plusieurs modèles coexistent, l'algorithme n'arrive pas à déterminer les différents modèles (Forsyth et Ponce, 2003).

II.2.2.4 Transformée de Hough

Selon (Duquenoy et Taleb-Ahmed, 2006), la transformée de Hough permet la détection, dans une image, de formes décrites analytiquement, formes incluant des lignes droites, cercles et ellipses. Elle permet aussi sous certaines conditions toute détection de forme quelle qu'en soit leur description. Elle est utilisée dans de nombreuses applications telles que la détection d'objets, la détection de mouvement, la reconnaissance de caractère... Duquenoy et Taleb-Ahmed la décrivent ainsi. La transformée de Hough cherche à déterminer les paramètres géométriques d'une forme au moyen d'une procédure de vote : chaque point de l'image, de la forme, vote pour un ou plusieurs points dans l'espace des paramètres. La dimension de cet espace dépend de la forme recherchée (deux dimensions pour une droite, trois pour un cercle). Plusieurs approches de vote existent. Nous présentons ci-après l'approche décrite dans (Duda et Hart, 1971, 1972), approche qualifiée de transformation *one-to-many* par Duquenoy et Taleb-Ahmed.

Duda et Hart (1971) proposent de détecter des lignes (ou des courbes) dans des images en travaillant dans l'espace des paramètres (Θ, ρ) . Dans cet espace, l'équation d'une droite est : $\rho = x \cdot \cos(\Theta) + y \cdot \sin(\Theta)$. En restreignant Θ à l'intervalle $[0, \pi]$, toute droite (d) dans le plan xy correspond à un unique point dans le plan $\Theta\rho$ pour lequel l'angle Θ est l'angle polaire de la normale à la droite (d) (figure II.6). Considérant l'ensemble de points $\{(x_1, y_1), \dots, (x_n, y_n)\}$, ils y cherchent des lignes droites en transformant chaque point (x_j, y_j) en une courbe sinusoidale dans le plan $\Theta\rho$, courbe définie par l'expression (II.2).

$$\rho = x_j \cdot \cos(\Theta) + y_j \cdot \sin(\Theta) \quad (\text{II.2})$$

Les courbes correspondant à des points colinéaires ont un point d'intersection commun dans le plan $\Theta\rho$, qui définit la ligne passant par les points colinéaires. Duda et Hart en déduisent que le problème de la détection de points colinéaires est celui de la recherche des courbes (II.2) concurrentes.

Les intervalles de recherche en ρ et en Θ doivent être quantifiés. Des indices correspondant aux valeurs de ρ et de Θ définissent les lignes et colonnes d'un tableau, appelé accumulateur. Pour chaque point (x_j, y_j) de l'image, la courbe correspondante, décrite par l'équation (II.2), est enregistrée dans l'accumulateur en incrémentant le compteur de points des cellules (ind_Θ, ind_ρ) traversées par la courbe. Ainsi, une recherche des cellules dont la valeur du compteur est supérieure à un seuil permet de trouver les paramètres (Θ_j, ρ_j)

des normales aux droites par lesquelles passent le plus grand nombre de points, à l'erreur de quantification ($\Delta\rho$ et $\Delta\Theta$) près.

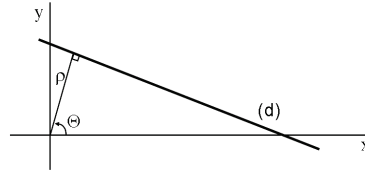


FIGURE II.6. Paramètres (Θ, ρ) tels que Θ est l'angle polaire de la normale à la droite (d) (Duda et Hart, 1971)

L'algorithme est le suivant :

1. Soit un ensemble \mathcal{P} de N points, $\mathcal{P} = \{z_1, \dots, z_n\}$ où $z_j = (x_j, y_j), \forall j \in [1, n]$;
2. Soit le problème ainsi quantifié : $\forall z_j, \rho \in [\rho_{min}, \rho_{max}]$, ρ est arrondi à $\Delta\rho$ près, $\Theta \in [0, \pi]$ avec Θ incrémenté de $\Delta\Theta$
3. Soit $Acc [j_\rho, j_\Theta]$ un tableau de N_ρ lignes et de N_Θ colonnes, dimensions définies conformément au choix de quantification du problème.
4. Début
 - Pour tout $z_j, j \in [1, n]$,
 - Pour tout $\Theta \in [0, \pi]$ incrémenté de $\Delta\Theta$
 - Calculer ρ selon II.2
 - $ind_\rho = f(\rho)$, où $f(\rho)$ respecte les règles de quantification adoptées
 - $Acc [ind_\rho, ind_\Theta] + = 1$
 - Fin pour
 - Fin pour
 - $\forall (ind_\rho, ind_\Theta)$, rechercher les valeurs max de $Acc [ind_\rho, ind_\Theta]$ supérieures à un seuil.
5. Fin

Dans le cas de données issues d'un télémètre le principe est le même. Ainsi Pfister *et al.* (2003) cherchent à trouver des droites à partir de chacun des points de l'ensemble \mathcal{P} étudié. Chaque point est transformé en une ligne discrétisée dans l'espace de Hough. Cette transformation est basée sur la paramétrisation d'une ligne en coordonnées polaires avec une distance normale à l'origine ρ et un angle Θ (figure II.6). De même que pour l'image, pour chaque point de l'ensemble, les paramètres ρ et Θ de chaque ligne sont calculés et enregistrés dans un accumulateur. Quand une cellule (ind_ρ, ind_Θ) de l'espace de Hough est incrémentée, la coordonnée du point est associée et sauvegardée. Ainsi, quand un pic est déterminé, les points associés sont extraits. Dans ce cas, les auteurs peuvent trier les points du balayage en sous-ensembles colinéaires et déterminer l'orientation du segment de ligne.

La transformée de Hough est une approche récursive, obligeant ainsi d'attendre la fin de l'acquisition pour démarrer les traitements. Elle est sensible à la quantification choisie. Un échantillonnage trop fin, en ρ et Θ augmente le temps de calcul. D'autre part, elle ne regarde pas la continuité des points colinéaires. Par conséquent, la position d'une droite peut être perturbée par la présence de points se trouvant dans une autre partie de l'image considérée, et donc sans relation avec la forme recherchée.

II.2.2.5 Approche de maximisation de l'espérance (EM, *Expectation-Maximization*)

C'est une approche itérative, formalisée par Dempster *et al.* (1977). Elle est utilisée en classification de données ou pour l'apprentissage automatique. Elle sert également à l'extraction de droite en vision par ordinateur (Forsyth et Ponce, 2003) et en robotique (Pfister *et al.*, 2003). Elle repose sur la recherche du maximum de vraisemblance ou du maximum *a posteriori* pour estimer les paramètres de modèles statistiques, pour lesquels le modèle dépend de variables non observées ou aberrantes (*outliers*). À chaque itération, on calcule un nouveau modèle qui améliore la log-vraisemblance des données du modèle précédent, jusqu'à convergence. Chaque nouveau modèle est obtenu par l'exécution successive de deux étapes (*Expectation* (E) et *Maximization* (M)) :

- l'évaluation de l'espérance (E) se fait par calcul de l'espérance de la vraisemblance en tenant compte des dernières variables observées ;
- pour l'étape de maximisation (M), le maximum de vraisemblance des paramètres est estimé en maximisant la vraisemblance trouvée à l'étape E.

Les paramètres trouvés en M deviennent le point de départ d'une nouvelle phase d'évaluation de l'espérance, et on itère ainsi.

Par exemple, Liu *et al.* (2001) utilisent l'algorithme de maximisation de l'espérance pour ajuster un modèle, de type plan de faible complexité, à partir de données 3D collectées par des télémètres et une caméra panoramique fixés sur un robot mobile. L'algorithme est le suivant, à partir d'un modèle de type plan $\Theta^{[0]}$, une séquence de modèle $\Theta^{[1]}$, $\Theta^{[2]}$... est générée par l'exécution des deux étapes E et M. À l'étape E, l'espérance des correspondances inconnues $E[c_{ij}]$ et $E[c_{i*}]$ est estimée pour le n^e modèle $\Theta^{[n]}$, puis à l'étape M, un nouveau modèle $\Theta^{[n+1]}$ de vrai maximum est calculé. Comme autre exemple, Dempster *et al.* (1977) utilisent l'algorithme EM pour le calcul du maximum de vraisemblance à partir de données incomplètes. Dans cet article, les données complètes $f(..|..)$ sont liées aux données incomplètes $g(..|..)$ par $g(y|\phi) = \int_{x(y)} f(x|\phi) dx$ où y est la mesure observée, et les auteurs cherchent directement la valeur de ϕ qui maximise $g(y|\phi)$ via l'algorithme EM.

Un résumé de l'algorithme EM est le suivant :

1. Soit un ensemble \mathcal{P} de N points $\mathcal{P} = \{z_{\theta_a}, \dots, z_{\theta_{a+N}}\}$
2. Répéter :
 - Générer les paramètres d'une droite aléatoire
 - Initialiser le poids des points restants
 - Répéter
 - Calculer le poids des points du modèle de droite (étape *Expectation*)
 - Réestimer les paramètres du modèle de droite (étape *Maximization*)
 - Tant que le maximum de N ensembles n'est pas atteint.
3. Tant que N essais ne sont pas atteints ou que la droite n'est pas trouvée.
4. Si droite trouvée
 - Stocker la droite
 - Éliminer les *inliers*
 - Retourner en 2
5. Sinon terminer

La difficulté de la méthode EM réside dans le choix d'une bonne valeur initiale.

II.2.3 Identification et classification d'objets

Différents types d'objets se côtoient dans un environnement routier. On peut citer les piétons, voitures, camions, bicyclettes, motocyclettes, arbres, barrières, etc. Ces différents objets ne sont pas vus de la même manière par un même capteur. Typiquement leur gabarit diffère, mais pas seulement, leur dynamique aussi. Si dans le cadre de notre étude, nous ne nous sommes intéressés qu'aux véhicules et n'avons donc pas cherché à identifier les objets détectés, dans un cas de cibles multiples hétérogènes, c'est une étape qui serait indispensable afin d'utiliser le bon modèle d'objet pour l'estimation de son centre de gravité, mais aussi pour la détermination des paramètres d'initialisation des filtres associés. Il est à noter que les paramètres d'estimation de la dynamique des objets en sortie du module de suivi seront utiles afin d'affiner l'identification des objets détectés, quitte à ensuite supprimer un filtre et en initialiser un autre pour un objet qui, à l'origine, aurait été mal classé compte tenu de l'absence d'information sur sa dynamique.

Le problème d'identification et de classification n'étant pas l'objet de cette thèse, nous ne nous y attarderons pas plus. Cependant, le lecteur intéressé trouvera de nombreuses méthodes dans la littérature.

II.3 Modélisation des objets

La modélisation consiste à créer des modèles (un modèle étant un objet) qui seront reproduits par imitation. Cette modélisation, plus ou moins fidèle à l'objet d'origine, dépend du type (radar, lidar, vidéo... humain) et de la qualité du perceuteur (champs de vision, cf. ouverture, profondeur, précision...) ainsi que des objectifs applicatifs du concepteur (dénombrement d'obstacles, localisation précise... art figuratif). La plupart du temps, elle n'est qu'une représentation abstraite de l'objet perçu, seules certaines de ses caractéristiques étant retenues. Ainsi, l'étude de la littérature sur la détection et le suivi d'obstacles montre différents types de modèles selon la nature du capteur de perception, les conditions de luminosité (nuit, jour, brouillard, pluie...), la nature de l'objet suivi (véhicule, piéton...).

Prenons comme exemple, les articles (O'Malley *et al.*, 2011; Idler *et al.*, 2006), qui concernent le problème de la détection et du suivi de nuit des véhicules sur des séquences vidéo. O'Malley *et al.* (2011) utilisent une caméra positionnée à l'arrière du véhicule afin de détecter les phares avant des véhicules qui le suivent. Une configuration fixée à une faible exposition permet de garder la perception des phares, cibles parmi les objets les plus brillants. Idler *et al.* (2006) utilisent une caméra infrarouge située à l'avant du véhicule afin de détecter les phares arrières des véhicules qui le précèdent. Ces auteurs ont utilisé deux modèles d'objet, le premier, le cercle est utilisé pour la détection, le second, le rectangle est utilisé comme modèle pour suivre les objets détectés. Considérons l'article de (Idler *et al.*, 2006), l'image obtenue par caméra y est binarisée en fonction d'un seuil, de sorte que l'image résultante soit constituée de groupe de pixels blancs représentant les sources de lumière. Après l'application d'une détection de bord à l'image binaire, ils ont comparé et apparié deux à deux les cercles lumineux obtenus afin de déterminer les centres de véhicules (qui correspondent au point situé au milieu des deux feux). Ces centres permettent de créer une boîte rectangulaire pour le suivi des véhicules par filtrage particulière.

À partir des mêmes modèles, les auteurs de (Masoud et Papanikolopoulos, 2001) réalisent le suivi de piétons à partir d'images en niveau de gris, issues d'une caméra fixe. Ils utilisent trois niveaux de traitement : les images brutes, puis la détection de forme, enfin le piéton modélisé comme une zone (tâche) rectangulaire avec un certain comportement dynamique. Ce rectangle est alors suivi par un filtre de Kalman.

Dans ces deux exemples, les auteurs ont choisi d'utiliser deux modèles pour caractériser les objets, un pour la détection et l'autre pour le suivi. Il est possible de ne choisir qu'un modèle comme l'on fait les auteurs de (Ge *et al.*, 2009). Dans cet article, les auteurs réalisent la détection et le suivi de piétons de nuit, par caméra infrarouge. Les objets sont modélisés par une région d'intérêt (ROI : *Region Of Interest*) qu'ils mettent en forme par des rectangles dont les dimensions sont définies à partir d'un seuillage de l'image (les piétons sont plus lumineux que le reste de l'image).

On peut également citer (Petrovskaia et Thrun, 2008; Petrovskaia et S. Thrun, 2009) qui décrivent un module de suivi de véhicules en mouvement. Le module assure le suivi des véhicules à partir d'une plateforme mobile évoluant à haute vitesse qui est équipée de télémètres laser. Ils s'intéressent à la construction de représentations 2D des véhicules à partir de données 3D. Ils utilisent un modèle géométrique rectangulaire du véhicule et estiment la forme des véhicules. (Petrovskaia *et al.*, 2012) propose une évolution de leur méthode au cas notamment des piétons.

On remarque ici que le choix du modèle se rapproche souvent d'un rectangle.

Dans notre cas d'une détection et d'un suivi d'obstacles via les informations d'un lidar, nous verrons au chapitre III que nous utilisons une modélisation classique du système sous forme de vecteur d'état liée au centre de gravité de l'objet. Nous utiliserons dans un module dit de surveillance une modélisation des véhicules sous forme de boîte englobante dont on retiendra en particulier le centre de gravité. Nous utiliserons dans un autre module intégré dédié à la détection et au suivi une modélisation que l'on peut appeler modélisation inverse : à partir de la connaissance d'un centre de gravité, une boîte englobante est construite à partir de laquelle on retrouve une modélisation sous forme d'échos d'un véhicule tels que ceux qu'un lidar nous fournirait.

II.4 Association de données

L'architecture fonctionnelle classiquement adoptée, figure II.1, comporte trois étapes de traitement : la détection, l'association et le suivi d'objets. Cependant, les étapes de détection et d'association d'objets peuvent introduire des difficultés au niveau du suivi des objets. L'étape de détection, au cours de laquelle les mesures en sortie du capteur (données brutes) sont prétraitées, peut engendrer des pertes d'informations menant à des cas de non détection ou à des résultats aberrants (tels que des fausses alarmes). L'origine de ces pertes est la suivante. Lors du prétraitement des données, on extrait les objets à suivre. Cela revient à ne prendre qu'une partie de la mesure concernant les objets. En réalisant le prétraitement, on s'octroie donc la possibilité de perdre des informations du fait, par exemple, d'une détection trop approximative ou d'une occultation d'un objet (celui-ci devenant trop petit en nombre de mesures n'est alors pas assimilable à un objet à suivre). Plusieurs solutions sont possibles pour pallier cette difficulté. Une première solution est d'améliorer le processus de détection dans l'intention d'augmenter la probabilité de détection des objets tout en minimisant la probabilité de fausse alarme et accroître les performances des algorithmes d'association temporelle. Une seconde solution est de mettre en œuvre une approche méthodologique qui traite de façon conjointe les problèmes de détection et de suivi (voir figure III.4).

Les problèmes d'association de données sont le résultat de mauvaises associations entre les éléments de l'instant courant et ceux de l'instant précédent. Dans le contexte général du suivi multicible, la détection et le suivi des objets à partir de mesures laser augmentent le problème de l'association de données (Bar-Shalom et Fortmann, 1988). En effet, pour chaque nouvel échantillon de mesures, le module de surveillance cherche à l'associer à une piste précédemment trouvée. Une piste correspond à un ensemble de variable caractérisant un unique objet suivi. Le problème de l'association de données est amplifié par les problèmes de détection. Ainsi, si un objet n'est plus détecté par manque d'informations (occultation, mauvaise détection...), l'objet est éliminé des pistes à suivre. Aussi, si un objet est détecté par erreur, une piste est créée. Enfin, si une piste n'est pas associée à un objet, du fait d'une mauvaise détection ou d'un mauvais suivi, des pistes supplémentaires sont alors créées inutilement car plusieurs pistes suivent le même objet.

Plusieurs méthodes sont proposées pour effectuer l'étape d'association de données.

Parmi les plus populaires, on peut citer les approches de suivi à hypothèses multiples (Reid, 1979) (MHT : *Multiple Hypothesis Tracking*) qui, pour un horizon temporel donné, envisagent toutes les associations possibles entre les mesures et les pistes. La méthode MHT traite de l'association de séquences

d'observations et de l'évaluation de la probabilité de chaque séquence. L'objectif est de garder les hypothèses les plus probables. L'évaluation dépend de trois paramètres : la probabilité d'association entre la piste et la nouvelle observation, le vecteur d'état et la matrice de covariance de l'erreur de la piste (Reid, 1979).

Une autre classe de méthodes, appelée association probabiliste conjointe de données ou *Joint Probabilistic Data Association* (JPDA) en anglais, modélise le problème d'association de manière probabiliste (Bar-Shalom et Fortmann, 1988). Pour cette méthode, toutes les hypothèses d'association sont étudiées et une probabilité est calculée à chaque hypothèse construite. C'est cette probabilité qui permet de déterminer la meilleure hypothèse possible. Le cas des fausses alarmes est pris en compte. L'inconvénient ici est la croissance du nombre d'hypothèses dans le cas où il y a des observations appartenant à plusieurs pistes (Bar-Shalom et Fortmann, 1988; Bar-Shalom et Li, 1995).

II.5 Méthode de suivi : filtrage

Nous nous intéressons aux techniques de filtrage permettant d'estimer les états d'un système dynamique à partir de mesures incomplètes. Pour cela, le système dynamique est classiquement modélisé par un système d'état composé d'une équation de dynamique et d'une équation de mesure comme suit :

$$\begin{cases} X_t = f_t(X_{t-1}) + v_t \\ Z_t = h_t(X_t) + n_t \end{cases} \quad (\text{II.3})$$

avec f_t la fonction de dynamique (ou flot du système), h_t la fonction de mesure, v_t le bruit de modèle et n_t le bruit de mesure.

On montre que la solution du problème d'estimation à minimum de variance est donnée par la densité de probabilité conditionnelle $p(X_t|Z_{0:t})$. Celle-ci peut être décomposée en deux lois : une loi de transition $p(X_t|X_{t-1})$ liée à l'équation de dynamique et une loi d'observation $p(Z_t|X_t)$ liée à celle de mesure. Cette densité de probabilité s'écrit :

$$p(X_t|Z_{0:t}) = \prod_{j=1}^t p(X_j|X_{j-1})p(X_0) \prod_{j=1}^t p(Z_j|X_j)$$

La résolution du problème d'estimation d'état mène à différentes solutions de filtrage selon que :

- les fonctions f_t et h_t sont linéaires et les bruits v et n sont gaussiens,
- les fonctions f_t et h_t sont non-linéaires et (ou) les bruits v et n sont non-gaussiens.

Résolution du problème d'estimation d'état lorsque les fonctions f et h sont linéaires et les bruits v et n sont gaussiens

Dans le cas de fonctions linéaires et de bruits gaussiens, la densité de probabilité $p(X_t|Z_{0:t})$ est gaussienne, car les lois d'observation et de transition le sont. Le système peut alors être décrit par ses deux premiers moments (moyenne et variance). Le filtre de Kalman (Bar-Shalom et Li, 1995) permet d'estimer la moyenne ($\hat{X}_{t|t} = E(X_t|Z_{0:t})$) et la variance est la covariance ($P_{t|t} = E(\hat{X}_{t|t}\hat{X}_{t|t}^T|Z_{0:t})$).

Résolution du problème d'estimation d'état lorsque les fonctions f et h sont non-linéaires ou les bruits v et n sont non-gaussiens

Dans le cas de fonctions non-linéaires ou de bruits non-gaussiens, le système est défini par une équation des moments de dimension infinie. La littérature traite ce cas selon deux orientations :

- les fonctions f_t et h_t sont linéarisées,
- les non-linéarités du système sont traitées.

En linéarisant les fonctions f_t et h_t , le système est de nouveau défini par ses deux premiers moments. Le filtre de Kalman étendu (EKF : *Extend Kalman Filter*) (Bar-Shalom et Fortmann, 1988) est alors construit sur ces linéarisations. Il repose sur l'approximation de la loi par une gaussienne dont les moments sont calculés par linéarisation des équations d'état. Cependant, la linéarisation réalisée par l'EKF entraîne une erreur systématique due à l'approximation effectuée (voir figure II.7), pouvant conduire à une divergence de la solution. Pour atténuer ces erreurs, on peut utiliser un modèle itératif pour améliorer la linéarisation du modèle (on parle alors de filtre de Kalman étendu itéré), ou encore choisir un système de coordonnées judicieux. Pour un filtre construit sur l'estimation des deux premiers moments, l'idéal serait que la moyenne de la loi approximée soit égale à la moyenne de la loi exacte, et qu'il en soit de même pour la covariance. Par construction, le filtre de Kalman étendu ne permet pas de garantir cette égalité. Pour résoudre ce problème, Julier et Uhlmann (1997) ont proposé une évolution de l'EKF qui repose sur la transformation « *unscented* » : le filtre de Kalman sans parfum (UKF : *Unscented Kalman Filter*).

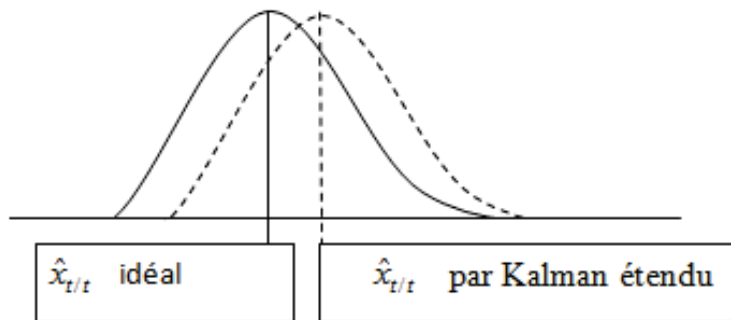


FIGURE II.7. Biais de l'estimateur de Kalman étendu

Ces premières méthodes sont limitées à l'estimation des deux premiers moments de la loi conditionnelle. Parmi l'ensemble des méthodes qui traitent les non-linéarités du système, on peut trouver notamment les méthodes de maillage de l'espace d'état (Lindner et Wanielik, 2009) ou les méthodes de filtrage particulaire (Moral *et al.*, 1992; Gordon *et al.*, 1993; Takeuchi *et al.*, 2010).

Le maillage est une approche numérique qui consiste à évaluer la fonction de densité de probabilité sur une grille dans l'espace d'état. L'espace d'état est alors décomposé en N cellules. Cependant, le choix des paramètres n'est pas trivial et il faut un grand nombre de points dans la grille pour obtenir une bonne approximation de la densité de probabilité. L'augmentation du coût de calcul, augmente rapidement avec la dimension de l'espace d'état (Arulampalam *et al.*, 2002) ce qui rend son utilisation malaisée en pratique.

Les méthodes de filtrage particulaire, également appelées méthodes de Monte-Carlo Séquentielles, ont été introduites par (Moral *et al.*, 1992; Gordon *et al.*, 1993). Ces méthodes reposent directement sur l'estimation de la densité de probabilité, solution du problème de filtrage non-linéaire. Contrairement aux méthodes de maillage, ces méthodes, basées sur la loi des grands nombres, ont des performances moins sensibles à la dimension de l'espace d'état. Les filtres particuliers ne résolvent pas explicitement les équations du problème, ils sont donc applicables quelles que soient les non-linéarités de modèle.

Nous détaillons dans les sections qui suivent le filtre de Kalman et ses dérivés (section II.6), puis le filtrage particulaire (section II.7). Dans chacune de ces sections, nous présentons les techniques de filtrage et donnons quelques exemples d'application des filtres au problème de la détection et du suivi d'obstacles dans les systèmes automobiles. Notons cependant que l'ensemble de ces filtres ont de bien plus nombreuses applications dans d'autres secteurs d'activité (finance, robotique, ingénierie...).

II.6 Filtre de Kalman et ses filtres dérivés

Nous présentons brièvement dans cette partie le filtre de Kalman applicable au cas linéaire et gaussien et ses filtres dérivés les plus populaires : le filtre de Kalman étendu et le filtre de Kalman sans parfum applicables au cas non-linéaires.

II.6.1 Filtre de Kalman

Le filtre de Kalman a été proposé par Kalman au début des années 1960. C'est un estimateur récursif. Cela signifie que pour estimer l'état courant, seules l'estimation de l'état précédent et les mesures actuelles sont nécessaires. L'historique des observations et des estimations n'est ainsi pas requise. L'état du filtre est représenté par deux variables : $\hat{X}_{t|t}$, l'estimation de l'état à l'instant t ; $P_{t|t}$, la matrice de covariance de l'erreur (une mesure de la précision de l'état estimé).

Le filtre de Kalman peut se décomposer en deux étapes (Bar-Shalom et Li, 1995) :

- une étape de prédiction qui exploite l'équation de dynamique du système pour estimer le vecteur d'état X_t à l'instant t et la covariance d'erreur de prédiction associée ;
- une étape de correction qui exploite les mesures disponibles à l'instant t pour corriger l'estimation précédente dans le but d'obtenir une estimation plus précise.

L'algorithme qui en découle est le suivant (Bar-Shalom et Li, 1995; Chow *et al.*, 2007) :

- Soient les conditions initiales suivantes :
 $\hat{X}_{0|0} = X(0)$, avec $X(0)$ une variable aléatoire gaussienne de moyenne \bar{X}_0 et de covariance P_0 ;
 $P_{0|0} = P_0$.
- **Étape de prédiction :**
 Prédiction de l'état : $\hat{X}_{t|t-1} = F\hat{X}_{t-1|t-1}$
 Covariance de l'erreur de prédiction : $P_{t|t-1} = FP_{t-1|t-1}F^T + Q_t$
- **Étape de correction :**
 Correction d'état : $\hat{X}_{t|t} = \hat{X}_{t|t-1} + K_t(Z_t - H\hat{X}_{t|t-1})$
 Covariance de l'erreur d'estimation : $P_{t|t} = (I - K_tH)P_{t|t-1}$
 Gain du filtre : $K_t = P_{t|t-1}H^T (HP_{t|t-1}H^T + R_t)^{-1}$
- où F et H sont les matrices de dynamique et d'observation du système.
 I est la matrice identité. Et Q_t et R_t sont les matrices de covariance des bruits blancs gaussiens additifs de moyenne nulle v_t et n_t (respectivement) du système dynamique (équation II.3).

On remarque que la correction apportée à l'état prédit est pondérée par la matrice de gain K_t dont l'interprétation est la suivante :

- un état prédit peu précis et une mesure précise impliquent un grand gain K_t et par conséquent une grande correction apportée, ceci traduit une confiance de l'estimateur dans les mesures ;
- inversement, un état prédit précis et une mesure peu précise impliquent un faible gain K_t , l'estimateur aura cette fois confiance dans le modèle dynamique.

II.6.2 Filtre de Kalman étendu (EKF)

Le filtre de Kalman étendu ou *Extended Kalman Filter* (EKF) est une extension du filtre de Kalman permettant de résoudre des problèmes faiblement non-linéaires. C'est Bucy qui développa le filtre de Kalman étendu en 1965.

L'EKF garde la même structure (prédiction, mise à jour) que le filtre de Kalman, cependant une étape de linéarisation (autour de l'estimation courante du système) est introduite en prenant pour matrices de dynamique et d'observation du système les jacobiniennes des fonctions f_t et h_t respectivement (Bar-Shalom et Fortmann, 1988; Bar-Shalom et Li, 1995; Wan et Merwe, 2000, 2001; Chow *et al.*, 2007; Constantin *et al.*, 2008; Kandepe *et al.*, 2008).

Les équations du filtre deviennent :

– Soient les conditions initiales suivantes :	
$\hat{X}_{0 0} = X(0)$, avec $X(0)$ une variable aléatoire gaussienne de moyenne \bar{X}_0 et de covariance P_0 ;	
$P_{0 0} = P_0$.	
– Prédiction :	
Prédiction de l'état :	$\hat{X}_{t t-1} = f_t(\hat{X}_{t-1 t-1})$
Covariance de l'erreur de prédiction :	$P_{t t-1} = F_t P_{t-1 t-1} F_t^T + Q_t$
– Correction :	
Correction d'état :	$\hat{X}_{t t} = \hat{X}_{t t-1} + K_t (Z_t - h(\hat{X}_{t t-1}))$
Covariance de l'erreur d'estimation	$P_{t t} = (I - K_t H_t) P_{t t-1}$
Gain du filtre :	$K_t = P_{t t-1} H_t^T (H_t P_{t t-1} H_t^T + R_t)^{-1}$
– où $F_t = \nabla_X f_t _{\hat{X}_{t-1 t-1}}$ et $H_t = \nabla_X h_t _{\hat{X}_{t t-1}}$	
Et Q_t et R_t sont les matrices de covariance des bruits blancs gaussiens additifs de moyenne nulle v_t et n_t (respectivement) du système dynamique (équation II.3).	

Dans l'EKF la distribution de l'état est approchée par une variable aléatoire gaussienne. Cette variable \hat{X} est ensuite propagée à travers la linéarisation analytique de premier ordre du système non-linéaire. En tant que tel, l'EKF peut être considéré comme fournissant, une approximation au premier ordre. Cependant, cette approximation peut introduire des erreurs importantes dans l'estimation de la moyenne et la covariance de la variable aléatoire transformée, ce qui peut conduire à des sous-performances et parfois à la divergence du filtre.

Ce sont ces défauts qui seront abordés dans la section suivante en utilisant le filtre de Kalman sans parfum.

II.6.3 Filtre de Kalman sans parfum (UKF)

Le filtre de Kalman sans parfum ou *Unscented Kalman filter* (UKF) a été développé par Julier et Uhlmann en 1997 (Constantin *et al.*, 2008). Ce filtre est une évolution des filtres de Kalman qui n'utilise pas la linéarisation des équations d'état et d'observation, pouvant être des sources d'erreurs.

Comme pour l'EKF la distribution de l'état est représentée par une variable aléatoire gaussienne. Cette variable est désormais calculée en utilisant un ensemble minimal de N points d'échantillonnage caractéristique. Ces points d'échantillonnage ont pour but de déterminer la moyenne et la covariance de la variable aléatoire gaussienne. Lorsqu'ils sont propagés à travers le système non-linéaire, ils capturent la moyenne et la covariance *a posteriori* avec une précision du second ordre pour toutes les non-linéarités (Julier et Uhlmann, 1997; Wan et Merwe, 2000, 2001; Chow *et al.*, 2007; Morelande et Moran, 2007; Constantin *et al.*, 2008; Kandepe *et al.*, 2008; Wang *et al.*, 2009).

L'UKF est une extension de la transformation sans parfum (*Unscented Transformation*). La transformation sans parfum est une méthode pour calculer les statistiques d'une variable aléatoire issue d'une transformation non-linéaire et définie comme suit :

- on considère la propagation d'une variable aléatoire (X) de dimension (L) à travers une fonction non-linéaire ($B = h(X)$) de moyenne ($\hat{X}_{t|t-1}$) et de covariance ($P_{X_t|Z_t|t}$);
- pour calculer la statistique de B on crée une matrice (χ) représentant les points sigma et composée des éléments (χ_j) définis ci-dessous (équation II.4);

$$\begin{cases} \chi_0 = \hat{X}_{t|t-1} \\ \chi_j = \hat{X}_{t|t-1} + \left(\sqrt{(L+\lambda)P_{X_t|Z_t|t}} \right)_j \quad \forall j \in [1, L] \\ \chi_j = \hat{X}_{t|t-1} - \left(\sqrt{(L+\lambda)P_{X_t|Z_t|t}} \right)_{j-L} \quad \forall j \in [L+1, 2L] \end{cases} \quad (\text{II.4})$$

- chacun des éléments (χ_j) de la matrice est propagé selon la fonction non-linéaire (B);
- la moyenne ($\hat{Z}_{t|t-1}$) et la covariance ($P_{\tilde{Z}_{t|t-1}}$) du point sigma sont approximées en utilisant le poids (W_j).

W_j est défini comme suit (équation II.5) :

$$\begin{cases} W_0^{(m)} = \lambda / (L + \lambda) \\ W_0^{(c)} = \lambda / (L + \lambda) + (1 - \alpha^2 + \beta) \\ W_j^{(m)} = 1 / (2(L + \lambda)) \quad \forall j \in [1, 2L] \end{cases} \quad (\text{II.5})$$

Avec $\lambda = \alpha^2(L+k) - L$ un paramètre scalaire et L la dimension de l'espace,

α une constante : $1 \leq \alpha \leq 1.e^{-4}$,

β intègre les connaissances préalables de la distribution X ; pour une distribution gaussienne $\beta = 2$ (Constantin *et al.*, 2008),

enfin, $\left(\sqrt{(L+\lambda)P_{X_t|Z_t|t}} \right)_j$ est la j^{e} colonne de la racine carrée de la matrice.

L'état de la variable aléatoire est redéfini par augmentation de l'état original avec la variable de bruit.

$$X_t^a = [X_t^T \quad v_t^T \quad n_t]^T$$

La transformation sans parfum du point sigma est alors appliquée à cette nouvelle variable aléatoire d'état augmenté afin de déterminer la matrice sigma (χ_t^a) correspondante.

Les équations détaillées du filtre sont les suivantes :

- Soient les conditions initiales suivantes :
 $\hat{X}_{0|0} = X(0)$, avec $X(0)$ une variable aléatoire gaussienne de moyenne \bar{X}_0 et de covariance P_0 ;
 $P_{0|0} = P_0$;
 $\hat{X}_{0|0}^a = [\hat{X}_{0|0}^T \quad 0 \quad 0]^T$;
 $P_{0|0}^a = \begin{bmatrix} P_{0|0} & 0 & 0 \\ 0 & Q_t & 0 \\ 0 & 0 & R_t \end{bmatrix}$
- **Calcul du point Sigma :**
 $\chi_{t|t}^a = \left[\hat{X}_{t|t}^a \quad \hat{X}_{t|t}^a + \gamma \sqrt{P_{t|t}^a} \quad \hat{X}_{t|t}^a - \gamma \sqrt{P_{t|t}^a} \right]$

<p>– Prédiction :</p> $\mathcal{X}_{j,t t-1}^X = f_t(\mathcal{X}_{j,t-1 t-1})$ <p>Prédiction de l'état : $\hat{X}_{t t-1} = \sum_{j=0}^{2L} W_j^m \mathcal{X}_{j,t t-1}^X$</p> <p>Covariance prédite : $P_{t t-1} = \sum_{j=0}^{2L} W_j^c [\mathcal{X}_{j,t t-1}^X - \hat{X}_{t t-1}] [\mathcal{X}_{j,t t-1}^X - \hat{X}_{t t-1}]^T$</p> $B_{j,t t-1} = h_t(\mathcal{X}_{j,t t-1}^X, \mathcal{X}_{j,t-1 t-1}^n)$ <p>Mesure prédite : $\hat{Z}_{t t-1} = \sum_{j=0}^{2L} W_j^m B_{j,t t-1}$</p> <p>– Correction :</p> <p>Covariance de la mesure : $P_{\tilde{Z}_t Z_t} = \sum_{j=0}^{2L} W_j^c [B_{j,t t-1} - \hat{Z}_{t t-1}] [B_{j,t t-1} - \hat{Z}_{t t-1}]^T$</p> <p>Covariance état/mesure : $P_{X_t Z_t} = \sum_{j=0}^{2L} W_j^c [\mathcal{X}_{j,t t-1}^X - \hat{X}_{t t-1}] [B_{j,t t-1} - \hat{Z}_{t t-1}]^T$</p> <p>Gain du filtre : $K_t = P_{X_t Z_t} P_{\tilde{Z}_t Z_t}^{-1}$</p> <p>Correction de l'estimé : $\hat{X}_t = \hat{X}_{t t-1} + K_t (Z_t - \hat{Z}_{t t-1})$</p> <p>Correction de la covariance : $P_t = P_{t t-1} - K_t P_{\tilde{Z}_t Z_t} K_t^T$</p> <p>– où $X^a = [X^T \quad v^T \quad n^T]^T$, $\mathcal{X}^a = [(\mathcal{X}^X)^T \quad (\mathcal{X}^v)^T \quad (\mathcal{X}^n)^T]^T$, $\gamma = \sqrt{L + \lambda}$, λ paramètre scalaire ($\lambda = \alpha^2(L + k) - L$), L dimension de l'espace, W_i le poids (défini par l'équation II.5), Q_t et R_t sont les matrices de covariance des bruits blancs gaussiens additifs de moyenne nulle v_t et n_t (respectivement) du système dynamique II.3.</p>

II.7 Filtrage particulière

II.7.1 Principe

Le principe du filtrage particulière repose sur l'exploration de l'espace d'état à l'aide de N particules aléatoires. Ces particules sont indépendantes, elles ont chacune leur évolution propre en fonction des mesures relevées par les capteurs à chaque instant t . Ces particules vont simuler une « trajectoire » possible, c'est-à-dire l'évolution d'un processus respectant les mêmes équations que celle du processus à estimer. Une particule fournit deux informations :

- un vecteur d'état de même structure que le vecteur d'état du processus estimé,
- un poids représentatif de la probabilité que ce vecteur soit celui du processus à estimer.

Elles sont modifiées suivant l'équation d'état du système (ceci correspond à l'étape de prédiction) et les poids sont ajustés en fonction des observations (étape dite de correction). Cependant, après un certain nombre de mesures, la plupart des particules ont un poids négligeable. C'est le phénomène de dégénérescence des poids. Le système de particules est appauvri et ne peut donc plus représenter correctement la densité de probabilité avec pour conséquence une divergence possible du filtre. Pour corriger ce phénomène, une étape supplémentaire dite de rééchantillonnage est introduite. Les particules de poids le plus élevé sont favorisées en les répliquant à l'identique, celles de poids faible qui se trouvent dans les régions les moins probables sont peu ou pas du tout choisies et disparaissent. Cette étape permet de concentrer la capacité du réseau

de particules dans les zones où la probabilité est maximale, augmentant ainsi la précision de l'estimation (Diginext, 1996).

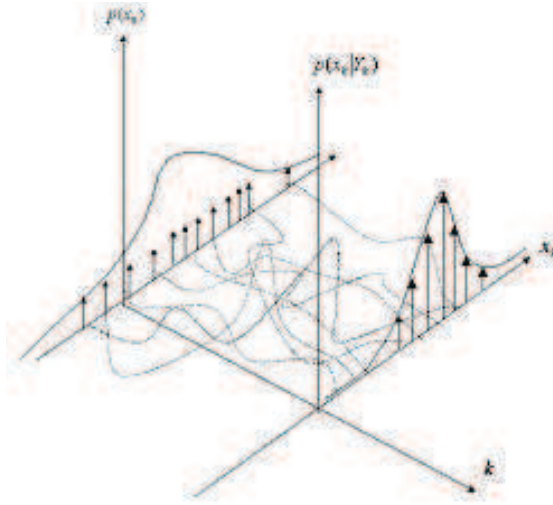


FIGURE II.8. Évolution des particules (figure extraite de Diginext (1996))

Dans sa version minimale, l'algorithme obtenu comporte cinq étapes :

- à l'instant $t = 0$:
 - initialisation des poids et des particules
- à chaque réception d'une mesure Z_t de l'instant t :
 - propagation du support de chaque particule i ,
 - mise à jour des poids $w^{(i)}$ et normalisation de ceux-ci,
 - estimation de l'état,
 - si nécessaire redistribution des particules et mise à jour des poids (rééchantillonnage).

II.7.2 Échantillonnage d'importance

Le but de la méthode de filtrage particulaire est d'estimer la distribution $p(X_t|Z_{0:t})$. Cette méthode utilise les lois de Bayes. Elle peut dans cette optique être mise en œuvre de manière séquentielle à chaque arrivée d'une nouvelle observation (Z_t). On utilisera alors la loi d'importance notée $q(X_t|Z_{0:t})$ pour échantillonner les réalisations X . Ces réalisations seront appelées particules. Rappelons que chaque particule représente une copie du vecteur d'état X_t associée à un poids w (qui caractérise la validité du positionnement de la particule dans l'espace d'état) (Chen, 2003; Doucet *et al.*, 2000b).

On peut montrer :

$$w_t = \frac{p(X_{0:t}|Z_{1:t})}{q(X_{0:t}|Z_{1:t})} \quad (\text{II.6})$$

Le caractère Markovien de X_t permet d'écrire $p(X_t|X_{0:t-1}, Z_{1:t-1}) = p(X_t|X_{t-1})$ et donc :

$$p(X_{0:t}|Z_{1:t}) = \frac{p(X_{0:t-1}|Z_{1:t-1})p(X_t|X_{t-1})p(Z_t|X_t)}{p(Z_t|Z_{1:t-1})} \quad (\text{II.7})$$

Pour obtenir un algorithme récursif, la fonction d'importance q doit vérifier la relation suivante (II.8)

$$q(X_{0:t}|Z_{1:t}) = q(X_{0:t-1}|Z_{1:t-1})q(X_t|X_{0:t-1}, Z_{1:t}) \quad (\text{II.8})$$

Si on remplace q et p (respectivement équation II.8 et II.7) dans la relation w_t (équation II.6), on obtient alors la relation récurrente suivante qui réactualise les poids w_t en fonction de $q(X_t|X_{0:t-1}, Z_{1:t})$:

$$w_t \propto w_{t-1} \frac{p(X_t|X_{0:t-1})p(Z_t|X_t)}{q(X_t|X_{0:t-1}, Z_{1:t})}$$

Si g désigne la densité de probabilité du bruit de mesure, alors $p(Z_t|X_t) = g(Z_t - h_t(X_t|_{t-1}))$. Le poids $w_t^{(i)}$ pour une particule i à l'instant t devient :

$$w_t^{(i)} \propto w_{t-1}^{(i)} \frac{p(X_t^{(i)}|X_{0:t-1}^{(i)})p(Z_t|X_t^{(i)})}{q(X_t^{(i)}|X_{0:t-1}^{(i)}, Z_{1:t})} \quad (\text{II.9})$$

L'estimée de l'état X_t est alors égale à :

$$\hat{X}_t = \sum_{i=1}^N w_t^{(i)} X_t^{(i)} \quad (\text{II.10})$$

et la matrice de covariance est estimée par :

$$P_t \approx \sum_{i=1}^N w_t^{(i)} (X_t^{(i)} - \hat{X}_t)(X_t^{(i)} - \hat{X}_t)^T$$

L'inconvénient de cette méthode de filtrage particulaire est la dégénérescence des poids des particules, dégénérescence accentuée lorsque le bruit de mesure est faible.

II.7.3 Problème de la dégénérescence des particules

On parle de dégénérescence du réseau particulaire lorsqu'un nombre significatif de particules ont des poids nul ou proches de zéro. Au cours du temps, la variance de l'estimateur particulaire augmente. Dans ces conditions, il est difficile d'éviter le phénomène de dégénérescence. Les poids étant faibles, cela implique qu'un grand effort de calcul doit être fourni pour mettre à jour les particules et ils ont peu d'impact sur l'approximation de la densité de probabilité $p(X_t|Z_{1:t})$. Pour résoudre ce problème on peut utiliser deux méthodes :

- le choix de la densité d'importance optimale pour l'application visée,
- le rééchantillonnage (ou redistribution) des poids.

II.7.3.1 Résolution de la dégénérescence par choix d'une densité d'importance optimale pour une application donnée

Cette méthode consiste à déterminer la bonne densité d'importance $q(X_t|X_{t-1}, Z_{1:t})$. Pour être efficace, elle doit minimiser la variance des poids $var(w_t^{(i)})$ et donc maximiser le nombre de particules efficaces. Si nous attachons une importance à cette densité q , c'est qu'elle conditionne le bon fonctionnement de l'algorithme. Elle doit satisfaire deux conditions :

- le support de la densité d'importance doit recouvrir le support de la loi de densité conditionnelle (sinon cela signifierait que l'espace d'approximation n'englobe pas l'espace d'état et donc que toutes les mesures ne soient pas prises en compte),
- la densité d'importance doit tenir compte de l'observation récente.

Dans la littérature, on trouvera plusieurs façons d'optimiser le choix de la densité d'importance, dans des cas généraux de traitement de données pour du suivi, en voici quelques exemples :

- Arulampalam *et al.* (2002) propose comme densité d'importance optimale $q(X_t|X_{0:t-1}, Z_{1:t}) = p(X_t|X_{t-1}^{(i)}, Z_t)$. Cependant, les auteurs remarquent deux inconvénients à ce choix, la difficulté potentielle à calculer $p(X_t|X_{t-1}^{(i)}, Z_t)$ et à évaluer l'intégrale sur un nouvel état. Malgré tout, l'utilisation de cette densité est possible dans deux cas, quand X_t appartient à un ensemble fini (dans ce cas, l'intégrale devient une somme et l'échantillonnage est possible) et lorsque $p(X_t|X_{t-1}^{(i)}, Z_t)$ est gaussien.
- Dahia (2005) évoque plusieurs possibilités pour le choix de la densité d'importance, l'une est de prendre $q(X_t|X_{0:t-1}, Z_{1:t})$ égale à une loi de densité Gaussienne, telle que la moyenne et la matrice de covariance soient calculées par les équations de Kalman en linéarisant les équations d'état et de mesure. L'autre est de choisir la densité *a priori* $q(X_t|X_{0:t-1}, Z_{1:t}) = p(X_t|X_{t-1})$.
- Klein (2006) constate que pour son application, le meilleur choix de densité d'importance est $q(X_{0:t}|Z_{1:t}) = p(X_{0:t}|Z_{1:t})$. Cependant, ce choix revient à ne pas se servir d'une densité d'importance. Il propose alors la densité d'importance $q(X_t|X_{0:t-1}, Z_{1:t}) = p(X_t|X_{t-1}^{(i)}, Z_t)$ et constate qu'en pratique ce choix est peu réaliste car il faut être capable d'échantillonner selon $q(\cdot|X_{0:t-1}^{(i)}, Z_{0:t})$, mais aussi d'estimer $p(Z_t|X_{0:t}^{(i)})$. L'auteur conclut qu'une solution approximative peut être fournie par des méthodes de Monte-Carlo ou par l'utilisation d'un développement de Taylor à l'ordre deux de $l(X_t) = \ln p(X_t|X_{t-1}^{(i)}, Z_t)$.

Pour atteindre des performances élevées en termes de résultats, le coût de calcul additionnel est souvent jugé trop important par les auteurs. C'est pourquoi, il devient plus intéressant d'utiliser d'autres méthodes pour diminuer la dégénérescence des poids. Dans ce contexte, le paragraphe suivant s'intéresse au rééchantillonnage des poids (également appelé redistribution).

II.7.3.2 Résolution de la dégénérescence par rééchantillonnage des poids

Le principe du rééchantillonnage est décrit dans plusieurs documents tels que (Chen, 2003; Dahia, 2005; Arulampalam *et al.*, 2002; LeGland, 2003; Crisan et Doucet, 2002). Comme on l'a vu dans les parties précédentes, le filtrage particulaire conduit après plusieurs itérations à une dégénérescence des particules. Pour résoudre ce problème de dégénérescence, une étape dite de rééchantillonnage est ajoutée au filtre. Elle réalise une redistribution des particules du filtre sur les régions de probabilité élevée en supprimant les particules de poids faible et en dupliquant celles de poids élevé (voir figure II.9 extraite de (Freitas *et al.*, 2000; Chen, 2003)).

Ce rafraîchissement du nuage de particules limite alors le problème de dégénérescence. Cependant, s'il est trop fréquent, il aura un effet similaire à celui de l'échantillonnage en appauvrissant l'échantillon de particules. L'idéal est de permettre aux particules d'explorer l'espace d'état sur plusieurs pas de temps avant d'être supprimées. Pour cela, on se propose d'effectuer la redistribution lorsque le nombre de particules efficaces est en dessous d'un seuil fixé, c'est le rééchantillonnage adaptatif. Avant de réaliser le rééchantillonnage à proprement parler, une première étape au rééchantillonnage adaptatif est le choix du seuil. Ce seuil est fixé. Il permet de suivre l'évolution du poids des particules et de prévenir l'appauvrissement des particules. Pour cela, on pose (Chen, 2003; Arulampalam *et al.*, 2002) :

- N le nombre de particules du filtre,
- N_S le seuil,
- N_{eff} le nombre de particules efficaces.

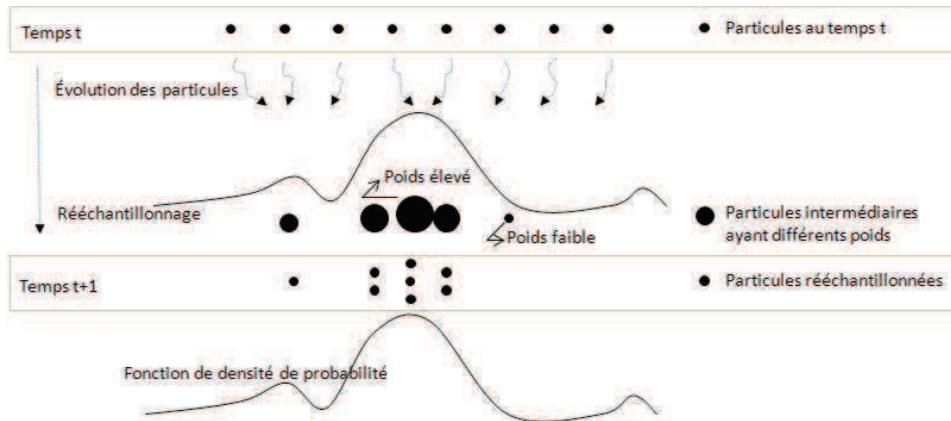


FIGURE II.9. Rééchantillonnage des particules

En pratique, il est difficile de calculer de façon exacte le nombre de particules efficaces défini par

$$N_{eff} = \frac{N_S}{1 + var(w_t^{*,(i)})} \text{ où } w_t^{*,(i)} = p(X_t^{(i)} | Y_{1:t}) / q(X_t^{(i)} | X_{t-1}^{(i)}, Y_t)$$

Dans (Chen, 2003; Arulampalam *et al.*, 2002) une estimation de ce nombre de particule efficace est donnée par : $N_{eff} = 1 / \sum_{i=1}^N (w_t^{(i)})^2$ où $(w_t^{(i)})$ est le poids normalisé de la particule (i) à l'instant t . La méthode consiste alors à comparer N_{eff} et N_S . Si $N_{eff} < N_S$, une redistribution des particules est réalisée, sinon l'algorithme se poursuit avec l'étape de calcul des particules.

Le rééchantillonnage se réalise quant à lui ainsi :

Soit Φ une fonction de répartition utilisée pour redistribuer l'ensemble des particules sur l'espace d'état (de telles fonctions sont généralement utilisées pour générer des tirages aléatoires selon des lois données), dont la forme est $\Phi(X) = \int_{-\infty}^X p(X) dX$ dans le cas continu ou $\Phi(X) = \sum_{i=1}^X w^{(i)}$ dans le cas discret. En se basant sur le principe de la fonction de répartition, la redistribution s'effectue ainsi :

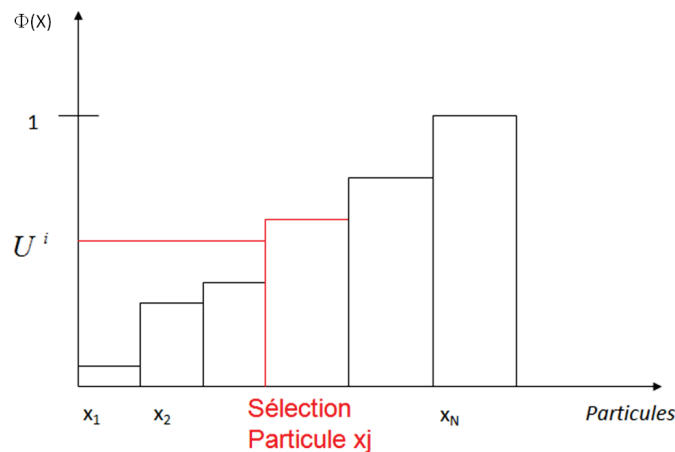
- on crée la fonction de répartition (Φ) du système discret,
- on tire un vecteur U de taille i , $i \in [1, N]$, uniformément sur $[0, 1]$,
- en inversant la fonction de répartition, on identifie les particules sur lesquelles la redistribution aura lieu, les « anciennes » particules disparaissent (voir figure II.10).

Sur cette figure, on peut voir que la particule X_j a été sélectionnée par un élément du vecteur U . Si cette étape réduit l'effet du problème de dégénérescence, elle a aussi des inconvénients :

- elle limite la possibilité de paralléliser l'utilisation des particules, car elles sont combinées,
- les particules qui ont un poids élevé sont statistiquement sélectionnées à de nombreux pas de temps. Ceci conduit à une perte de la diversité des particules et l'échantillon résultant contient beaucoup de points répétés. Ce problème est appelé « appauvrissement d'échantillon ». Il est plus important dans le cas de bruit de processus faible.

On remarquera également que le rééchantillonnage rompt l'indépendance des particules, ce qui ne permet plus l'application des théorèmes classiques de convergence tels que la loi des grands nombres et le théorème de la limite centrale, par exemple (Chen, 2003).

Dans ce document, nous avons présenté une méthode de redistribution basée sur le principe d'une fonction de répartition. Il en existe d'autres que nous n'avons pas détaillées ici (par exemple, Arulampalam *et al.* (2002) proposent une approche par récurrence).

FIGURE II.10. Sélection des particules selon le vecteur U

II.8 Présentation de différents filtres particuliers

Après avoir présenté d'une façon générale le fonctionnement des méthodes de Monte Carlo séquentielle, nous allons, dans cette partie, présenter quelques filtres particuliers que nous avons étudiés :

- le filtre à échantillonnage séquentiel par importance (SIS : *Sequential Importance Sampling*) ;
- le filtre Bootstrap ou à rééchantillonnage avec échantillonnage par importance (SIR : *Sampling Importance Resampling*) ;
- le filtre particulaire à rééchantillonnage adaptatif (SIR adaptatif) ;
- le filtre particulaire auxiliaire (APF : *Auxiliary Particle Filter*) ;
- le filtre de Rao-Blackwell (RBF : *Rao-Blackwell Filter*).

Les principales différences existant entre ces filtres sont la technique utilisée pour limiter la dégénérescence des particules ainsi que la prise en compte des linéarités.

II.8.1 Filtre à échantillonnage séquentiel par importance (SIS)

Le filtre d'échantillonnage séquentiel par importance comporte une phase de prédiction et une autre de mise à jour :

Prédiction : chaque échantillon passe dans le modèle de système pour obtenir des échantillons au temps t ;

Mise à jour : à la réception de la mesure (Z_t), la probabilité de chaque échantillon précédent est évaluée et les poids ($w^{(i)}$) de tous les échantillons sont normalisés.

Les différentes étapes de réalisation du filtre sont présentées ci-dessous pour tout $t \geq 0$:

- $\forall i \in [1, N]$ extraction de l'échantillon $X_t^{(i)} \sim q(X_t | X_{0:t-1}^{(i)}, Z_{0:t})$ et mise en mémoire $X_{0:t}^{(i)} = \{X_{0:t-1}^{(i)}, X_t^{(i)}\}$
- $\forall i \in [1, N]$ calcul du poids d'importance $w_t^{(i)}$ de la particule i à l'instant t (équation II.9)

$$w_t^{(i)} = w_{t-1}^{(i)} \frac{p(Z_t | X_t^{(i)}) p(X_t^{(i)} | X_{t-1}^{(i)})}{q(X_t^{(i)} | X_{0:t-1}^{(i)}, Z_t)}$$
- $\forall i \in [1, N]$ normalisation des poids d'importance

$$w_t^{(i)} = \frac{w_t^{(i)}}{\sum_{j=1}^N w_t^{(j)}} \quad (\text{II.11})$$
- calcul du filtre \hat{X}_t (équation II.10) en fonction des particules $X_t^{(i)}$ et de leur poids $w_t^{(i)}$ respectif

$$\hat{X}_t = \sum_{i=1}^N w_t^{(i)} X_t^{(i)}$$

Le problème de ce filtre est que les poids d'importance de la distribution deviennent de plus en plus déséquilibrés avec le temps. Après un certain nombre d'itérations, le filtre SIS dégénère, l'échantillon s'appauvrit (Doucet *et al.*, 2000b; Chen, 2003; Arulampalam *et al.*, 2002; Crisan et Doucet, 2002). Afin d'atténuer le problème d'appauvrissement de l'échantillon, les solutions proposées par les filtres présentés ci-après se focalisent sur l'amélioration des schémas d'échantillonnage et de rééchantillonnage et sur la réduction de la variance.

II.8.2 Filtre Bootstrap ou SIR

Le filtre Bootstrap (ou SIR) a été développé par Gordon *et al.* (1993). Ce filtre est une évolution du filtre SIS auquel une étape de rééchantillonnage a été ajoutée. Le filtre est construit suivant les trois phases suivantes (Spengler, 2005) :

Prédiction : chaque échantillon passe dans le modèle de système pour obtenir des échantillons au temps t ;

Mise à jour : à la réception de la mesure (Z_t), la probabilité de chaque échantillon précédent est évaluée et les poids ($w^{(i)}$) de tous les échantillons sont normalisés.

Rééchantillonnage « systématique » (voir partie II.7.3.2).

Les différentes étapes de réalisation du filtre sont présentées ci-dessous pour tout $t \geq 0$:

- initialisation du poids $w_t^{(i)} = 1/N$ et de la particule i à l'instant $t = 0$, $X_0^{(i)} \sim p(X_0)$
- $\forall i \in [1, N]$ échantillonnage d'importance, calcul de $\hat{X}_t^{(i)} = p(X_t | X_{t-1}^{(i)})$ et mise en mémoire de l'échantillon $\hat{X}_{0:t}^{(i)} = \{\hat{X}_{0:t-1}^{(i)}, \hat{X}_t^{(i)}\}$
- $\forall i \in [1, N]$ mise à jour du poids d'importance $w_t^{(i)}$ de la particule i à l'instant t

$$w_t^{(i)} = p(Z_t | \hat{X}_t^{(i)}) \quad (\text{II.12})$$
- $\forall i \in [1, N]$ normalisation des poids d'importance $w_t^{(i)}$ à l'aide de l'équation II.11
- $\forall i \in [1, N]$ rééchantillonnage, génération de N nouvelles particules ($X_t^{(i)}$) issues de l'ensemble $\hat{X}_t^{(i)}$ et des poids d'importances normalisés $w_t^{(i)}$ (voir partie II.7.3.2)
- ré-initialisation des poids après rééchantillonnage $w_t^{(i)} = 1/N$
- calcul de l'état estimé \hat{X}_t en fonction des particules $X_t^{(i)}$ et de leur poids $w_t^{(i)}$ respectif suivant l'expression II.10

L'inconvénient du filtre Bootstrap est l'appauvrissement de l'échantillonnage dû, dans la plupart des cas, au rééchantillonnage systématique du filtre (Arulampalam *et al.*, 2002; Chen, 2003).

II.8.3 Filtre à rééchantillonnage adaptatif (SIR adaptatif)

Le filtre SIR adaptatif réduit le problème de dégénérescence des particules en ne provoquant pas l'appauvrissement de celles-ci. Ici le rééchantillonnage est adaptatif. Cela permet de minimiser la variance des poids. Comme pour le filtre SIR, cet algorithme comporte trois étapes :

Prédiction : chaque échantillon passe dans le modèle de système pour obtenir des échantillons au temps t ;

Mise à jour : à la réception de la mesure (Z_t), la probabilité de chaque échantillon précédent est évaluée et les poids ($w^{(i)}$) de tous les échantillons sont normalisés.

Rééchantillonnage « adaptatif » (voir partie II.7.3.2)

Les différentes étapes de réalisation du filtre sont présentées ci-dessous pour tout $t \geq 0$:

- initialisation du poids $w_t^{(i)} = 1/N$ et de la particule i à l'instant $t = 0$, $X_0^{(i)}$
- $\forall i \in [1, N]$ échantillonnage d'importance, calcul de $\hat{X}_t^{(i)} = p(X_t | X_{t-1}^{(i)})$
et mise en mémoire de l'échantillon $\hat{X}_{0:t}^{(i)} = \{\hat{X}_{0:t-1}^{(i)}, \hat{X}_t^{(i)}\}$
- $\forall i \in [1, N]$ mise à jour du poids d'importance $w_t^{(i)}$ de la particule i à l'instant t à l'aide de la formule II.12
- $\forall i \in [1, N]$ normalisation des poids d'importance à l'aide de l'équation II.11
- si besoin rééchantillonnage. C'est-à-dire si $N_{eff} < N_S$
avec $N_{eff} = 1 / \sum_{i=1}^N (w_t^{(i)})^2$
- Alors on rééchantillonne l'échantillon, on génère N nouvelles particules ($X_t^{(i)}$) issues de l'ensemble $\hat{X}_t^{(i)}$ et des poids d'importance normalisés $w_t^{(i)}$ (voir partie II.7.3.2)
- calcul du filtre \hat{X}_t en fonction des particules $X_t^{(i)}$ et de leur poids $w_t^{(i)}$ respectif selon l'expression II.10

II.8.4 Filtre particulaire auxiliaire (APF)

Une faiblesse potentielle des filtres particulaires, décrite précédemment, est l'approximation faite par les particules de la densité du filtre. Cette approximation n'est pas suffisante pour caractériser la densité du fait de l'utilisation d'une approximation finie. Dans un filtre SIR classique, la variance des poids peut être élevée. Dans ce cas de figure, la redistribution permet d'avoir beaucoup de particules ayant un poids élevée et quelques particules ayant un poids plus faible. La question que se posent Pitt et Shephard (1999) est : « est-il possible de redistribuer plus efficacement les particules de sorte à avoir des poids plus équilibrés ? » Ce phénomène est d'autant plus important quand il existe des valeurs aberrantes. Pour éliminer ce problème, Pitt et Shephard (1999) introduisent le filtre particulaire auxiliaire, dont l'idée est d'augmenter l'existence des « bonnes » particules $\{X^{(i)}\}$ dans le sens des probabilités prédictives. L'APF diffère du filtre SIR, car il renverse l'ordre d'échantillonnage et rééchantillonnage qui est possible lorsque les poids d'importance sont dépendants de X_t (Pitt et Shephard, 1999; Chen, 2003; Arulampalam *et al.*, 2002). La procédure de l'APF comporte deux étapes. À la première étape, on fait évoluer les particules avec de grandes probabilités prédictives. À la deuxième étape, on réévalue les particules et on trace les états augmentés. Ceci est équivalent à faire une proposition qui a une forte probabilité conditionnelle *a priori*, évitant ainsi l'échantillonnage inefficace. L'idée de la variable auxiliaire peut être utilisée dans les filtres SIS ou SIR.

- on effectue un premier rééchantillonnage, on obtient donc un ensemble de particules redistribuées avec un poids (w_r) et un nouveau vecteur d'état (X_r),
- on initialise (au temps $t - 1$) l'ensemble des particules aux valeurs obtenues au premier rééchantillonnage. Le poids est initialisé au poids après rééchantillonnage ($w_{t-1} = w_r(t - 1)$) et le vecteur est initialisé à l'instant $t - 1$ à la valeur du vecteur à l'instant $t - 1$ après rééchantillonnage ($w_{t-1} = w_r(t - 1)$).

Les différentes étapes de réalisation du filtre sont présentées ci-dessous pour tout $t \geq 0$:

- $\forall i \in [1, N]$ calcul de $\mu_t^{(i)} = E[p(X_t|X_{t-1}^{(i)})]$
- $\forall i \in [1, N]$ mise à jour du poids d'importance $w_t^{(i)}$ de la particule i à l'instant t à l'aide de l'équation suivante :

$$w_t^{(i)} = w_{t-1}^{(i)} p(Z_t | \mu_t)$$
- $\forall i \in [1, N]$ normalisation des poids d'importance à l'aide de l'équation II.11
- on rééchantillonne l'échantillon, on génère N nouvelles particules ($X_t^{(i)}$) et ($\xi^{(i)}$)
- $\forall i \in [1, N]$ calcul de $X_t^{(i)} \sim p(X_t^{(i)} | X_{t-1}^{(i)}, \xi^{(i)})$ et mise à jour du poids suivant la seconde étape.

$$w_t^{(i)} \propto \frac{p(Z_t | X_t^{(i)})}{p(Z_t | \mu_t^{(\xi=i)})}$$
- calcul du filtre \hat{X}_t en fonction des particules $X_t^{(i)}$ et de leur poids $w_t^{(i)}$ respectif suivant l'expression II.10

Grâce à cette technique d'échantillonnage préalable selon les poids, Pitt et Shephard (1999) espèrent contraindre le déploiement des particules dans les régions de l'espace d'état.

Une comparaison entre les filtres APF et SIR sur l'efficacité statistique dans le contexte de la mesure aléatoire a été réalisée par Chen (2003). Pitt et Shephard (1999) ont montré que lorsque la probabilité ne varie pas au cours du temps, alors la variance de l'APF est inférieure à celle du filtre SIR. La méthode APF peut être vue comme une première étape avant le filtrage. La particule $X_{t-1}^{(i)}$ se propage à travers une variable auxiliaire $\xi_t^{(i)}$ à l'étape de temps suivante afin d'améliorer l'échantillonnage de la partie postérieure. D'autre part, l'APF rééchantillonne $p(X_{t-1} | Z_{0:t})$ au lieu de $p(X_t | Z_{0:t})$ utilisée dans SIR, par conséquent, il réalise la plus faible variance car l'estimation antérieure est plus fiable. Avec ce principe l'APF gagne en précision sur les informations issues du modèle de vraisemblance et évite l'échantillonnage inefficace parce que les particules avec une faible probabilité sont réputées moins informatives, en d'autres termes, les particules à échantillonner sont poussées vers la région de forte probabilité. Mais comme la probabilité conditionnelle n'est pas insensible à l'état, la différence entre l'APF et le filtre SIR reste insignifiante. Cette technique a fait ses preuves et améliore le comportement global de l'algorithme. Bien que comprenant des étapes de calculs supplémentaires, le besoin en rééchantillonnage est fortement réduit, ce qui est le plus intéressant selon Klein (2006).

II.8.5 Filtre de Rao-Blackwell

Les différentes méthodes de filtrage particulière, précédemment décrites, réalisent l'échantillonnage sur l'ensemble du vecteur d'état ce qui implique que les parties linéaires et non-linéaires du vecteur sont traitées indifféremment. On sait cependant que, dans le cas linéaire, le filtre de Kalman est optimal. Le filtre de Rao-Blackwell prend en compte cet aspect dans le calcul du filtre. En effet, le modèle d'espace d'état va être réécrit (équation II.13) afin de dissocier la partie linéaire de la partie non-linéaire (Chen, 2003; Doucet *et al.*, 2000a; Nordlund, 2000, 2002).

Ainsi le système initial (équation II.3) qui était :

$$\begin{cases} X_t = f_t(X_{t-1}) + v_t \\ Z_t = h_t(X_t) + n_t \end{cases}$$

devient :

$$\begin{cases} X_t^P = f_t^P(X_{t-1}^P) + F_{t-1}^P(X_{t-1}^P)X_{t-1}^K + v_{t-1}^P \\ X_t^K = f_t^K(X_{t-1}^P) + F_{t-1}^K(X_{t-1}^P)X_{t-1}^K + v_{t-1}^K \\ Z_t = h_t(X_t^P) + H_t(X_t^P)X_t^K + n_t \end{cases} \quad (\text{II.13})$$

où P et K dénotent respectivement la partie du vecteur d'état considérée particulière et de Kalman. F et H sont les matrices dynamiques du système qui décrivent le modèle. v^P , v^K et n représentent les bruits de modèle et de mesure.

Dans ce cas, $X_t = [X_t^P \ X_t^K]^T$. Cette méthode a pour but de calculer la fonction de densité de probabilité :

$$p(X_t^P, X_t^K | Z_{0:t}) = p(X_t^K | X_{0:t}^P, Z_{0:t}) p(X_{0:t}^P | Z_{0:t}) \quad (\text{II.14})$$

où $p(X_t^K | X_{0:t}^P, Z_{0:t})$ peut être réécrit ainsi $p(X_t^K, X_{0:t}^P)$.

On traite les deux parties du vecteur X selon leurs caractéristiques, afin d'estimer au mieux la densité de probabilité $p(X|Z)$. Pour le vecteur X^K , la densité de probabilité est donnée par $p(X_t^K | X_t^P, Z_t) = \mathcal{N}(\hat{X}_t^K, P_t^K)$ où la moyenne (\hat{X}^K) et la covariance (P^K) sont calculées par le filtre de Kalman. Pour le vecteur X^P , le calcul est réalisé par un filtre particulière. Le résultat est un algorithme utilisant à la fois les équations de Kalman et les équations des filtres particulières dont voici le détail.

– **Au temps $t = 0$:**

– Initialisation du nuage de N particules par dispersion de celles-ci.

Pour chaque particule du nuage, il existe un filtre de Kalman qui lui est associé.

De ce fait, d'abord, on initialise les N particules $(X_t^{(i)}, Z_t^{(i)})_{i \in [1, N]}$ du filtre particulière ainsi que leur poids $W^{(i)} = 1/N$, puis on initialise les N filtres de Kalman associés à chaque position.

– Calcul du filtre à $t = 0$ en fonction des poids des particules $\hat{X}_0 = \sum_{i=0}^N X_0^{(i)} w_0^{(i)}$

– **Au temps $t > 0$:**

– Évolution des N particules à l'instant t .

$$X_t^{P,(i)} = \mathcal{N} \left(f^P(X_{t-1}^{P,(i)}) + F_{t-1}^P \hat{X}_{t-1}^{K,(i)}, F_{t-1}^P P_{t-1}^K (F_{t-1}^P)^T + G_{t-1}^P Q_{t-1}^P (G_{t-1}^P)^T \right) \quad (\text{II.15})$$

L'expression II.15 est une loi de transition de l'état particulaire $t-1$ à t .

Elle se décompose de la façon suivante :

$$X_t^{P,(i)} = f^P(X_{t-1}^{P,(i)}) + F_{t-1}^P \hat{X}_{t-1}^{K,(i)} + \mathcal{N} \left(0, F_{t-1}^P P_{t-1}^K (F_{t-1}^P)^T + (Q_{t-1}^P)^T \right) \quad \forall i \in [1, N]$$

– Correction des N filtres de Kalman à l'instant t .

$$K_t = P_{t|t-1}^K (F_t^P)^T (F_t^P P_{t|t-1}^K (F_t^P)^T + G_t^P Q_t^P (G_t^P)^T)^{-1}$$

$$P_{t|t}^K = P_{t|t-1}^K - K_t F_t^P P_{t|t-1}^K$$

$$\hat{X}_{t|t}^{K,(i)} = \hat{X}_{t|t-1}^{K,(i)} + K_t \left(X_{t+1}^{P,(i)} - f^P(X_t^{P,(i)}) - F_t^P \hat{X}_{t|t-1}^{K,(i)} \right) \quad \forall i \in [1, N]$$

On remarquera ici que K_t et $P_{t|t}^K$ sont identiques pour les N filtres de Kalman.

– Calcul du poids au temps t et normalisation.

Le calcul est réalisé suivant la loi gaussienne du nouveau vecteur X_t^P calculé.

$$\text{calcul du poids } w_t^{(i)} = p(Z_t | X_t^{(i)}) w_t^{(i)} \quad \forall i \in [1, N]$$

$$\text{normalisation } w_t^{(i)} = \frac{w_t^{(i)}}{\sum_j w_t^{(j)}}$$

– Calcul du filtre Rao-Blackwell à l'instant t en fonction du poids $\hat{X}_t = \sum_{i=0}^N X_t^{(i)} w_t^{(i)}$

– Rééchantillonnage si besoin (consulter partie II.7.3.2).

– Prédiction du filtre de Kalman.

$$P_{t+1|t}^K = (F_t^K - D_t F_t^P) P_{t|t}^K (F_t^K - D_t F_t^P)^{-1} + G_t^K \bar{Q}_t^K (G_t^K)^T$$

$$\hat{X}_{t+1|t}^{K,(i)} = (F_t^K - D_t F_t^P) \hat{X}_{t|t}^{K,(i)} + D_t (X_{t+1}^{P,(i)} - f^P(X_t^{P,(i)})) \quad \forall i \in [1, N]$$

$$\bar{Q}_t^K = Q_t^K - S_t^T (Q_t^P)^{-1} S_t$$

$$D_t = G_t^K S_t^T (Q_t^P)^{-1} ((G_t^P)^T G_t^P)^{-1} (G_t^P)^T$$

II.9 Exemples d'application au suivi de cibles

Le problème de la détection et du suivi d'obstacles dans les transports routiers est né dans les années 1980 avec les grands programmes de recherches (par exemple en Europe, Eureka - Prometheus de janvier 1987 à janvier 1995, Ertico - ITS Europe depuis 1991, IVHS lancé en 1991 aux États Unis et devenu ITS en 1994...). C'est donc à partir de cette époque que l'on commence à s'intéresser au problème de filtrage dans des applications automobiles. C'est naturellement que les chercheurs ont majoritairement testé le plus connu à l'époque, le filtre de Kalman car ayant déjà fait l'objet de nombreuses publications dans l'aérospatiale et la balistique pour la résolution de problèmes similaires d'association, de détection et de suivi (Kalman, 1960; Robert, 1970; Aulay et Denlinger, 1973; Reid, 1979). Le problème du filtrage était alors étudié de façon idéalisée comme un problème linéaire. On cherchait à résoudre les difficultés liées au caractère non-linéaire des systèmes, en amont avec des algorithmes de détection et d'association et en aval avec des algorithmes de décision.

Un véhicule en mouvement doit faire face à différentes contraintes suivant l'environnement où il se trouve. Par exemple, la présence d'immeubles, d'obstacles, de piétons et l'existence de conditions différentes lumineuses influent la technique de suivi mise en place. Nous abordons quelques-unes de ces contraintes ci-après.

C'est à partir du début des années 90, que la détection de piétons est de plus en plus étudiée et prise en compte dans les systèmes d'aide à la conduite. Les articles suivants font référence à ce type de suivi (Heikkilä et Silven, 1999; Masoud et Papanikolopoulos, 2001; Xu et Fujimura, 2002; Bertozzi *et al.*, 2004;

Thombre et Nirmal, 2009; Ge *et al.*, 2009; Sato *et al.*, 2010; Moqqaddem *et al.*, 2011). Dans (Sato *et al.*, 2010), les auteurs cherchent à mettre en place un système d'aide à la conduite permettant de réduire les accidents entre véhicule et piéton. Pour cela, ils mettent en place une méthode de suivi de piétons dans un environnement urbain qui utilise un capteur lidar multicouche. L'image 3D obtenue par le balayage laser est transférée dans un plan 2D, puis dans une grille. Une étape de détection des piétons est nécessaire, car, dans un premier temps, tous les objets présents sont dans l'enregistrement de la scène. Un modèle de boîte de dimension définie par l'objet à suivre, ici un piéton, permet d'éliminer de la grille les objets de type véhicule. Les piétons sont ensuite suivis par un filtre de Kalman associé à une méthode de voisin le plus proche.

À partir des années 2000, certains chercheurs s'attardent sur l'étude de nuit à l'aide de capteurs infrarouges (Xu et Fujimura, 2002; Ge *et al.*, 2009; O'Malley *et al.*, 2011). Lors du suivi de nuit, les véhicules ont une apparence différente de celle de jour en raison des conditions différentes de luminosité. Dans l'article de (O'Malley *et al.*, 2011), les auteurs partent du constat que durant la nuit, les véhicules sont visibles par leur phare. Ils proposent un système de traitement des images vidéo pour détecter et suivre les véhicules de nuit à partir de leur phares. Les régions de forte luminosité sont segmentées à partir des pixels de l'image. Les véhicules sont définis en tant que cible par une analyse des symétries bilatérales de ces régions. Enfin, les problèmes de perspective et de distorsion sont corrigés par une transformation des projections, qui permet d'éliminer les fausses alarmes (provenant des lampadaires).

On observera qu'avec le temps et les évolutions techniques (notamment au niveau des capteurs), le filtre de Kalman (et également ses dérivées) a été utilisé en association avec différentes approches telles que l'utilisation d'une détection des lignes de la route (Tan *et al.*, 2007), l'utilisation de plusieurs capteurs (Darms *et al.*, 2008; Baig *et al.*, 2011), ou encore la prise en compte d'une donnée temps (Julier et Uhlmann, 2005; Huck *et al.*, 2011).

Une fois le problème de la détection et du suivi posé dans le domaine routier, la communauté scientifique s'est intéressée à prendre en compte la non-linéarité des systèmes dans les algorithmes de filtrage. Le filtre EKF a été une première solution apportée. On retrouve ainsi, les mêmes problématiques de détection et suivi d'objets présents sur la route, (Bourassa et Kamoun, 1993; Cheng et Zheng, 2007; Khitwongwattana et Maneewarn, 2008; Kmiolek et Ruichek, 2008; Tango *et al.*, 2008; Xu *et al.*, 2009; Otto *et al.*, 2012) et le problème de détection de lignes et donc des bordures ou bords de route (Chuanjin *et al.*, 2003; Fardi *et al.*, 2003; Eidehall et Gustafsson, 2004; Wijesoma *et al.*, 2004; Eidehall *et al.*, 2005; Tian *et al.*, 2006; Meuter *et al.*, 2009; Wang *et al.*, 2011). Les méthodes de détection et suivi d'objets proposées sont diverses. Elles varient notamment selon le capteur utilisé, le type de route étudiée, la mise en place d'une classification des objets. Dans Xu *et al.* (2009), les auteurs mettent en place une méthode de détection et de suivi de piétons pour estimer la possibilité de collision entre véhicule et piéton. Pour cela, ils utilisent deux séquences d'images issues d'une caméra vidéo. À partir de celles-ci, un module de classification permet de détecter les piétons. Pour chaque piéton détecté, les auteurs vérifient s'il est présent dans l'image précédente auquel cas, ils mettent à jour l'EKF sinon ils initialisent un nouveau filtre EKF pour estimer la position du piéton aux instants suivants. Dans (Eidehall et Gustafsson, 2004), les auteurs combinent le suivi de lignes (réalisé à partir d'un système de vision) et le suivi de véhicules (réalisé à partir des informations d'un radar) pour leur système d'aide à la conduite. Pour cela, ils mettent en place trois modèles d'état basés sur trois approximations : l'omission du paramètre clothoïde, la linéarisation de fonction trigonométrique et la linéarisation de fonction trigonométrique associé à des approximations. À partir de ces trois exemples, les auteurs concluent que sans la fusion de la prédiction de ligne avec le suivi de ligne, le filtre est moins performant qu'avec la fusion. L'utilisation de l'EKF permet de prendre en compte les non-linéarité du système. Pour les auteurs, c'est une alternative à la linéarisation des fonctions trigonométriques.

Développé en 1997, ce n'est qu'à partir de 2008 que le filtre UKF (largement utilisé en ingénierie pour estimer les facteurs et paramètres de l'objet à suivre à partir de données bruitées) est utilisé dans les systèmes d'aide à la conduite (Meuter *et al.*, 2008; Olmeda *et al.*, 2009). Dans ces articles, les auteurs réalisent le suivi de piétons à partir de caméra pour les systèmes d'assistance au conducteur. Dans (Olmeda *et al.*, 2009),

les auteurs suivent chaque piéton indépendamment des autres à l'aide d'une caméra thermique infrarouge. Les piétons sont détectés par extraction et évaluation, dans les images infrarouges de la distribution de la chaleur dans le corps humain. Ainsi, chaque objet ayant une taille et une température appropriées est extrait de l'image par seuillage. L'image binaire alors obtenue est corrélée à un modèle probabiliste qui permet de mettre en place une région d'intérêt autour de l'objet. Le filtre UKF est alors mis en œuvre pour estimer le vecteur position, vitesse selon les directions x et y de l'objet.

Après avoir présenté le filtre de Kalman, l'EKF et l'UKF, nous nous attardons maintenant sur les filtres particuliers apparus dans les années 90. C'est dans les années 2000 où le problème de détection et de suivi d'obstacles sur route est déjà bien connu et formalisé, à la fois pour des systèmes linéaires (résolu par Kalman) que pour des systèmes non-linéaires (jusqu'alors résolu par EKF ou UKF) que les filtres particuliers commencent à être utilisés pour la détection et le suivi de systèmes non-linéaires. La problématique est identique à celle vue pour les filtres de Kalman et ses dérivés. On retrouve ainsi, les mêmes problématiques de détection et suivi d'objets présents sur la route, (Idler *et al.*, 2006; Shao *et al.*, 2007; Arndt *et al.*, 2007; Wang *et al.*, 2009a; Thuy et Leon, 2009; Gidel *et al.*, 2009; Takeuchi *et al.*, 2010) et le problème de détection des signalisations horizontales (Liu *et al.*, 2008; Danescu et Nedeveschi, 2009; Teng *et al.*, 2010; Liu *et al.*, 2010). Les méthodes de détection et suivi d'objet sont cette fois encore diverses et variées. Elles tiennent compte notamment du capteur utilisé, du type de route étudié, de la mise en place d'une classification des objets.

Dans (Idler *et al.*, 2006), les auteurs utilisent une caméra infrarouge située à l'avant du véhicule afin de détecter les phares arrières des véhicules qui le précèdent. L'image obtenue par la caméra est binarisée en fonction d'un seuil, de sorte que l'image résultante soit constituée de groupe de pixels blancs représentant les sources de lumière. Après l'application d'une détection de bord à l'image binaire, les auteurs ont comparé deux à deux les groupes afin de déterminer les centres de véhicules (qui correspondent au point situé au milieu des deux feux). Ces centres permettent de créer une boîte rectangulaire qui est utilisée dans le suivi des véhicules par filtrage particulière.

Dans (Liu *et al.*, 2008), les auteurs présentent une méthode de détection et de suivi des marquages au sol, à partir de caméra. Ils utilisent les lignes de bords pour détecter les marques des différentes voies. Une méthode d'estimation de l'orientation de la route est utilisée pour supprimer les lignes ayant des attributs de marquage au sol impossible. Une méthode de mesure de confiance est utilisée pour choisir les marques des lignes de voie candidates au suivi. Un filtre particulière est alors utilisé pour prédire les valeurs futures des lignes basées sur les observations passées. Les auteurs testent leur méthode dans différentes conditions dégradées et constatent que le taux de détection est élevé et le taux de fausse alarme est quant à lui faible.

L'article de (Khan *et al.*, 2005) ne traite pas des problèmes de détection et de suivi en milieu routier, mais nous le présentons ici car il a permis de mettre en évidence la nécessité d'une connaissance *a priori* du nombre de cibles à traiter. Dans cet article, les auteurs étudient le cas de la détection et du suivi de fourmis. Ils partent du constat que les fourmis ont un comportement imprévisible notamment au niveau de leur trajectoire. Ils s'intéressent ainsi aux interactions entre les différentes cibles qui vont modifier les trajectoires et être source de perte de suivi (si deux cibles sont très proches, les cibles peuvent être fusionnées et il est difficile lors de leur séparation de retrouver leur trajectoire antérieure). Les auteurs étudient le comportement d'un filtre particulière selon que le nombre de fourmis cibles est fixe ou variable. Ils montre que l'augmentation rapide et élevé du nombre de cibles a des conséquences sur les différentes méthodes mises en place. Ils mettent en évidence l'importance d'une connaissance *a priori* du nombre de cibles.

II.10 Conclusion

Dans ce chapitre, nous avons présenté un état de l'art des techniques de détection, association et filtrage qui font l'objet de nombreuses publications. Cette présentation a été faite en suivant l'enchaînement classique des traitements d'un système de détection et de suivi.

On a montré que l'étape de détection, qui fournit une information agrégée de la mesure initiale en sortie du capteur, est source potentielle de pertes d'informations. L'étape de détection est suivie d'une étape d'association temporelle entre un objet juste détecté et un objet précédemment estimé. Cette association est réalisée à partir d'une modélisation encore simplifiée de l'objet, ce qui peut introduire de mauvaises associations (objets dédoublés ou au contraire erreur d'associations). Enfin, les filtres appliqués aux mesures associées ne s'exercent que sur une version abrégée des objets détectés (correspondant à certains paramètres du modèle détecté). On peut donc craindre une perte en précision du suivi.

Dans le chapitre suivant, nous proposons une méthode de détection, association et suivi intégrée qui permet dans l'étape de filtrage de garder, dans son entièreté, la mesure de l'objet par le capteur.

Chapitre III

Approche conjointe de détection et suivi

III.1 Introduction

Dans ce chapitre, nous nous intéressons à la résolution du problème de détection et de suivi d'obstacles sur route, et plus particulièrement de véhicules, à partir des images de distances, plus précisément couples d'informations (angles, distances), d'un capteur actif de type télémètre à balayage.

Le chapitre précédent, a mis en évidence différentes méthodes pour le suivi d'obstacles. Contrairement à la majorité des approches habituelles, nous avons choisi d'exploiter les mesures brutes du capteur lidar dans le processus de suivi. Généralement, la prise en compte des données capteurs pour le suivi d'obstacles passe par une étape de prétraitement. Cette étape permet d'extraire les véhicules présents dans la scène et de délivrer une information sur sa position en coordonnées cartésiennes. Cet aspect a été détaillé dans le paragraphe II.1, sous forme d'architecture fonctionnelle. Il est connu que l'étape de prétraitement peut engendrer des pertes d'informations menant à des cas de non détection ou à des données aberrantes. Par ailleurs, les non-linéarités liées à la transformation polaire-cartésien des mesures lidar ne permettent plus de préserver la statistique des bruits de mesure après traitement. Seule une approche exploitant les données brutes permet de garantir l'optimalité des traitements qui en découlent (association, suivi, etc.). Ces limites ont été mises en évidence dans le paragraphe II.4.

Une autre difficulté importante concerne la nature spatialement distribuée des mesures lidar liées à un objet. C'est généralement cet aspect qui conduit à procéder à une étape de prétraitement des mesures afin de détecter les objets présents dans la scène. La difficulté est alors de pouvoir associer chaque impact à un unique véhicule tout en prenant en compte la variabilité temporelle du nombre d'impacts à lui associer (dépendant de l'orientation, l'éloignement, la présence d'occultation, etc.). Afin de contourner cette difficulté, nous détaillons dans ce chapitre une nouvelle méthode qui repose sur une approche intégrée du problème de détection et suivi d'obstacles.

Dans un premier temps, nous présenterons le problème que nous voulons résoudre (section III.2). Les sections suivantes détailleront l'architecture de traitement conjointe que nous avons mis en place (section III.3, III.4 et III.5). L'approche proposée sera ensuite décrite étape par étape dans la section III.6. Enfin quelques exemples issus de différents simulateurs permettront de comparer la méthode mise en place à une méthode plus classique dans la section III.7.

III.2 Présentation du problème

Nous cherchons à caractériser le mouvement relatif des véhicules visibles par rapport au véhicule porteur du capteur. À cette fin, nous traitons des données issues de scénarios routiers simples tels que celui de l'exemple suivant (figure III.1).

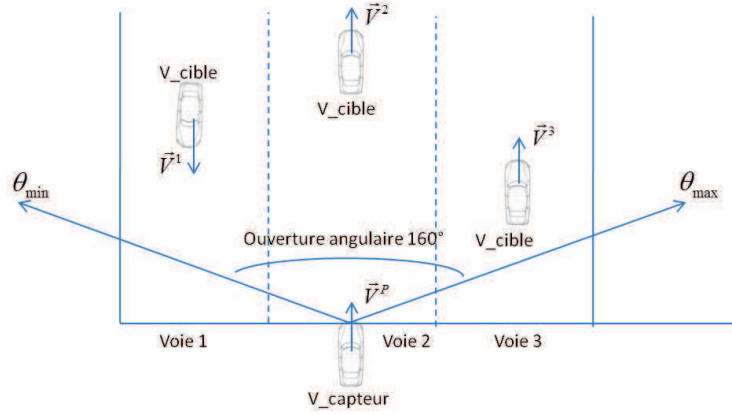


FIGURE III.1. Illustration d'un scénario de détection et de suivi

Soit un scénario comportant un véhicule porteur (désigné $V_{capteur}$) et trois véhicules cibles (V_{cible}). Le véhicule porteur évolue avec une vitesse \vec{V}^P . Les véhicules cibles sont placés dans le champ de vision du capteur, et sont animés d'une vitesse \vec{V}^j relative. On parle de véhicule cible pour tout véhicule situé dans le champ de vision. Le lidar balaye la scène avec une résolution angulaire $\Delta\theta$ et délivre des informations de position sous forme d'impacts repérés en coordonnées polaires (ρ, θ) avec ρ la distance capteur/impact et θ l'angle de tir, comme détaillé sur la figure III.2. Le champ de vision est défini par l'ouverture angulaire ($\theta_{min}, \theta_{max}$) du capteur et sa plage de mesure de distance [ρ_{min}, ρ_{max}].

Dans notre étude, le capteur lidar dispose d'une résolution angulaire de $\Delta\theta = 0,25^\circ$, d'une ouverture angulaire de 160° ($\theta_{min} = -80^\circ, \theta_{max} = 80^\circ$) et d'une plage de mesure de distance ($\rho_{min} = 10\text{m}, \rho_{max} = 180\text{m}$).

La figure III.2 présente un véhicule cible dans le repère capteur dont l'origine est définie par la position du capteur. Le repère capteur est défini comme suit :

- X croît vers la droite du capteur,
- Y croît vers l'avant du véhicule porteur,
- Z croît vers le haut.

Dans notre cas, nous travaillons avec un lidar mononappe, nous n'utiliserons donc pas la composante Z . Les impacts sont repérés en coordonnées polaires comme suit :

- ρ est la distance,
- θ est l'angle de tir du capteur relativement à l'axe y .

Le capteur délivre les mesures en coordonnées polaires, en bleu sur la figure III.2. Le modèle de véhicule quant à lui est représenté par une boîte englobante (en rouge). Les coordonnées polaires sont transformées en coordonnées cartésiennes (X, Y) par les équations :

$$\begin{cases} X = \rho \sin \theta \\ Y = \rho \cos \theta \end{cases}$$

et inversement :

$$\begin{cases} \rho = \sqrt{X^2 + Y^2} \\ \theta = \arctan(X/Y) \end{cases}$$

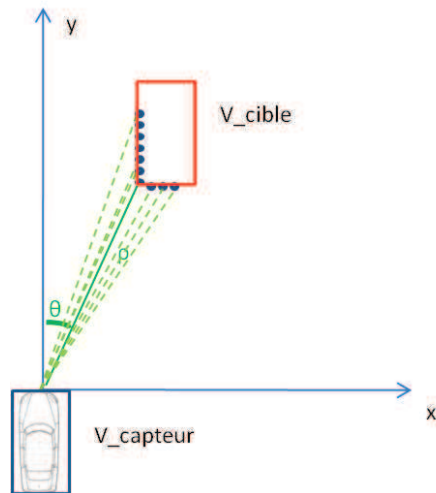


FIGURE III.2. Représentation d'un véhicule dans le repère du capteur

La figure III.3 présente les différentes configurations possibles porteur-cible en termes de mobilité (un ou deux côtés). Cette figure représente également la mesure de la scène tracée en figure III.1. Les véhicules cibles situés sur les voies à droite et à gauche du véhicule porteur ont deux côtés visibles, par contre, le véhicule cible situé dans la même voie que le véhicule porteur n'a d'impacts que sur l'arrière du véhicule. Étant donnée la résolution angulaire constante du capteur, plus la distance est grande entre le capteur et sa cible, moins il y a d'impacts sur le véhicule

Ce caractère non prévisible et très variable du nombre d'impacts selon les configurations relatives entre le capteur et ses cibles rend difficile une estimation précise des positions et des tailles des véhicules cibles.

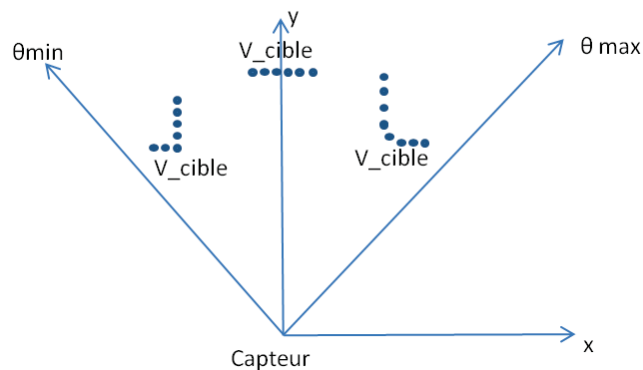


FIGURE III.3. Représentation des impacts lidar pour différentes orientations de véhicules dans le référentiel du capteur

III.3 Architecture fonctionnelle proposée

La figure III.4 illustre l'architecture fonctionnelle de la méthode ITPF (*Integrated tracking [method] based on particle filtering*) proposée. Un module de suivi d'objet DAT (*Detection, Association and Tracking*) met en œuvre un filtre ITPF par objet suivi. Chaque filtre ITPF prend directement en charge les mesures (ρ, θ) en sortie du capteur (i.e. données brutes), mesures restreintes à la zone d'étude de forte probabilité de présence des objets à suivre. À partir de celles-ci, il estime les paramètres dynamiques de chaque objet. Ces filtres reposent sur un traitement conjoint de la détection, de l'association et du suivi des objets, permettant d'éviter les pertes d'informations et les erreurs qu'introduiraient l'exécution d'étapes successives de détection et d'association dans l'algorithme récursif de suivi d'objets. Pour cela, ils mettent chacun en œuvre un filtre particulière réalisant, de façon conjointe, cette détection, association et suivi pour des véhicules connus *a priori*.

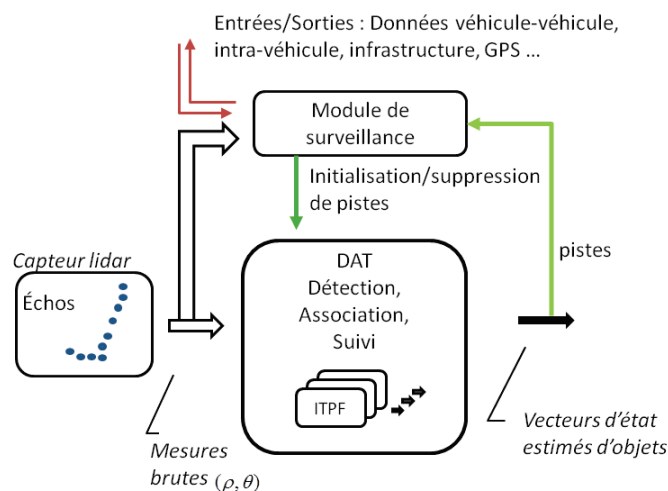


FIGURE III.4. Architecture fonctionnelle de l'approche ITPF proposée

Dans notre contexte de travail (l'environnement routier), on comprend rapidement que la difficulté réside en cette connaissance *a priori* des véhicules qu'il nous faut créer. Cette difficulté est levée par l'ajout d'un module de surveillance chargé de détecter puis gérer les apparitions et disparitions des objets dans le champ de vision du capteur. Ce module fonctionne parallèlement au module DAT de suivi. Il supervise l'initialisation ou la résiliation des pistes du DAT afin de tenir compte aussi bien de l'arrivée d'un nouveau véhicule que de sa disparition définitive.

Comme on peut le constater dans l'exposé de sa structure générale, notre approche ITPF est fondamentalement différente de l'approche habituelle (rappelée dans le paragraphe II.1) qui utilise une étape de prédétection en charge de la segmentation des données reçues. Ici notre module de surveillance effectue bien une détection, mais uniquement dans le but d'obtenir la connaissance *a priori* de l'environnement autour du véhicule porteur. Il interagit ainsi avec le module DAT sans pour autant interférer sur le fonctionnement des filtres ITPF. La modélisation retenue est fortement non-linéaire et les méthodes de Monte Carlo séquentielles sont particulièrement appropriées à la prise en compte de cette non-linéarité.

Nous expliquons en détail, ci-après en section III.4 l'architecture, le rôle et les algorithmes du module de surveillance, de l'approche ITPF. Le module DAT, son rôle et les algorithmes ITPF sont ensuite expliqués respectivement en section III.5 et III.6.

III.4 Module de surveillance

L'architecture fonctionnelle minimale du module de surveillance est donnée en figure III.5. Dans un premier temps, les mesures brutes sont traitées pour chaque balayage (instant t) afin d'en extraire le nombre d'objets présents dans le champ de vision du capteur. Ces objets à suivre sont ensuite comparés aux pistes en cours d'exécution dans le module DAT dans le but d'en créer de nouvelles si nécessaire.



FIGURE III.5. Architecture fonctionnelle simplifiée du module de surveillance

Cette architecture fonctionnelle est ici simplifiée, car elle ne tient pas compte des autres tâches possibles du module de surveillance : la gestion des données reçues sur une ou plusieurs interfaces de communication en provenance d'autres capteurs (GPS, capteurs de détection embarqués sur d'autres véhicules, autres capteurs internes au véhicule), la gestion des données de suivi d'objets transmises via ces mêmes interfaces à destination d'autres organes du véhicule ou de véhicules voisins.

Le paragraphe suivant III.4.1 détaille le détecteur d'objets de ce module de surveillance en s'appuyant sur la littérature existante et le paragraphe III.4.2 présente le fonctionnement du gestionnaire de pistes.

III.4.1 Détecteur d'objets

Le détecteur d'objets du module de surveillance (figure III.6) a pour mission d'extraire les objets à suivre présents dans le champ de vision du capteur et d'estimer la position de leur centre de gravité. Il a en particulier la responsabilité de détecter les objets d'intérêt, pour un contexte applicatif connu *a priori*, dont il présentera les paramètres au gestionnaire des pistes.

Pour cela, le détecteur tient compte des caractéristiques du capteur (ici le lidar), ainsi que de celle du contexte applicatif visé (environnement routier, nature des objets à détecter). Dans notre contexte d'application, un objet doit être suivi lorsqu'il remplit les conditions suivantes : il doit appartenir à une zone d'étude et posséder plus de trois mesures. Dans le paragraphe suivant, nous présentons l'algorithme de détection développé pour le module de surveillance.

L'algorithme de détection de notre module de surveillance a pour objectif d'identifier les objets d'intérêt dans chaque image lidar (image définie comme l'ensemble des mesures z_θ fournies lors d'un balayage) et d'en calculer le centre de gravité. Ces informations sont, à l'issue de l'étape de détection, transmises au module de gestion de pistes du module de surveillance.

Nous avons réalisé un algorithme en quatre points :

- la restriction de la recherche d'objets à une zone d'étude, la zone d'étude étant conçue pour supprimer les impacts qui ne sont pas en adéquation avec l'objet modélisé. Cette zone d'étude est en particulier dictée par l'application traitée. On peut en effet exploiter la connaissance du réseau routier issue d'un système d'informations géographiques (SIG) pour restreindre l'espace à surveiller ;
- le partitionnement des mesures retenues en groupes susceptibles de représenter un même objet. Cette étape permet de partitionner l'espace de surveillance en sous-espaces disjoints contenant un unique objet à suivre ;
- la segmentation des mesures des partitions, précédemment identifiées ;
- le calcul du centre de gravité de chaque objet.

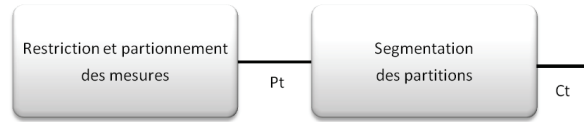


FIGURE III.6. Entités étudiées du détecteur d'objet

Nous présentons ci-après chacun de ces points dans quatre paragraphes distincts que nous compléterons par quelques résultats de détection.

III.4.1.1 Restriction des mesures d'un balayage à une zone d'étude

L'ensemble des mesures z_θ fournies par le lidar à chaque balayage provient d'un nombre et d'un type variés d'objets. Ces objets doivent être sélectionnés selon le contexte applicatif étudié et la modélisation retenue. Pour prendre en compte cet aspect, nous spécifions à l'intérieur de cet ensemble de données des zones de recherche qui permettent de se focaliser *a priori* sur les mesures qui, avec une forte présomption, correspondent aux objets recherchés. La définition de cette zone nécessite la connaissance de la position du capteur par rapport à la voie. Par exemple, nous éliminons d'office via les zones d'études les mesures qui, compte tenu du domaine applicatif et de la nature des objets recherchés, pourront être considérés comme des données aberrantes (*outliers*) vis-à-vis de l'application ciblée. Nous ne gardons que celles qui sont susceptibles, lors de la phase de segmentation, d'être expliquées par nos modèles d'objets (*inliers*).

On définit une **zone d'étude** comme une région de taille déterminée que l'on crée à l'intérieur du champ de vision du capteur afin de restreindre la zone d'étude. Cette restriction aux mesures susceptibles d'être liées à un objet d'intérêt est réalisée en éliminant les mesures considérées comme « *outliers* », sources d'erreur de segmentation et donc de suivi, et qui peuvent engendrer une divergence des filtres mis en œuvre. Les mesures rejetées sont alors considérées comme des données aberrantes vis-à-vis des objets d'intérêt et du contexte d'étude.

Des zones d'étude différentes peuvent ainsi être définies en fonction de la nature de l'objet à suivre, de l'environnement routier dans lequel il évolue et de notre connaissance du comportement du capteur dans ce contexte.

On note $\mathcal{A}_{\text{Objet_Environnement}}$ une zone d'étude spécifique à un type d'objet dans un environnement donné et on la définit ainsi :

$$\mathcal{A}_{\text{Objet_Environnement}} = \{\mathcal{A}_1[\text{attributs}] \cup \dots \cup \mathcal{A}_k[\text{attributs}]\}_{k \geq 1}$$

où \mathcal{A}_k définit une zone particulière de la zone d'étude. Les attributs permettent de caractériser \mathcal{A}_k de manière unique dans son environnement.

Pour notre étude, nous nous plaçons dans un contexte de suivi de véhicules en environnement routier. Les mesures z_θ en provenance des glissières de sécurité ou des arbres peuvent être une gêne à la détection des voitures. De même, il peut être difficile d'affirmer, *a priori* et avec certitude, l'existence d'une automobile qui se trouverait en limite du champ de vision du capteur à partir d'un nombre tronqué de points. L'ensemble de ces éléments est alors considéré comme « *outliers* » vis-à-vis de l'application étudiée. La figure III.7 illustre une telle zone d'étude définie dans ce contexte. L'exemple pris utilise le simulateur SiVIC (Livic, 2011). Le capteur lidar a une ouverture angulaire de 160° degrés, un pas de tir de 0,25 degrés, une vision minimale de 10 mètres, une portée de 180 mètres. Chaque objet situé à une distance supérieure (considérée comme l'infini du capteur) sera également positionné à 180 mètres. La largeur de la zone d'étude est spécifiée afin d'écarter les glissières de sécurité de notre étude. Sa profondeur (de 10 à 120 mètres) permet ici de supprimer les points à l'infini, ceux susceptibles de se combiner avec eux lors de la segmentation et, enfin, des objets trop proches dont certains peuvent être partiellement détectables.

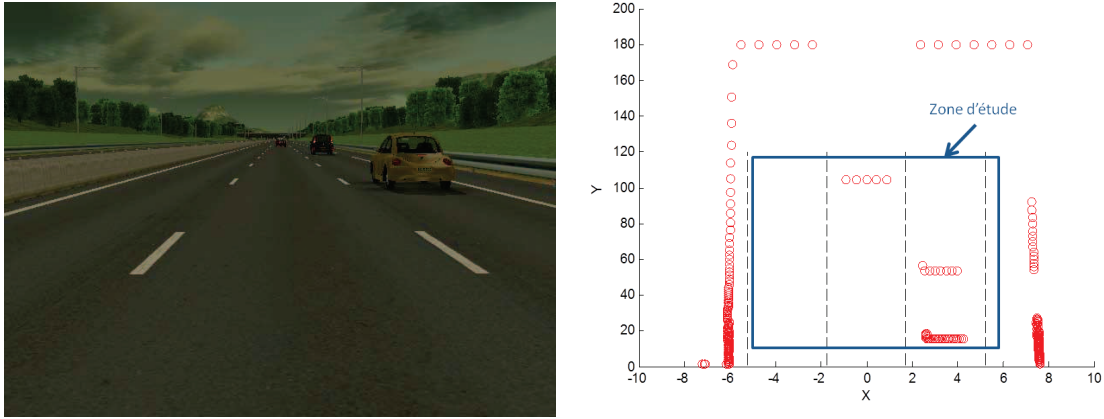


FIGURE III.7. Zone d'étude d'une autoroute à trois voies : la zone est construite de façon à rejeter les points de bordures de routes

III.4.1.2 Algorithme de partitionnement des mesures d'un balayage utilisant le seuil de Santos

Parmi l'ensemble des mesures contenues dans la zone d'étude $\mathcal{A}_{objet_environnement}$, une des difficultés consiste maintenant à les partitionner afin d'associer un ensemble de mesures à un objet unique. Pour cela, nous exploitons une approche basée sur le seuil de Santos précédemment introduit.

L'algorithme résumé ci-après (algorithme 1) cherche à regrouper dans des partitions $\mathcal{P}(n)$ les mesures z_θ de l'ensemble Z_t pour chaque balayage t qui appartient à la zone d'étude $\mathcal{A}_{objet_environnement}$. Une nouvelle partition est créée lorsque la distance euclidienne $d(z_{\theta-}, z_\theta)$ (équation III.1) entre deux mesures successives de la zone d'étude est supérieure au seuil de Santos (équation III.2).

$$d(z_{\theta-}, z_\theta) = \sqrt{\rho_{\theta-}^2 + \rho_\theta^2 - 2(\rho_{\theta-})(\rho_\theta) \cos(\Delta\theta)} \quad (III.1)$$

$$S_{de_Santos} = \frac{\sqrt{2(1 - \cos(\Delta\theta))}}{\cot(\beta) \cos(\Delta\theta/2) - \sin(\Delta\theta/2)} \min\{\rho_{\theta-}, \rho_\theta\} + C_{bruit} \quad (III.2)$$

Algorithm 1 Algorithme de surveillance : partie 1, partitionnement des données

Require: $Z_t = \{z_\theta; \theta \in \{\theta_{min}, \theta_{min} + \Delta\theta, \dots, \theta_{max}\}\}$ l'ensemble des mesures du balayage de l'instant t ;
 $\mathcal{A}_{Voiture_Autoroute}$ la zone d'étude définie afin de détecter des voitures dans un contexte autoroutier avec un lidar aux caractéristiques connues ;

Ensure: N_p le nombre de partition dans l'ensemble des partitions \mathcal{P}_t ; $\{\mathcal{P}(n); \forall n \in [1, N_p]\} \subset Z_t$ l'ensemble des partitions trouvés à l'instant t ; $N_{points}(\mathcal{P}(n))$ le nombre de points de la partition n .

```

1:  $t = 0$ 
2: while Nouveau cycle  $t$  de lecture de mesures lidar do
3:    $\theta^- = \theta_{min}$ 
4:    $N_p = 1$ 
5:   for  $\theta, \theta = (\theta_{min} + \Delta\theta) : \Delta\theta : \theta_{max}$  do ▷ Partitionnement de  $Z_t$ 
6:     if  $z_{\theta^-} \notin \mathcal{A}_{Voiture\_Autoroute}$  then
7:        $z_{\theta^-} = z_\theta$ 
8:     else if  $z_\theta \in \mathcal{A}_{Voiture\_Autoroute}$  then
9:       calculer  $d(z_{\theta^-}, z_\theta)$  selon équation (III.1)
10:      calculer  $S_{de\_Santos}$  selon équation (III.2)
11:      if  $d(z_{\theta^-}, z_\theta) > S_{de\_Santos}$  then ▷ Nouvelle partition
12:         $N_p = N_p + 1$ 
13:         $\mathcal{P}(N_p) \leftarrow z_\theta$ 
14:         $N_{points}(\mathcal{P}(N_p)) = 1$ 
15:      else ▷ Partition courante
16:         $\mathcal{P}(N_p) \leftarrow z_\theta$ 
17:         $N_{points}(\mathcal{P}(N_p)) = N_{points}(\mathcal{P}(N_p)) + 1$ 
18:      end if
19:       $z_{\theta^-} = z_\theta$ 
20:    end if
21:  end for

```

III.4.1.3 Algorithme de segmentation basé sur la théorie de *Split and Merge*

Dans l'étape de segmentation, sont considérés, comme étant potentiellement issues d'un véhicule, les partitions $\mathcal{P}(n)$ qui contiennent un minimum de trois mesures. Il est en effet difficile, à partir de moins de points, de garantir avec certitude la nature de l'objet, son positionnement, voire son existence (*outlier* possible).

Une méthode directe consisterait à estimer la position du centre de gravité des véhicules directement à partir des partitions et d'un modèle de véhicule prédéfini. Toutefois, l'estimation du centre de gravité est, dans ce cas, incorrecte car ne respectant ni l'orientation, ni la position du véhicule sur la route. Ceci est notamment dû au caractère non prévisible du nombre d'impacts sur le véhicule (consulter le paragraphe III.2 et notamment la figure III.3). Une étape de segmentation des mesures de chaque partition précédemment trouvée est donc nécessaire afin d'extraire des mesures des informations sur les côtés de l'objet. Ces informations couplées au modèle d'objet permettent alors d'estimer son centre de gravité.

Dans le cadre du suivi de véhicules sur route, nous avons choisi de mettre en œuvre un algorithme (*Split and merge*) afin de retrouver, dans chacune des partitions, des segments de droites, un segment représentant un côté d'un véhicule. Dans la mesure où les partitions de mesures représentent déjà chacune un véhicule, la phase de fusion *Merge* de l'algorithme devient inutile. Seule la phase de division de l'algorithme, *Split*, est donc conservée. La description de l'algorithme est la suivante.

Pour chaque partition $\mathcal{P}(n)$, nous traçons une droite entre le premier et le dernier impact ($z_{\theta_{low}}$ et $z_{\theta_{high}}$) de la partition $\mathcal{P}(n)$ (figure III.8). Cette droite notée (d) a alors pour équation :

$$(d) : y = ax + b \text{ avec } a = \frac{y_{\theta low} - y_{\theta high}}{x_{\theta low} - x_{\theta high}} \text{ et } b = y_{\theta low} - ax_{\theta low} \quad (\text{III.3})$$

Pour trouver le point de rupture, s'il existe, nous déterminons la distance orthogonale des impacts de la partition $\mathcal{P}(n)$ à cette droite (d) .

$$d(z_{\theta}, (d)) = \frac{|ax_{\theta} - y_{\theta} + b|}{\sqrt{1 + a^2}} \quad (\text{III.4})$$

Soit $z_{\theta R}, z_{\theta R} \in [z_{\theta low}, z_{\theta high}]$, l'impact pour lequel la distance $d(z_{\theta}, (d))$ est maximale. La valeur $d(z_{\theta R}, (d))$ obtenue au point $z_{\theta R}$ est alors comparée à une distance seuil expérimentalement fixée (S_{fixe}). Si $d(z_{\theta R}, (d))$ est supérieure à cette distance seuil alors l'impact $z_{\theta R}$ est le point de rupture de la droite (d) et la droite (d) est divisée en deux droites, (d_1) et (d_2) , passant par $z_{\theta R}$. Le centre de gravité de la voiture est ensuite calculé à partir de la connaissance des côtés (d_1) et (d_2) de l'objet et du modèle de l'objet. Si la valeur de $d(z_{\theta R}, (d))$ est inférieure à celle de la distance seuil S_{fixe} , on se trouve alors dans la configuration V2 de la figure III.3 et le calcul du centre de gravité de la voiture est réalisé à partir de la connaissance d'un seul côté (d) de l'objet et du modèle de l'objet.

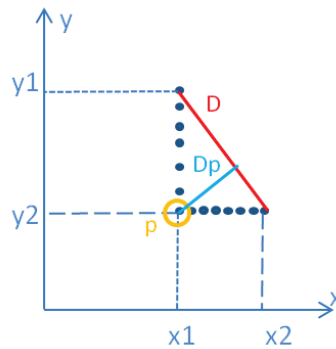


FIGURE III.8. Point de rupture d'un ensemble de mesures dont on cherche les droites d'interpolation

La méthode de segmentation réalisée est résumée par l'algorithme 2 :

Algorithm 2 Algorithme de surveillance : partie 2, segmentation des données (suite de l'algorithme 1)

Require: N_{points_min} le nombre de mesures minimales acceptées par partition; N_p le nombre de partition dans l'ensemble des partitions \mathcal{P}_t ; $\{\mathcal{P}(n); \forall n \in [1, N_p]\} \subset \mathcal{Z}_t$ l'ensemble des partitions trouvés à l'instant t ; $N_{points}(\mathcal{P}(n))$ le nombre de points de la partition n ; S_{fixe} une valeur seuil préalablement déterminée expérimentalement.

Ensure: $\mathcal{C}_t = \{C_{\mathcal{P}(n)}; \forall n \in [1, N_p] \cap (N_{points}(\mathcal{P}(n)) > N_{points_min})\}$, où \mathcal{C}_t est l'ensemble des centres de gravité trouvés dans la zone d'étude à l'instant t dans \mathcal{P}_t .

```

22:    $N_{obj\_de\_c_t} = 0$ 
23:   for  $n, n = 1 : N_p$  do ▷ Segmentation des partitions  $\mathcal{P}(n)$ 
24:       if  $N_{points}(\mathcal{P}(n)) > N_{points\_min}$  then
25:            $z_{\theta low} = \arg \min_{z_{\theta} \in \mathcal{P}(n)} \{\theta\}$ 
26:            $z_{\theta high} = \arg \max_{z_{\theta} \in \mathcal{P}(n)} \{\theta\}$ 
27:           Construire  $(d)$ ;  $\{z_{\theta low}, z_{\theta high}\} \in (d)$  selon équation (III.3)
28:            $z_{\theta R} = \arg \max_{z_{\theta} \in \mathcal{P}(n)} \{d(z_{\theta}, (d))\}$ 
29:           if  $d(z_{\theta R}, (d)) > S_{fixe}$  then ▷  $z_{\theta R}$  point de rupture de  $(d)$ 
30:               Construire  $(d_1)$ ;  $\{z_{\theta low}, z_{\theta R}\} \in (d)$  selon équation (III.3)
31:               Construire  $(d_2)$ ;  $\{z_{\theta R}, z_{\theta high}\} \in (d)$  selon équation (III.3)
32:                $C_{\mathcal{P}(n)} = \text{fonction}_{C_{\mathcal{P}(n)}}((d_1), (d_2), V_{Model})$  selon procédure (4)
33:           else
34:                $C_{\mathcal{P}(n)} = \text{fonction}_{C_{\mathcal{P}(n)}}((d), V_{Model})$  selon procédure (3)
35:           end if
36:            $\mathcal{C}_t \leftarrow C_{\mathcal{P}(n)}$ 
37:            $N_{obj\_de\_c_t} = N_{obj\_de\_c_t} + 1$ 
38:       end if
39:   end for

```

III.4.1.4 Calcul du centre de gravité de la partition

L'algorithme de segmentation a permis de déterminer le nombre de droites présentes dans chaque partition (de une à deux droites). Selon la position relative d'un objet au capteur, la segmentation d'une partition mène à une des cinq représentations (détections) d'un véhicule, illustrées en figure III.9. Ce sont le nombre de segments de droite de la partition et la configuration de l'objet qui conditionnent le calcul du centre de gravité. Deux cas sont alors à distinguer pour ce calcul, soit la partition est composée de deux segments de droites, soit elle n'est composée que d'un segment de droite.

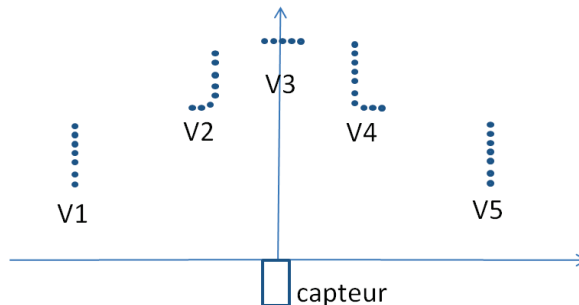


FIGURE III.9. Illustration des cinq configurations possibles d'une partition selon la position relative de l'objet au capteur

Pour notre choix d'un modèle de véhicule rectangulaire, on peut distinguer les cas suivants :

- **La partition est composée d'un unique segment de droite** (le cas du véhicule V3) :

Dans ce cas l'estimation du centre de gravité est donnée par :

$$\begin{cases} X_{cdg} = \text{mean}(X) \\ Y_{cdg} = \min(Y) + l_{modele}/2 \end{cases}$$

- **La partition est composée de deux segments de droites :**

Soit une partition $\mathcal{P}(n)$ composée de deux droites :

$$\begin{cases} (d_1) : X_1 = a_1 Y_1 + b_1 \\ (d_2) : Y_2 = a_2 X_2 + b_2 \end{cases}$$

Les couples (X_1, Y_1) et (X_2, Y_2) représentent respectivement les sous-ensembles d'impacts des droites 1 et 2. L'ordre d'apparition de ces deux droites dans la partition est fonction de la position relative du véhicule au capteur. Ces deux couples peuvent selon les cas être inversés.

L'orientation du véhicule est donnée par le coefficient directeur « a » (équation III.3) de la droite (d) antérieurement calculée entre le premier et le dernier impact de la partition. Cette connaissance de l'orientation du véhicule nous permet d'estimer le centre de gravité de la partition. En effet, elle donne une information de la position en x du centre de gravité (X_{cdg}) relativement au point de rupture. Suivant la configuration, deux cas sont possibles, soit on ajoute, à l'ordonnée à l'origine (b_1) la demi largeur de la droite (L_{Droite}), soit on la soustrait (pour les véhicules V2 et V5 de la figure III.9 par exemple). Par contre, elle est inutile pour la position selon y du centre de gravité (Y_{cdg}), car le côté du véhicule observé est toujours le plus proche du véhicule porteur. La coordonnée Y_{cdg} sera donc toujours positionnée à plus une demi longueur (l_{modele}) du centre du segment de droite observé.

Considérons par exemple des droites (d_1) et (d_2) respectivement parallèles à l'axe des ordonnées et des abscisses, les valeurs de b_1 et b_2 sont alors les suivantes :

$$\begin{cases} b_1 = b_1 - L_{Droite}/2 & \text{si } a < 0 \\ b_1 = b_1 + L_{Droite}/2 & \text{si } a > 0 \\ b_2 = b_2 + l_{modele}/2 \end{cases}$$

Le centre de gravité est alors donné par :

$$\begin{cases} X_{cdg} = \text{mean}(a_1 * Y_1 + b_1) \\ Y_{cdg} = \text{mean}(a_2 * X_2 + b_2) \end{cases}$$

Le centre de gravité estimé est alors transmis au module de gestion de piste du module de surveillance. Les méthodes de calcul du centre de gravité sont résumées dans les algorithmes 3 et 4.

Algorithm 3 Estimation du centre de gravité à partir d'un segment de droite

Require: $\mathcal{P}(n)$ la partition n . (X, Y) les éléments qui la composent forment un segment de droite (d). V_{modele} le modèle de véhicule, $l_{modele} = 3,5\text{m}$ sa longueur et $L_{modele} = 1,8\text{m}$ sa largeur.

Ensure: le centre de gravité $C_{\mathcal{P}(n)}$ du véhicule représenté par ce segment de droite.

- 1: Calculer la largeur du segment de droite : $L_{droite} = |\min(X) - \max(X)|$
 - 2: **if** $L_{droite} < 1$ **then**
 - 3: $L_{droite} = L_{modele}$
 - 4: **end if**
 - 5: Calculer la position en X du centre de gravité : $C_{\mathcal{P}(n)}(X) = \text{mean}(X)$
 - 6: Calculer la position en Y du centre de gravité : $C_{\mathcal{P}(n)}(Y) = \min(Y) + l_{modele}/2$
 - 7: $C_{\mathcal{P}(n)} \leftarrow [C_{\mathcal{P}(n)}(X), C_{\mathcal{P}(n)}(Y)]$
-

Algorithm 4 Estimation du centre de gravité à partir de deux segments de droites

Require: $\mathcal{P}(n)$ la partition n . (X_1, Y_1) et (X_2, Y_2) les éléments qui la composent forment deux droites (d_1) et (d_2) . V_{modele} le modèle de véhicule, $l_{modele} = 3,5\text{m}$ sa longueur et $L_{modele} = 1,8\text{m}$ sa largeur. a le coefficient directeur du segment de droite entre le premier et le dernier impact de la partition.

Ensure: le centre de gravité $C_{\mathcal{P}(n)}$ du véhicule représenté par ces segments de droites.

```

1: if  $|\min(X_1) - \max(X_1)| < 1$  then
2:   Calculer la largeur de la droite :  $L_{droite} = |\min(X_2) - \max(X_2)|$ 
3:   if  $L_{droite} < 1$  then
4:      $L_{droite} = L_{modele}$ 
5:   end if
6:   Estimer l'équation de la droite formée par les éléments de  $(X_1, Y_1)$  :  $Y_1 = a_1X_1 + b_1$ 
7:   Estimer l'équation de la droite formée par les éléments de  $(X_2, Y_2)$  :  $X_2 = a_2Y_2 + b_2$ 
8:   Translater la droite  $Y_1$  sur le centre de gravité :  $b_1 = b_1 + l_{modele}/2$ 
9:   if  $a < 0$  then
10:    Translater la droite  $X_1$  sur le centre de gravité :  $b_2 = b_2 + L_{droite}/2$ 
11:   else
12:    Translater la droite  $X_1$  sur le centre de gravité :  $b_2 = b_2 - L_{droite}/2$ 
13:   end if
14:   Calculer la position en  $X$  du centre de gravité :  $C_{\mathcal{P}(n)}(X) = \text{mean}(a_1X_1 + b_1)$ 
15:   Calculer la position en  $Y$  du centre de gravité :  $C_{\mathcal{P}(n)}(Y) = \text{mean}(a_2Y_2 + b_2)$ 
16: else
17:   Calculer la largeur de la droite :  $L_{droite} = |\min(X_1) - \max(X_1)|$ 
18:   if  $L_{droite} < 1$  then
19:      $L_{droite} = L_{modele}$ 
20:   end if
21:   Estimer l'équation de la droite formée par les éléments de  $(X_1, Y_1)$  :  $X_1 = a_1Y_1 + b_1$ 
22:   Estimer l'équation de la droite formée par les éléments de  $(X_2, Y_2)$  :  $Y_2 = a_2X_2 + b_2$ 
23:   Translater la droite  $Y_2$  sur le centre de gravité :  $b_2 = b_2 + l_{modele}/2$ 
24:   if  $a < 0$  then
25:    Translater la droite  $X_1$  sur le centre de gravité :  $b_1 = b_1 + L_{droite}/2$ 
26:   else
27:    Translater la droite  $X_1$  sur le centre de gravité :  $b_1 = b_1 - L_{droite}/2$ 
28:   end if
29:   Calculer la position en  $X$  du centre de gravité :  $C_{\mathcal{P}(n)}(X) = \text{mean}(a_1Y_1 + b_1)$ 
30:   Calculer la position en  $Y$  du centre de gravité :  $C_{\mathcal{P}(n)}(Y) = \text{mean}(a_2X_2 + b_2)$ 
31: end if
32:  $C_{\mathcal{P}(n)} \leftarrow [C_{\mathcal{P}(n)}(X), C_{\mathcal{P}(n)}(Y)]$ 

```

III.4.1.5 Résultats de détection

Dans ce paragraphe, nous présentons les résultats obtenus sur les données du simulateur SiVIC. Ces données sont complexes, car le rendu est proche de celui qu'on obtiendrait en environnement routier réel (figure III.10 du haut). L'objectif de ce paragraphe est de montrer comment sont détectés les centres de gravité des objets de la scène. Dans un premier temps, les mesures sont regroupées en partitions, celles-ci sont alors étudiées pour en extraire le centre de gravité comme décrit dans la section précédente. La figure III.10 en bas à gauche montre la détection d'une partition et son centre de gravité. Enfin, pour chaque partition trouvée, on estime le centre de gravité, comme le montre la figure III.10 du bas à droite. On remarque ici que les bords de route ne sont pas pris en compte lors de la création des partitions, car éliminés par la définition de la zone d'étude.

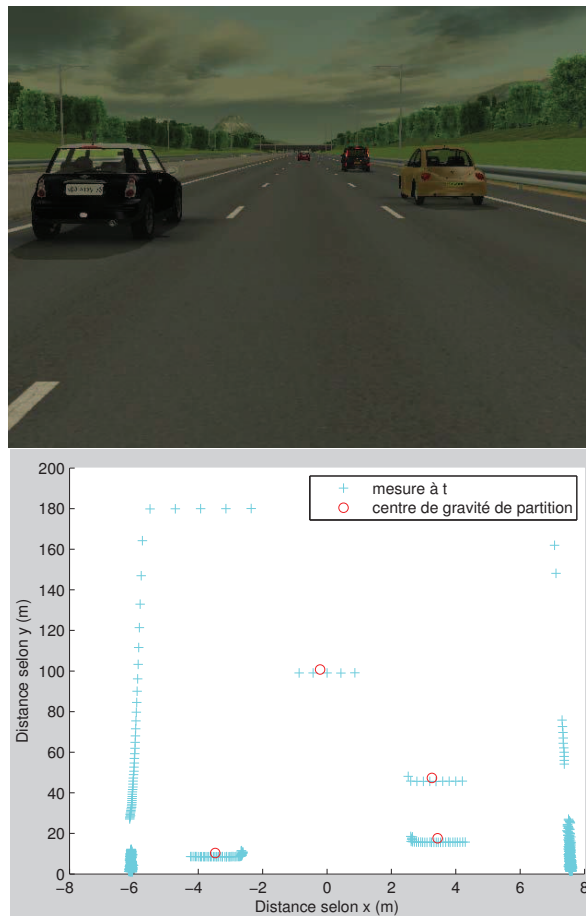


FIGURE III.10. Estimation du centre de gravité des partitions détectées (données SiVIC)

III.4.2 Gestionnaire de pistes

Le gestionnaire de piste a pour rôle de gérer les pistes, c'est-à-dire, d'initier leur création et leur disparition. Il doit être attentif à ne créer qu'une piste par objet et identifier les pistes qui doivent être éliminées. Son rôle de surveillance est indispensable, car il se doit de donner une connaissance *a priori* de l'environnement proche du véhicule porteur du capteur, mais aussi de ne pas saturer le calculateur avec des filtres de suivi inutiles. Le détecteur d'objets précédemment décrit permet de détecter les objets présents dans la scène au sein d'une zone d'étude. Cherchant à détecter des véhicules sur routes sans carrefour, nous avons défini en conséquence cette zone d'étude.

Ainsi, à partir de la comparaison des positions des objets identifiés comme véhicules (détectés) et de la connaissance, en provenance du DAT, des positions estimées des véhicules, le gestionnaire de pistes cherche à vérifier si ces véhicules détectés font déjà l'objet d'un suivi ou non. Dans le cas contraire, le module de surveillance identifie une piste à créer qu'il transmet au module DAT. Cette piste contient la position de l'objet et les paramètres caractérisant cet objet, en particulier celui de sa largeur initiale. Le DAT utilise ces informations pour initier un nouveau filtre ITPF ; il alimente la piste avec les positions estimées de l'objet suivi. Dans le cas où le module de surveillance constate que le véhicule détecté correspond aux paramètres d'une piste déjà connue, il ne fait rien, mais maintient la surveillance de cette piste.

Lors de sa surveillance, le module de segmentation calcule la largeur du véhicule qu'il transmet périodiquement au gestionnaire de piste. Celui-ci observe également en permanence la viabilité des pistes existantes : sont-elles encore mises à jour régulièrement, avec des mesures du capteur, par les filtres ITPF correspondants du DAT ? Lorsque l'une d'elle ne l'est plus, il détruit la piste et libère ainsi le filtre associé.

La figure III.11 illustre ces différents cas de création ou de surveillance de pistes. La méthode proposée est ensuite résumée dans l'algorithme 5 pour notre exemple des véhicules.

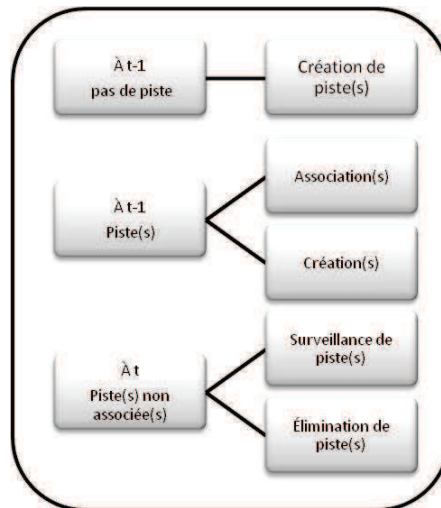


FIGURE III.11. Gestion des pistes d'une famille d'objet

Algorithm 5 Algorithme de surveillance : partie 3, algorithme de gestion des pistes (suite algorithme 2)

Require: $Track$, l'ensemble des pistes du module DAT qui contiennent à leur initialisation les \mathcal{C}_t du module de détection, et par la suite les X_t_Track estimés par les ITPF associés. \mathcal{C}_t l'ensemble des centres de gravité des objets détectés dans la zone d'étude. $N_{obj_de_C_t}$ le nombre d'objets de \mathcal{C}_t . Le paramètre $ITPF_Z_t(m)$ indique si le filtre ITPF associé à $Track(m)$ a bénéficié de mesures ($ITPF_Z_t(m) = ON$ à l'instant $t - 1$, ou non $ITPF_Z_t(m) = OFF$). $Active_Track$ l'ensemble des pistes associées à un véhicule. N_{Track} le nombre de piste actives.

Ensure: ensemble des nouvelles pistes (New_Track) à créer, ensemble des pistes à détruire ($Kill_Track$).

```

40:    $New\_Track = \emptyset$ 
41:    $Active\_Track = \emptyset$ 
42:   if  $Track == \emptyset$  then ▷ Création de pistes
43:     for  $k = 1 : N_{obj\_de\_C_t}$  do
44:        $New\_Track \leftarrow \mathcal{C}_t(k)$ 
45:        $Track \leftarrow \mathcal{C}_t(k)$ 
46:        $Active\_Track \leftarrow Track$ 
47:        $N_{Track} = N_{Track} + 1$ 
48:     end for
49:   else ▷ Recherche piste active
50:     for  $k = 1 : N_{obj\_de\_C_t}$  do
51:       Trouver  $d_{min}(X_t\_Track(m), \mathcal{C}_t(k)) = \arg \min_{m \in [1, N_{Track}]} \{d_{euclidien}(X_t\_Track(m), \mathcal{C}_t(k))\}$ 
52:       if  $d_{min}(X_t\_Track, \mathcal{C}_t(k)) < S_{association}$  then
53:          $Active\_Track \leftarrow Track(m)$ 
54:       else ▷ Création de pistes
55:         if  $N_{Track} < N_{Track\_max}$  then ▷ rmq : perte d'un véhicule à suivre si pas de piste (cf : filtre)
libre
56:            $New\_Track \leftarrow \mathcal{C}_t(k)$ 
57:            $Track \leftarrow \mathcal{C}_t(k)$ 
58:            $N_{Track} = N_{Track} + 1$ 
59:         end if
60:       end if
61:     end for
62:   end if
63:    $Kill\_Track = \emptyset$ 
64:   if  $Track \neq Active\_Track$  then ▷ au moins une piste non associée
65:     for  $m = 1 : N_{Track}$  do
66:       if  $Track(m) \notin Active\_Track$  then
67:         if  $ITPF\_Z_t(m) == OFF$  ou  $Track(m) \notin \mathcal{A}_{Voiture\_Autoroute}$  then
68:            $Kill\_Track \leftarrow Track(m)$ 
69:            $Track = Track - Track(m)$ 
70:            $N_{Track} = N_{Track} - 1$ 
71:         end if
72:       end if
73:     end for
74:   end if
75:    $t = t + \Delta t$ 
76: end while ▷ fin boucle « while » de l'algorithme 1

```

III.4.3 Conclusion

En conclusion, il est important de retenir que le module de surveillance travaille en parallèle du module DAT de détection, association et suivi conjoint. Il traite les mesures (ρ, θ) en sortie du capteur et échange des informations avec le DAT sur les pistes actives ou non. Il les initie en fonction de la nature des objets détectés et de l'environnement. En retour, il utilise les informations du DAT, notamment le dernier positionnement estimé des objets et la fréquence d'utilisation des mesures dans cette estimation afin de décider de maintenir sa surveillance ou d'arrêter un suivi.

III.5 Module de détection, association et suivi DAT

Le module DAT assure le suivi de tout nouvel objet que lui transmet le module de surveillance jusqu'à ce qu'il en reçoive une consigne d'arrêt de suivi, auquel cas il arrête le suivi de l'objet concerné. Ce suivi est assuré en toute indépendance vis-à-vis du module de surveillance, les seules interactions entre le DAT et le module de surveillance sont :

- l'initialisation par le DAT du suivi d'objets à partir des paramètres transmis par le module de surveillance, ces paramètres caractérisent à la fois les objets et leur environnement ;
- l'arrêt de suivi d'objet par le module DAT sur consigne du module de surveillance ;
- la transmission, par le module DAT, d'informations sur l'état des objets suivis au module de surveillance, afin que ce dernier puisse détecter de nouvelles apparitions et disparitions d'objets dans le champ de vision du capteur télémétrique.

Pour assurer ce suivi d'objets, le module DAT est composé d'un ensemble de filtres ITPF de suivi traitant chacun de façon conjointe le problème de la détection, de l'association et du suivi d'objet. Ces filtres ont une structure identique, mais diffèrent les uns des autres dès leur activation en fonction de leurs paramètres d'initialisation. Ces paramètres sont notamment déterminés par la nature de l'objet suivi (véhicule, piéton...), et donc par sa modélisation, son positionnement initial et l'environnement dans lequel il évolue (type de trafic, cf. autoroutier ou routier, départemental, urbain, rue, carrefour, nombre de voies...). Les activations ou désactivations des filtres ne dépendent que des consignes du module de surveillance. Réunis au sein de ce que nous appelons le module DAT, ils sont cependant strictement indépendants les uns des autres. La section III.6 suivante détaille l'approche ITPF qui réalise la détection, l'association et le suivi conjoint.

III.6 Approche conjointe de détection et suivi (ITPF)

Notre approche repose sur un traitement conjoint de la détection et du suivi des objets dont la modélisation est fortement non-linéaire. Elle est bâtie sur la mise en œuvre d'un filtre de Rao-Blackwell et de modèles d'objet adaptés. Le filtre cherche à estimer la position, la vitesse et l'accélération d'un objet particulier en tenant compte de son modèle. Cette modélisation est intégrée au calcul du poids des particules du filtre, point crucial de notre algorithme.

III.6.1 Modélisation d'état du problème

Soit le vecteur d'état $X_t = (x, V_x, a_x, y, V_y, a_y)_t^T$ où (x, y) est la position, (V_x, V_y) est la vitesse et (a_x, a_y) est l'accélération. Il s'agit ici des positions, vitesses et accélérations relatives au véhicule porteur (Vanpoeringhe *et al.*, 2012b). Le problème d'estimation dynamique d'un processus de Markov est généralement décrit par le système d'équations III.5 :

$$\begin{cases} X_t &= f_t(X_{t-1}) + v_t \\ Z_t &= h_t(X_t) + n_t \end{cases} \quad (\text{III.5})$$

avec f_t la fonction de dynamique, h_t la fonction de mesure, v_t le bruit de modèle et n_t le bruit de mesure. $v_t \sim \mathcal{N}(0, Q_t)$ et $n_t \sim \mathcal{N}(0, R_t)$ sont des bruits blancs gaussiens additifs de moyenne nulle. Q_t la matrice

de covariance du bruit de modèle. $R_t = \sigma_p^2 I_n$ la matrice de covariance du bruit de mesure (I_n est la matrice identité de dimension n représentant le nombre de mesure de distance dans une analyse).

III.6.1.1 Équation dynamique

Dans cette section, nous présentons l'équation dynamique du système III.5. Il s'agit d'un modèle cinématique continu (position, vitesse, accélération) :

$$\begin{cases} \dot{X} &= V_x \\ \dot{V}_x &= a_x \\ \dot{a}_x &= 0 \end{cases}$$

Dans notre étude, nous avons choisi d'exprimer les équations du mouvement relatif des véhicules par rapport au véhicule porteur. Comme indiqué dans le schéma général décrivant le problème étudié (paragraphe III.2, figure III.1), les caractéristiques dynamiques \vec{V}_t^S du véhicule porteur sont supposées estimées et délivrées par des capteurs de type GPS, odomètres... :

$$\vec{V}_t^S = \begin{pmatrix} V_t^{S,x} \\ V_t^{S,y} \end{pmatrix}$$

Le système d'équation III.5 peut être réécrit de manière matricielle avec F le flot dynamique du système et H la matrice d'observation du système, cependant dans ce paragraphe on ne s'intéresse qu'à l'équation dynamique. Nous ne détaillerons donc pas l'équation de mesure Z_t décrite dans le paragraphe suivant.

$$F = \begin{pmatrix} F_x & 0_{3 \times 3} \\ 0_{3 \times 3} & F_y \end{pmatrix}$$

où $0_{3 \times 3}$ est une matrice nulle 3×3 et

$$F_x = F_y = \begin{pmatrix} 1 & 0 & 0 \\ \Delta t & 1 & 0 \\ \Delta t^2/2 & \Delta t & 1 \end{pmatrix}$$

avec Δt qui désigne le pas d'échantillonnage.

L'équation matricielle discrétisée est alors :

$$X_t = F X_{t-1} + v_t \quad (\text{III.6})$$

où $v_t \sim \mathcal{N}(0, Q_t)$ un bruit blanc gaussien additif de moyenne nulle.

III.6.1.2 Équation de mesure

Dans cette étude, la difficulté ne réside pas dans la nature de la dynamique du véhicule, mais dans l'utilisation des mesures. En effet, les mesures fournies par le télémètre sont intrinsèquement distribuées. En outre, les mesures se rapportant à un véhicule sont en nombre variable dans le temps et sont dépendantes de la configuration géométrique entre le véhicule suivi et le véhicule porteur du capteur. Pour surmonter cette difficulté, nous intégrons la dynamique du système dans le processus de détection de l'objet. Cela nous permet d'envisager d'intégrer les données reçues à chaque instant, ce qui mènera à une amélioration globale de l'étape de détection et suivi. Cela aura aussi un impact sur les performances du suivi, notamment en termes de probabilité de fausse alarme.

Nous travaillons avec les mesures (ρ, θ) du télémètre et modélisons, sous forme d'équation d'état, les impacts liés au véhicule suivi. L'équation de mesure III.7 associée à la scène est :

$$Z_t = \begin{pmatrix} \rho_{\theta_{min},t} \\ \vdots \\ \rho_{\theta_{max},t} \end{pmatrix} = h_t(X_t) + n_t \quad (\text{III.7})$$

où les termes $\rho_{\theta_j,t}$ représentent les mesures de distance de tout le balayage, c'est-à-dire pour chaque angle θ_j compris entre θ_{min} et θ_{max} et équidistant d'un pas angulaire $\Delta\theta$. n_t est un bruit blanc gaussien de dimension m et de moyenne nulle où $m = (\theta_{max} - \theta_{min})/\Delta\theta$.

III.6.2 Approche proposée

La principale difficulté découle de la forte non-linéarité entre l'état X_t du système III.5, décrivant le mouvement du centre de gravité, et les mesures distribuées Z_t de ce véhicule. La solution proposée est basée sur l'exploitation des filtres particulaires pour leurs capacités à faire face à de telles non-linéarités. Les particules $X^{(i)} \forall i \in \{1, \dots, N\}$ caractérisent N hypothèses dans l'espace d'état. La dynamique du véhicule est décrite par les équations cinématiques sur les composantes positions (x, y) . Par nature, le modèle est partiellement linéaire (pour les composantes de vitesse et d'accélération). Parmi les différentes méthodes de filtrage particulaire, le filtre Rao-Blackwell est particulièrement adapté pour traiter ce modèle d'état. L'équation dynamique (III.5a) est alors séparée en deux parties (une partie non-linéaire et une partie linéaire) et devient III.8 (Vanpoeringhe *et al.*, 2012a) :

$$\begin{cases} X_{t+1}^P &= f^P(X_t^P) + F_t^P X_t^K + v_t^P \\ X_{t+1}^K &= F_t^K X_t^K + v_t^K \end{cases} \quad (\text{III.8})$$

Dans ce contexte, une autre équation de mesure associée à X^K doit être ajoutée au modèle d'état :

$$Z_t^K = X_{t+1}^P - f^P(X_t^P) = F_t^P X_t^K + v_t^P \quad (\text{III.9})$$

Seule la partie non-linéaire (équations III.7 et III.8 a) est traitée par un filtre particulaire, la partie linéaire (équations III.8b et III.9) est résolue par un filtre de Kalman (optimal dans ce cas). Le point central de notre approche est le modèle de mesure qui conduit au calcul des poids des particules.

III.6.2.1 Initialisation

Le vecteur d'état X_t décrit la position, la vitesse et l'accélération du véhicule à l'instant t .

$$X_t = (x, V_x, a_x, y, V_y, a_y)_t^T$$

Nous posons $X_t = (X^P, X^K)_t^T$ le nouveau vecteur d'état où X^P est la partie non-linéaire (cf : position) et X^K est la partie linéaire (cf : vitesse, accélération) de celui-ci. Avec :

$$X_t^P = \begin{pmatrix} x \\ y \end{pmatrix}_t \quad \text{et} \quad X_t^K = \begin{pmatrix} V_x \\ a_x \\ V_y \\ a_y \end{pmatrix}_t$$

Soient les équations de dynamiques suivantes (cf. chapitre II).

$$\begin{cases} X_{t+1}^P &= f^P(X_t^P) + F_t^P X_t^K + v_t^P \\ X_{t+1}^K &= F_t^K X_t^K + v_t^K \end{cases}$$

Le bruit de modèle v_t est un bruit blanc gaussien de moyenne nulle et de variance connue. On pose :

$$v_t = \begin{pmatrix} v_t^P \\ v_t^K \end{pmatrix} \in \mathcal{N} \left(0, \begin{pmatrix} Q_t^P & S_t \\ S_t & Q_t^K \end{pmatrix} \right)$$

S_t étant nul dans notre cas et Q_t^P et Q_t^K les matrices de covariance du bruit de modèle. On a

$$f^P = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad F^P = \begin{pmatrix} \Delta t & \Delta t^2/2 & 0 & 0 \\ 0 & 0 & \Delta t & \Delta t^2/2 \end{pmatrix} \quad F^K = \begin{pmatrix} 1 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Un nuage de N particules est créé à partir de X^P noté $X^{P,(i)}$ ($\forall i \in \{1, \dots, N\}$). Les particules $X^{(i)}$ ($\forall i \in \{1, \dots, N\}$) caractérisent N hypothèses dans l'espace d'état. Pour chaque particule $X^{P,(i)}$, un filtre de Kalman sur la partie linéaire est lancé et est noté $X^{K,(i)}$ ($\forall i \in \{1, \dots, N\}$).

Lors de la phase d'initialisation les dimensions du modèle sont également initiées. Ainsi lors de la création de la piste, la longueur est initialisée *a priori* à la dimension du modèle (fixée à 2,6m), la largeur quand à elle est mémorisée à la valeur transmise par le module de surveillance, si elle est irréaliste (par exemple inférieure à 1m) alors c'est la valeur du modèle fixe (1,8m) qui est prise en compte.

Au cours du suivi, nous cherchons à gagner en précision en affinant dynamiquement la connaissance du modèle utilisé dans le filtre en fonction des observations du module de surveillance, sans pour autant détériorer et perturber la dynamique propre du filtre. Pour cela, nous mettons à jour la largeur du modèle. La largeur est modifiée si et seulement si la largeur transmise par le module de surveillance est plus grande que celle précédemment utilisée. Nous ne permettons pas de diminuer la largeur afin de ne pas perdre d'information lors des cas de masquage, où la largeur visible peut être tronquée.

III.6.2.2 Évolution des particules

Chaque particule (i) peut être vue comme une hypothèse de position dans l'espace d'état, associée à un véhicule cible de la scène. À chaque instant t , les particules évoluent à l'aide de la loi de transition de l'état de l'instant $t-1$ à l'instant t .

$$\begin{aligned} X_t^{P,(i)} &\sim p(X_t^P | X_{t-1}^{P,(i)}, Z_{t-1}) \\ &\sim \mathcal{N}\left(f^P(X_{t-1}^{P,(i)}) + F_{t-1}^P \hat{X}_{t-1|t-1}^{K,(i)}, F_{t-1}^P P_{t-1|t-1}^K (F_{t-1}^P)^T + Q_{t-1}^P\right) \end{aligned}$$

Elle se décompose de la façon suivante :

$$\boxed{X_t^{P,(i)} \sim f^P(X_{t-1}^{P,(i)}) + F_{t-1}^P \hat{X}_{t-1|t-1}^{K,(i)} + \mathcal{N}\left(0, F_{t-1}^P P_{t-1|t-1}^K (F_{t-1}^P)^T + Q_{t-1}^P\right)}$$

$$X_t^{P,(i)} \sim \frac{1}{\sqrt{\det(F_{t-1}^P P_{t-1|t-1}^K (F_{t-1}^P)^T + Q_{t-1}^P) 2\pi}} \exp\left(-\frac{1}{2} (A^T (B)^{-1} A)\right)$$

avec

$$A = Z_t - f^P(X_{t-1}^{P,(i)}) + F_{t-1}^P \hat{X}_{t-1|t-1}^{K,(i)}$$

$$B = F_{t-1}^P P_{t-1|t-1}^K (F_{t-1}^P)^T + Q_{t-1}^P$$

On utilise les équations de correction du filtre de Kalman pour mettre à jour les N filtres de Kalman.

$$\begin{aligned} K_t &= P_{t|t-1}^K (F_t^P)^T (F_t^P P_{t|t-1}^K (F_t^P)^T + Q_t^P)^{-1} \\ P_{t|t}^K &= P_{t|t-1}^K - K_t F_t^P P_{t|t-1}^K \\ \hat{X}_{t|t}^{K,(i)} &= \hat{X}_{t|t-1}^{K,(i)} + K_t (X_{t+1}^{P,(i)} - f^P(X_t^{P,(i)}) - F_t^P \hat{X}_{t|t-1}^{K,(i)}) \end{aligned}$$

III.6.2.3 Pondération des particules : exploitation du modèle d'état

En règle générale, les approches usuelles de détection et suivi de véhicules dans des données lidar reposent sur l'estimation de la position du centre de gravité notamment. Pour cela, une étape de prédétection est nécessaire afin de détecter les objets et d'estimer leur centre de gravité. Dans notre étude, nous exploitons les mesures brutes d'un balayage $Z_t = \{\rho_\theta; \theta \in \{\theta_{min}, \theta + \Delta\theta, \dots, \theta_{max}\}\}$, pour l'ouverture angulaire $[\theta_{min}, \theta_{max}]$. Afin d'utiliser les mesures brutes tout au long de notre étude, notre solution repose sur la construction de l'ensemble des échos $Z_t^{(i)}$, pour chaque particule (i). C'est-à-dire, nous assurons une reconstruction des échos associés à la particule (i) sur l'ouverture angulaire $[\theta_{min}, \theta_{max}]$.

Le schéma de la figure III.12 représente un véhicule particulière reconstruit. Les mesures associées à ce véhicule sont représentées en bleu et le modèle de véhicule utilisé est représenté en rouge sur la figure.

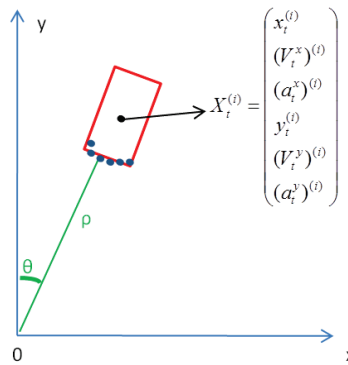


FIGURE III.12. Véhicule particulière reconstruit

L'orientation du véhicule est donnée par la vitesse absolue du véhicule porteur (\vec{V}^S) et la vitesse relative ($V_x^{(i)}, V_y^{(i)}$) du véhicule suivi par rapport au porteur. On pose $\Omega^{(i)}$ l'orientation du véhicule particulière (i).

$$\Omega^{(i)} = \tan^{-1} \left(\frac{V_y^S + V_y^{(i)}}{V_x^S + V_x^{(i)}} \right)$$

Les mesures particulières $Z_t^{(i)}$ sont générées à l'aide du vecteur estimé $X_t^{(i)}$ de la particule (i), de son orientation $\Omega^{(i)}$ et d'un modèle de véhicule prédéfini (voir annexe A.1.4).

$$Z_t^{(i)} = \begin{pmatrix} \rho_{\theta_{min}}^{(i)} \\ \vdots \\ \rho_{\theta_{max}}^{(i)} \end{pmatrix} \quad (\text{III.10})$$

où $[\theta_{min}, \theta_{max}]$ représente la plage sur laquelle la pondération doit être calculée (dans ce cas ouverture angulaire du capteur). Cette plage est identique pour toutes les particules. Et $\rho_{\theta_{min}}^{(i)}$ représente la distance pour l'angle θ_{min} de la particule (i).

Cette procédure permet de construire les échos associés à chaque particule. Cette approche a l'avantage de préserver les statistiques de mesure du bruit, à la différence du traitement de prédétection. Elle permet ainsi de conserver la propriété d'indépendance des bruits mesures, utilisée pour calculer les poids.

Nous pouvons écrire la log-vraisemblance $V(Z_t|X_t^{(i)})$ à partir de l'équation de mesure en utilisant le produit scalaire :

$$\begin{aligned} V(Z_t|X_t^{(i)}) &\propto \|Z_t - Z_t^{(i)}\|_{R_t}^2 \\ &\propto \langle Z_t - Z_t^{(i)}, R_t^{-1}(Z_t - Z_t^{(i)}) \rangle \\ &\propto \|Z_t\|_{R_t}^2 + \|Z_t^{(i)}\|_{R_t}^2 - 2 \langle Z_t, R_t^{-1}Z_t^{(i)} \rangle \end{aligned}$$

Où

- $\|\cdot\|_{R_t}^2$ est équivalent à $(\cdot)^T R_t^{-1}(\cdot)$.
- $\langle \cdot, \cdot \rangle$ désigne le produit scalaire.

On peut réécrire $\langle Z_t, R_t^{-1}Z_t^{(i)} \rangle$ qui représente l'intercorrélacion entre la mesure et la « mesure particulière » :

$$\begin{aligned} \langle Z_t, R_t^{-1}Z_t^{(i)} \rangle &= (Z_t - Z_t^{(i)})^T R_t^{-1}(Z_t - Z_t^{(i)}) \\ &= \frac{1}{\sigma_\rho^2} \sum_{\theta=\theta_{min}}^{\theta_{max}} (\rho_{\theta,t} - \rho_{\theta,t}^{(i)})^2 \end{aligned}$$

où $[\theta_{min}, \theta_{max}]$ représente la plage sur laquelle la pondération doit être calculée. Le produit scalaire peut être ainsi réduit, car l'erreur sur l'angle est généralement supposée négligeable (cf. données constructeur).

III.6.2.3.1 Calcul du poids sur la zone complète

On définit le poids de chacune des particules (i) sur l'intégralité des mesures par III.11 :

$$\boxed{w_t^{(i)} = P(Z_t|Z_t^{(i)})w_{t-1}^{(i)}} \quad (III.11)$$

$$w_t^{(i)} = \frac{w_{t-1}^{(i)}}{\sqrt{\det(R)}2\pi} \exp \frac{-1}{2\sigma_\rho^2} \sum_{\theta=\theta_{min}}^{\theta_{max}} (\rho_{\theta,t} - \rho_{\theta,t}^{(i)})^2$$

Les poids $w_t^{(i)}$ dépendent de la corrélation entre les mesures Z_t du télémètre laser et les échos $Z_t^{(i)}$ reconstruits à partir du modèle d'objet suivi. Avec cette approche, la procédure d'association habituelle est directement intégrée dans la structure du filtre particulière et n'est plus à envisager en tant qu'étape supplémentaire.

Le poids devient alors :

$$\boxed{w_t^{(i)} = \frac{P(Z_t|Z_t^{(i)})}{\sum_{j=1}^N P(Z_t|Z_t^{(j)})} w_{t-1}^{(i)}} \quad (III.12)$$

Il est ensuite normalisé selon l'équation II.11

III.6.2.3.2 Calcul du poids sur une restriction de l'espace d'observation

Les résultats montrent qu'avec cette approche, il y a une attirance entre les différents véhicules présents dans le champ de vision du capteur. Pour pallier ce problème, nous avons mis en place une zone de suivi ou région d'intérêt (*ROI*). Elle correspond à une restriction de l'espace d'état à l'instant t comme le montre la figure III.13. Cette zone est définie par la position des particules à cet instant t , particules précédemment construites à partir de l'ancienne position estimée du véhicule et de sa dynamique.

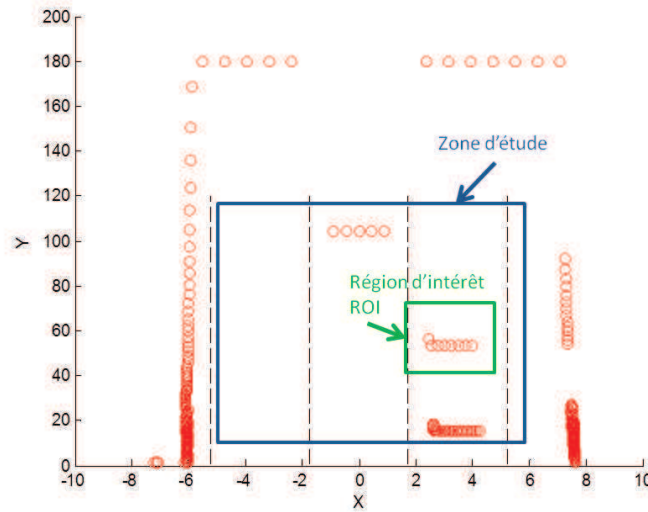


FIGURE III.13. Une région d'intérêt (ROI) de la zone d'étude considérée

Nous construisons cette zone comme une fenêtre parallélépipédique spécifiée, d'une part, par un intervalle angulaire $[\theta_{min}^{ROI}, \theta_{max}^{ROI}]$ et, d'autre part, par un intervalle en distance $[\rho_{min}^{ROI}, \rho_{max}^{ROI}]$. Ces intervalles sont extraits de l'ensemble des échos du nuage de particules. Ainsi, on y cherche le plus petit angle de vue θ_{min}^{ROI} et le plus grand angle de vue θ_{max}^{ROI} , ainsi que la plus petite distance particulaire au capteur ρ_{min}^{ROI} et la plus grande ρ_{max}^{ROI} comme le montre la figure III.14. Les valeurs ρ et θ minimales et maximales de la région ROI sont calculées par III.15 à partir des valeurs minimales et maximales de chacun des véhicules particulières (équations III.13 et III.14) :

$$\begin{cases} \rho_{\theta_{min}}^{min} = \min\{\rho_{\theta_{min}}^{(i)}, i = 1, \dots, N\} \\ \dots \\ \rho_{\theta_{max}}^{min} = \min\{\rho_{\theta_{max}}^{(i)}, i = 1, \dots, N\} \end{cases} \quad (\text{III.13})$$

$$\begin{cases} \rho_{\theta_{min}}^{max} = \max\{\rho_{\theta_{min}}^{(i)}, i = 1, \dots, N\} \\ \dots \\ \rho_{\theta_{max}}^{max} = \max\{\rho_{\theta_{max}}^{(i)}, i = 1, \dots, N\} \end{cases} \quad (\text{III.14})$$

d'où :

$$\begin{cases} (\rho_{min}^{ROI}, \theta_{min}^{ROI}) = \arg \min_{\rho_{\theta_i}^{min}} \{\rho_{\theta_{min}}^{min}, \dots, \rho_{\theta_{max}}^{min}\} \\ (\rho_{max}^{ROI}, \theta_{max}^{ROI}) = \arg \max_{\rho_{\theta_i}^{max}: \rho_{\theta_i}^{max} \neq \rho_{\theta_i}^{min}} \{\rho_{\theta_{min}}^{max}, \dots, \rho_{\theta_{max}}^{max}\} \end{cases} \quad (\text{III.15})$$

On s'accorde une marge d'erreur ε_p (que nous avons fixée ici à 0,5m) sur les distances minimales et maximales de la ROI, afin de tenir compte d'un éventuel biais de la position des particules par rapport aux mesures.

$$\begin{cases} \rho_{min}^{ROI} = \rho_{min}^{ROI} - \varepsilon_p \\ \rho_{max}^{ROI} = \rho_{max}^{ROI} + \varepsilon_p \end{cases}$$

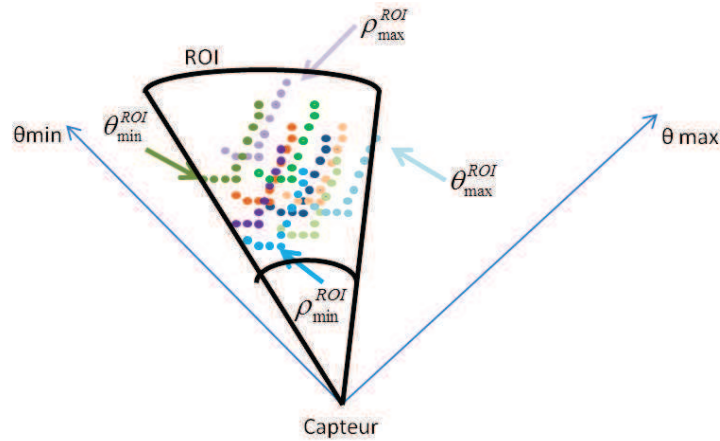


FIGURE III.14. Paramètres $(\theta_{min}^{ROI}, \theta_{max}^{ROI}, \rho_{min}^{ROI}, \rho_{max}^{ROI})$ d'une zone de suivi ou région d'intérêt (ROI)

C'est à l'intérieur de la zone de suivi ROI obtenue que l'on calcule le poids de chacune des particules en fonction de la mesure Z_t et des échos particuliers $Z_t^{(i)}$ s'y trouvant. Pour chaque impact θ de la mesure, tel que $\theta \in \{\theta_{min}^{ROI}, \theta_{min}^{ROI} + \Delta\theta, \dots, \theta_{max}^{ROI}\}$, on vérifie que ρ_θ appartient à l'intervalle de distance $[\rho_{min}^{ROI}, \rho_{max}^{ROI}]$. Si ce n'est pas le cas, la valeur ρ_θ est modifiée, car considérée comme inexistante : elle est mise à la valeur infinie.

Cette méthode donne de bons résultats, le véhicule est bien accroché. Cependant, nous nous sommes aperçus que le calcul du poids est réalisé aussi bien sur des mesures ρ_θ appartenant au véhicule que sur des mesures ρ_θ correspondant à l'infini. Ainsi, une attirance de ces mesures à l'infini peut avoir lieu. Pour remédier à ceci, nous avons décidé de mettre en place une seconde restriction de la région d'intérêt, appelée ROI^+ . La région ROI^+ est créée à partir des mesures brutes appartenant au véhicule. Les mesures brutes du capteur restant dans cette zone ROI^+ sont alors celles de la région ROI allégées de celles placées à l'infini (figure III.15).

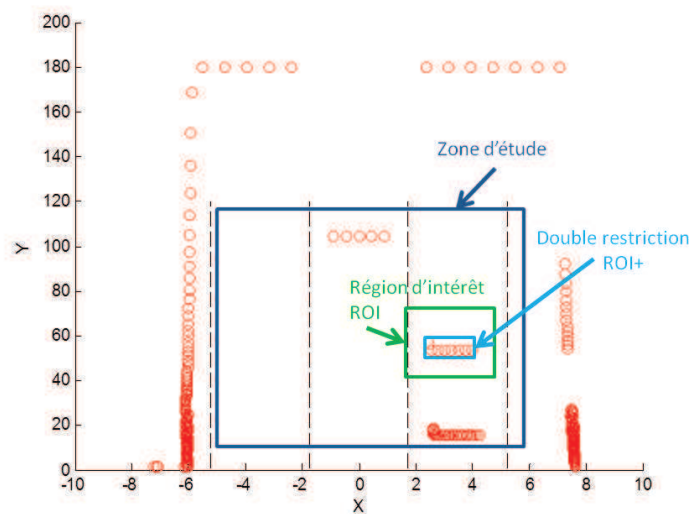


FIGURE III.15. Région ROI^+ d'une région d'intérêt ROI

La nouvelle région de calcul ROI^+ du poids est donc restreinte à :

$$\xi^{ROI^+} = \{\rho_{\theta,t}; \theta \in \{\theta_{min}^{ROI}, \theta_{min}^{ROI} + \Delta\theta, \dots, \theta_{max}^{ROI}\}; \rho_{\theta,t} < \rho_{inf}\} \quad (III.16)$$

Lorsque deux véhicules sont très proches au moment de leur initialisation (à une faible distance longitudinale du véhicule porteur, voire identique, et dans des voies successives), cette méthode présente des difficultés de convergence. En effet, lors de la phase d'initialisation, la vitesse et l'accélération du véhicule sont imparfaitement estimées, car encore en cours de convergence. Ces paramètres influent sur la rotation des véhicules particuliers reconstruits, véhicules utilisés pour la création de la région d'intérêt ROI . De ce fait, même si le nuage est correctement positionné, la rotation des véhicules particuliers tend à étendre cette région ROI à son véhicule voisin. Cette région devient alors trop large et les mesures brutes de deux véhicules à suivre peuvent alors se retrouver dans cette région. La deuxième restriction élimine ensuite les mesures brutes à l'infini et les deux véhicules s'attirent l'un et l'autre. Le calcul du poids devient erroné et le suivi ne converge pas correctement. Cette situation est représentée figure III.16. Pour pallier ce problème, nous imposons, durant les trois premières itérations du filtre, l'utilisation des mesures infini qui permettent de diminuer l'influence des véhicules entre eux. L'accrochage des deux véhicules se fait alors correctement. Par la suite, la convergence ayant eu lieu pour chacun des paramètres, il n'est plus nécessaire de conserver ces mesures à l'infini qui risquent d'être à l'origine d'une attirance à l'infini. La région ROI^+ est alors mise en œuvre.

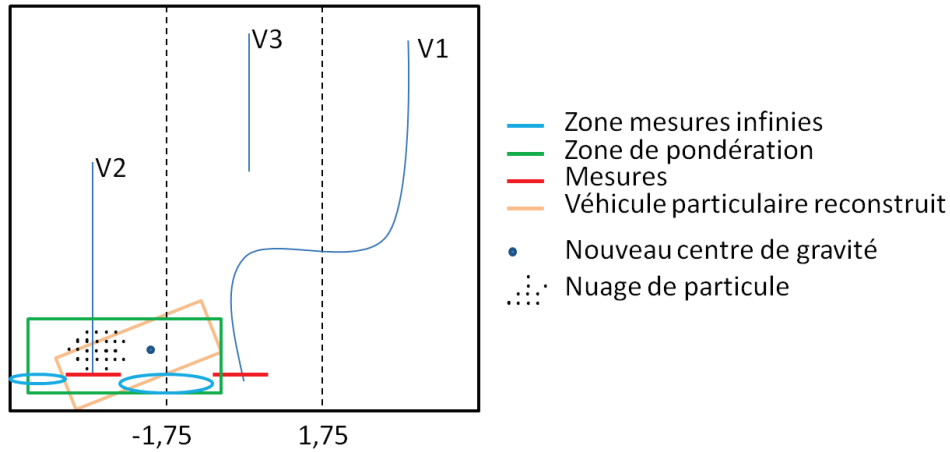


FIGURE III.16. Cas limite de la région ROI^+

Nous obtenons alors le calcul du poids pour les trois premières itérations suivant :

$$w_t^{(i)} = P(\rho_{\theta,t} | \rho_{\theta,t}^{(i)}) w_{t-1}^{(i)} \quad \text{avec } \theta \in \{\theta_{min}^{ROI}, \theta_{min}^{ROI} + \Delta\theta, \dots, \theta_{max}^{ROI}\} \quad (III.17)$$

soit :

$$w_t^{(i)} = \frac{w_{t-1}^{(i)}}{\sqrt{\det(R)} 2\pi} \exp \frac{-1}{2\sigma_\rho^2} \sum_{\theta \in \{\theta_{min}^{ROI}, \theta_{min}^{ROI} + \Delta\theta, \dots, \theta_{max}^{ROI}\}} (\rho_{\theta,t} - \rho_{\theta,t}^{(i)})^2$$

et pour les suivantes :

$$w_t^{(i)} = P(\rho_{\theta,t} | \rho_{\theta,t}^{(i)}) w_{t-1}^{(i)} \quad \text{avec } \theta \in \xi^{ROI^+} \quad (III.18)$$

soit :

$$w_t^{(i)} = \frac{w_{t-1}^{(i)}}{\sqrt{\det(R)} 2\pi} \exp \frac{-1}{2\sigma_p^2} \sum_{\theta \in \theta_{\xi^{ROI^+}}} (\rho_{\theta,t} - \rho_{\theta,t}^{(i)})^2$$

et le poids devient :

$$w_t^{(i)} = \frac{P(\rho_{\theta,t} | \rho_{\theta,t}^{(i)})}{\sum_{j=1}^N P(\rho_{\theta,t} | \rho_{\theta,t}^{(j)})} w_{t-1}^{(i)} \quad (\text{III.19})$$

Il est ensuite normalisé selon l'équation II.11.

En résumé, la mise en place d'une double zone de suivi (ROI et ROI^+) permet de réaliser la procédure d'association entre les « véhicules particuliers » reconstruits et les mesures brutes appartenant à un véhicule. Elle permet d'éviter, d'une part, qu'un autre véhicule proche du véhicule étudié attire à lui les particules (par élimination des échos lui appartenant), puis d'autre part, de créer une attirance à l'infini.

III.6.2.4 Redistribution des particules

Lorsque le nombre de particules efficaces n'est plus suffisant pour garantir une bonne estimation de la trajectoire, il devient nécessaire de redistribuer les particules. Ainsi, si le nombre de particules efficaces est inférieur à un seuil donné, on rééchantillonne la densité de probabilité conditionnelle. Dans ce cas, d'une manière globale, les particules de poids les plus forts sont favorisées par duplication et celles de poids faibles disparaissent.

III.6.2.5 Prédiction de la partie linéaire (Kalman) du filtre de Rao-Blackwell

Les vitesses et accélérations des particules sont prédites à partir de la composante de Kalman du filtre de Rao-Blackwell.

$$\begin{aligned} C_t &= S_t^T (Q_t^P)^{-1} \\ Q_{kk} &= Q_t^K - S_t^T (Q_t^P)^{-1} S_t \\ P_{t+1|t}^K &= (F_t^K - C_t F_t^P) P_{t|t}^K (F_t^K - C_t F_t^P)^T + Q_{kk} \end{aligned}$$

Dans notre cas, C_t est toujours nul puisque S_t , composante de la matrice de covariance du bruit, l'est. Le filtre peut alors se réécrire :

$$\begin{aligned} Q_{kk} &= Q_t^K \\ P_{t+1|t}^K &= F_t^K P_{t|t}^K (F_t^K)^T + Q_t^K \\ \hat{x}_{t+1|t}^{K,(i)} &= F_t^K \hat{x}_{t|t}^{K,(i)} \end{aligned}$$

III.6.2.6 Estimation d'état

L'estimation de l'état à l'instant t est calculée par le système d'équation suivant :

$$\begin{aligned} \hat{X}_t^{P,(i)} &= \sum_{i=1}^N X_t^{P,(i)} w_t^{(i)} \\ \hat{X}_t^{K,(i)} &= \sum_{i=1}^N X_t^{K,(i)} w_t^{(i)} \end{aligned}$$

On obtient l'estimation d'état du filtre $filtrer_t$ suivante :

$$filtrer_t = \begin{pmatrix} \hat{X}_t^{P,(i)}(1) \\ \hat{X}_t^{K,(i)}(1) \\ \hat{X}_t^{K,(i)}(2) \\ \hat{X}_t^{P,(i)}(2) \\ \hat{X}_t^{K,(i)}(3) \\ \hat{X}_t^{K,(i)}(4) \end{pmatrix}_t$$

qui correspond à :

$$filtrer_t = \begin{pmatrix} x_t^{(i)} \\ V_t^{(i)} \\ a_t^{(i)} \\ y_t^{(i)} \\ V_t^{(i)} \\ a_t^{(i)} \end{pmatrix}_t$$

III.7 Résultats

Les résultats présentés dans les sections III.7.1 et III.7.2 ci-dessous ont été obtenus sur des données simulées en provenance de deux simulateurs différents. Les simulations de III.7.1 ont été menées à partir d'un simulateur de trajectoire simplifié développé en début de thèse. Elles ont permis d'évaluer les erreurs d'estimations. Les simulations III.7.2 ont été menées à partir du simulateur SiVIC (Livic, 2011). Ce second simulateur permet d'obtenir des mesures capteurs proches de la réalité.

III.7.1 Cas synthétique de suivi multivéhicule

III.7.1.1 Scénario et caractéristiques du capteur

Dans cet exemple, nous utilisons un simulateur de trajectoire simplifié que nous avons mis en place. Ce simulateur ne prend pas en compte l'environnement extérieur aux voies de la route. En effet, les bords de la route, glissière de sécurité, arbres et autres objets ne sont pas représentés sous forme d'échos, seuls les objets à suivre (dans notre cas des véhicules) sont à l'origine de « mesures ». Le capteur a été caractérisé avec une ouverture angulaire de $(-80^\circ, 80^\circ)$ et une résolution angulaire de $\Delta\theta = 0,25^\circ$. Un objet est visible à partir de 5m et ne l'est plus à partir de 110m, on pose comme distance dite infinie 250m, distance où il n'y a pas d'obstacles détectables. Nous avons simulé des mesures en plaçant des objets de forme rectangulaire dont les dimensions moyennes sont équivalentes à celles d'un véhicule, dans le champ de vision du capteur. Pour chacune des positions simulées, nous ajoutons une erreur sur chacune des mesures d'écart type de 0,03m. Ces données sont résumées dans le tableau III.1. Le simulateur simplifié est utilisé ici afin d'évaluer les performances de l'approche proposée dans un contexte maîtrisé. On s'assure ainsi de pouvoir délivrer une évaluation représentative des performances en estimation de notre méthode.

Les résultats de cette section III.7.1 ont été obtenus avec le scénario suivant. La route est constituée de trois voies de circulation. Le véhicule porteur évolue sur la voie centrale durant tout le scénario (sa vitesse selon x, axe transversal aux voies, est donc nulle). Sa vitesse selon y, axe longitudinal aux voies, est de 15ms^{-1} . La scène est composée de trois véhicules. Dans notre étude, toutes les informations relatives à un véhicule cible sont données relativement au véhicule porteur : position, vitesse et accélération relatives caractérisent la dynamique par rapport au véhicule porteur. Chaque véhicule apparaît à un instant t différent.

	Notation	Valeur
Capteur	(V_x, V_y) en ms^{-1}	(0, 15)
	Num_voie	2
	$(\theta_{min}, \theta_{max})$ en $^{\circ}\text{C}$	(-80, 80)
	(d_{min}, d_{max}) en m	(5, 110)
	$(\Delta\theta, \Delta t)$ en $^{\circ}\text{C}$ et s	(0,25, 0,05)
	tir_infini en m	250

Tableau III.1. Paramètres de simulation du capteur positionné relativement à la route (données du simulateur de la thèse)

Leurs trajectoires sont présentées sur la figure III.17. Enfin, la durée du scénario est fixée à 15s avec un échantillonnage temporel de $\Delta t = 0,05\text{s}$ correspondant à un balayage de la scène (« scan »). Un résumé des paramètres du scénario de simulation est donné dans le tableau III.2.

	Notation	Valeur
Route	Nb_voie	3
	$largeur_voie$ en m	3,5
Cible 1	Num_voie	1
	(V_x, V_y) en ms^{-1}	(0, 0)
	$(a_x, a_y, Bruit_sur_acc)$ en ms^{-2}	(0, 0,3, 0,03)
	$d_vehicule_capteur$ en m	20
Cible 2	Num_voie	2
	(V_x, V_y) en ms^{-1}	(0, 0)
	$(a_x, a_y, Bruit_sur_acc)$ en ms^{-2}	(0, 0,2, 0,03)
	$d_vehicule_capteur$ en m	20
Cible 3	Num_voie	2
	(V_x, V_y) en ms^{-1}	(0, 0)
	$(a_x, a_y, Bruit_sur_acc)$ en ms^{-2}	(0, 0,5, 0,03)
	$d_vehicule_capteur$ en m	35

Tableau III.2. Paramètres de simulation initiaux des véhicules cibles relativement à la route (données du simulateur de la thèse)

Durant la simulation, les véhicules apparaissent à différents instants. Le véhicule situé dans la voie de gauche apparaît à $t \simeq 3\text{s}$ et a une trajectoire rectiligne. Le véhicule dans la voie centrale apparaît à $t = 0\text{s}$, il change de voie au cours de la simulation laissant apparaître un troisième véhicule situé dans la voie centrale, aux environs du temps $t = 7,5\text{s}$.

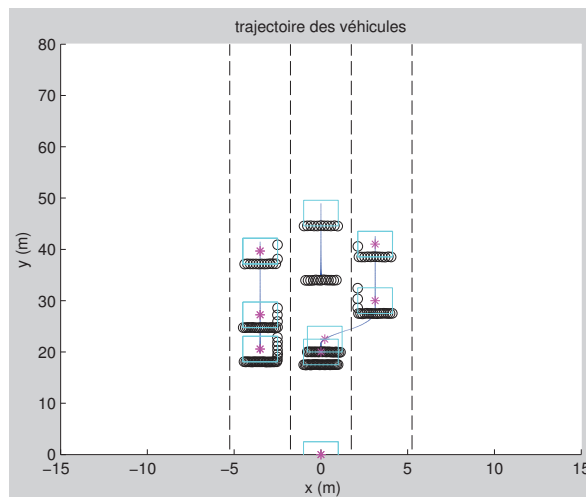


FIGURE III.17. Illustration des trajectoires relatives au véhicule porteur du capteur pour le scénario établi aux tableaux III.1 et III.2

III.7.1.2 Comparaison de l'approche proposée ITPF à une approche classique (CDG) sur les données du simulateur de la thèse

Nous comparons dans cette section notre approche ITPF avec une méthode classique, que nous appelons CDG, basée sur la détection, l'association, puis le suivi du centre de gravité des objets. Dans l'approche ITPF, nous utilisons l'algorithme précédemment expliqué qui comporte deux modules parallèles : le module de surveillance mettant en œuvre les algorithmes de Santos et de *Split and Merge* et le module DAT mettant en œuvre des filtres ITPF dont le poids des particules est calculé à partir de l'ensemble des mesures de l'objet suivi. Pour l'approche classique CDG, nous utiliserons comme algorithme de détection, puis d'association les mêmes algorithmes de Santos et de *Split and Merge*, puis l'algorithme de gestion de piste du module de surveillance. Par contre, le suivi n'est plus réalisé à partir de l'ensemble des mesures des objets. Pour chaque véhicule suivi, un filtre de Rao-Blackwell est mis en œuvre dont les mesures en entrées sont les centres de gravité précédemment détectés et associés par les modules de détection et d'association. Les paramètres de simulation utilisés par ces deux approches ITPF et CDG sont identiques. Les résultats sont présentés en vis-à-vis sur les figures. Ceux de la méthode ITPF se trouvent sur les figures de gauche et ceux de la méthode CDG sur celles de droite.

La figure III.18 présente une comparaison du nombre de pistes actives et du nombre de véhicules réellement présents, en fonction du temps. On vérifie la parfaite reconstruction par les méthodes proposées du nombre de véhicules présents à chaque instant. Ce qui est logique puisque nous utilisons les mêmes algorithmes de détection, d'association et de gestion de piste, que ce soit en parallèle d'un module DAT ou en séquentiel avec un module de filtrage (CDG). Une divergence aurait, cependant, pu être constatée en cas de dérive d'un des filtres de Rao-Blackwell, causant alors, provisoirement le double suivi d'un véhicule.

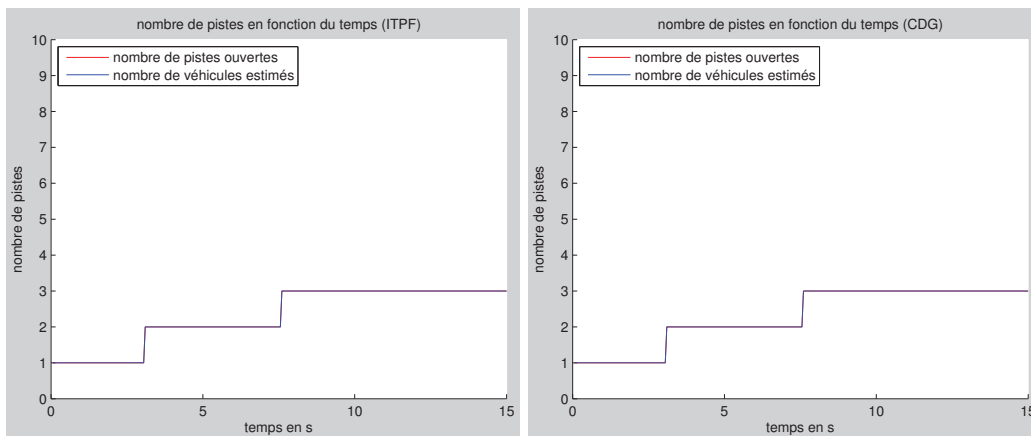


FIGURE III.18. Comparaison du nombre de pistes créées avec le nombre de véhicules pour les méthodes ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse

Les figures III.19 et III.21 suivantes montrent la perception de la scène par le lidar respectivement au temps de 5 secondes puis à celui de 10 secondes. Les figures III.20 sont des agrandissements de chacun des véhicules de la figure III.19 à $t = 5$ s. Celles de III.22, III.23 et III.24 sont des agrandissements de chacun des véhicules de la figure III.21 à $t = 10$ s. Pour chacune de ces figures, les mesures du lidar et la boîte englobante autour de la position estimée du centre de gravité ont été superposés, aussi bien pour les résultats de l'approche ITPF que CDG.

La reconstruction du modèle objet se fait à partir de la position estimée du centre de gravité, de la connaissance *a priori* des dimensions des véhicules (i.e. les largeur et longueur du modèle de véhicule utilisées dans le simulateur) et du calcul de l'orientation des véhicules. Cette reconstruction du modèle objet est représentée sous forme de boîte englobante (rectangle) sur les figures. L'orientation est obtenue à partir de l'estimation de leur vitesse en x et y . Sa précision est donc aussi impactée par l'estimation des accélérations des véhicules, accélération qui entre dans le calcul des vitesses.

On remarque que l'ensemble des véhicules estimés reconstruits par la méthode ITPF (gauche) sont superposés aux mesures, signe que le suivi et la détection sont correctement réalisés. Pour les figures de droites, représentant la détection et le suivi par la méthode CDG, l'orientation et la position des véhicules estimés reconstruits ne collent pas entièrement aux mesures, un décalage (translation, rotation) est parfois observé par rapport aux mesures.

L'analyse des figures III.25 à III.30 relatives à l'estimation des vitesses et accélération, puis aux erreurs sur la position des véhicules permettra de comprendre que les instabilités et imprécisions d'estimation se retrouvent dans ces résultats. On peut déjà constater que l'utilisation des mesures brutes pour la modélisation des véhicules semble permettre à la méthode ITPF de gagner en précision comparativement à l'approche CDG.

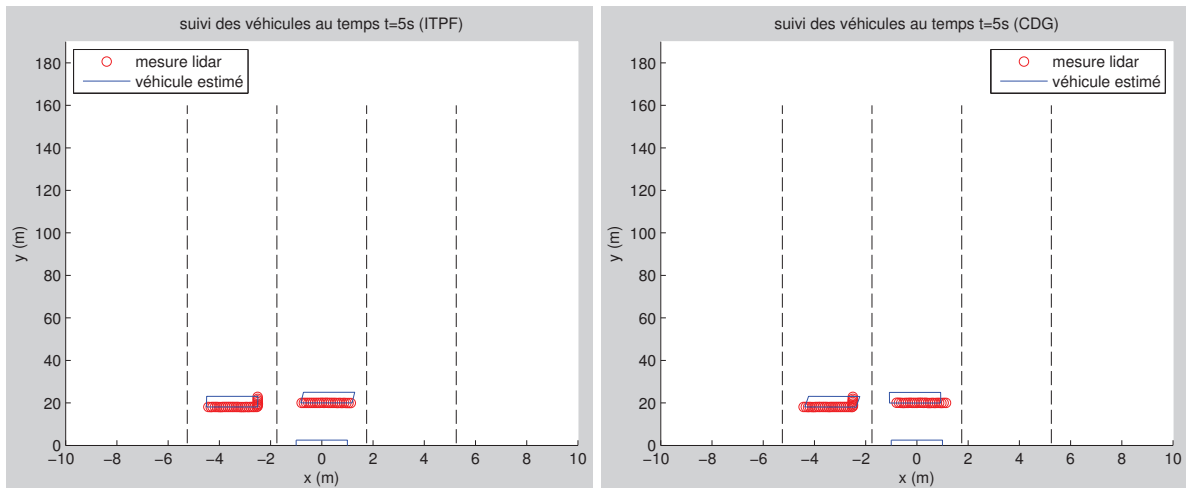


FIGURE III.19. Perception du lidar au temps $t = 5s$ (mesures en rouge) superposée aux modèles estimés des véhicules (rectangle en bleu) : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse

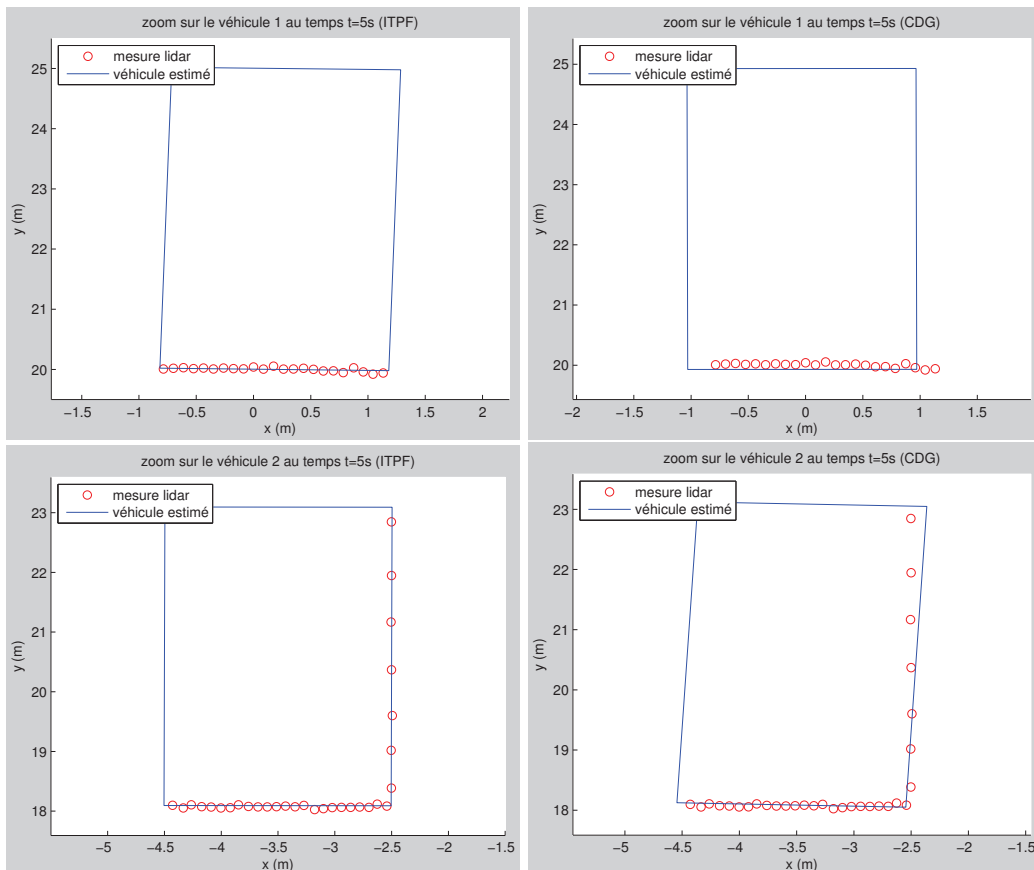


FIGURE III.20. Agrandissement des véhicules de la scène figure III.19, au temps $t = 5s$, véhicule de la voie centrale en haut et de la voie de gauche en bas : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse

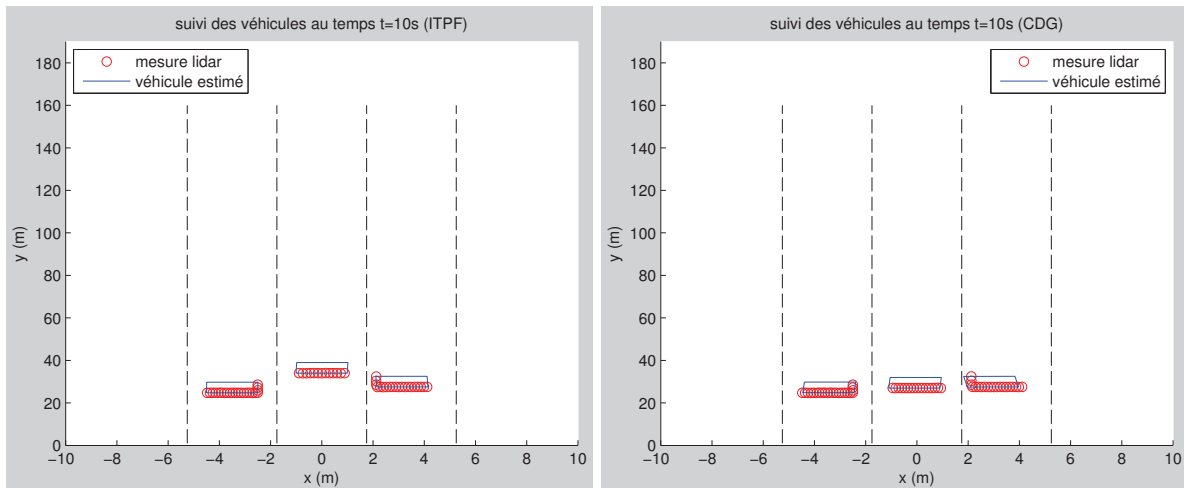


FIGURE III.21. Perception du lidar au temps $t = 10s$ (mesures en rouge) superposée aux modèles estimés des véhicules (rectangle en bleu) : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse

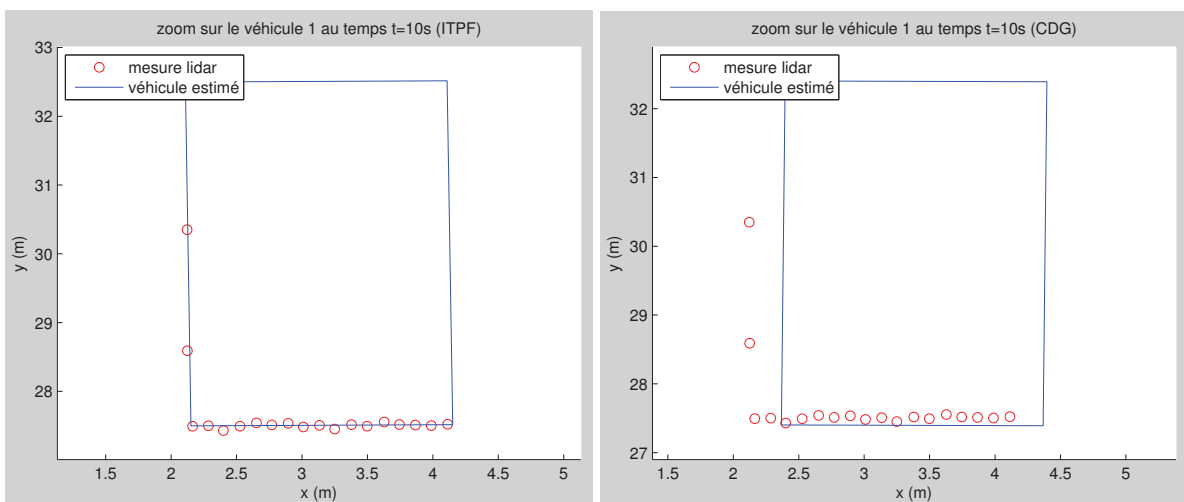


FIGURE III.22. Agrandissement du véhicule de la voie de droite, au temps $t = 10s$, de la figure III.21 : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse

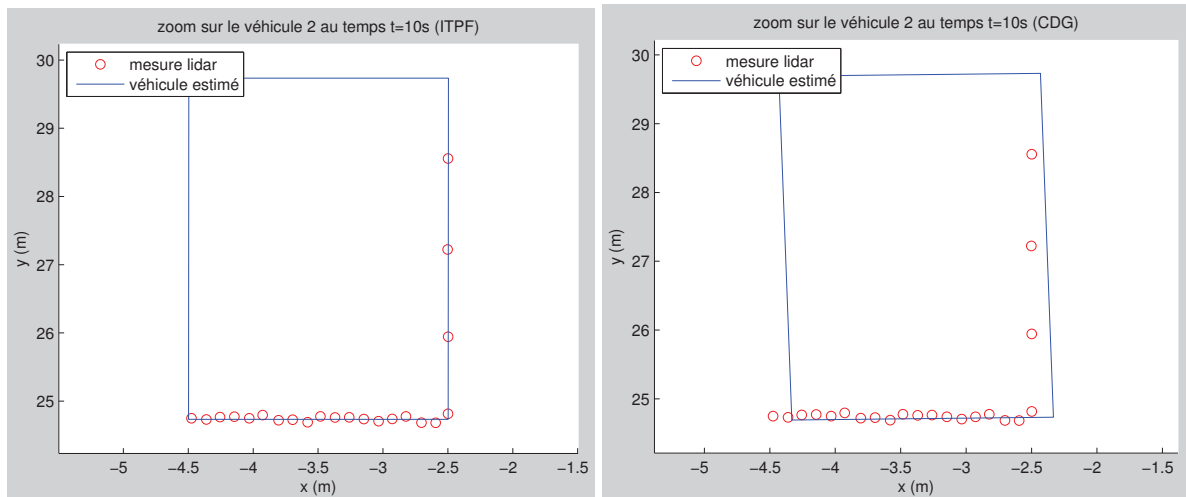


FIGURE III.23. Agrandissement du véhicule de la voie de gauche, au temps $t = 10s$, de la figure III.21 : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse

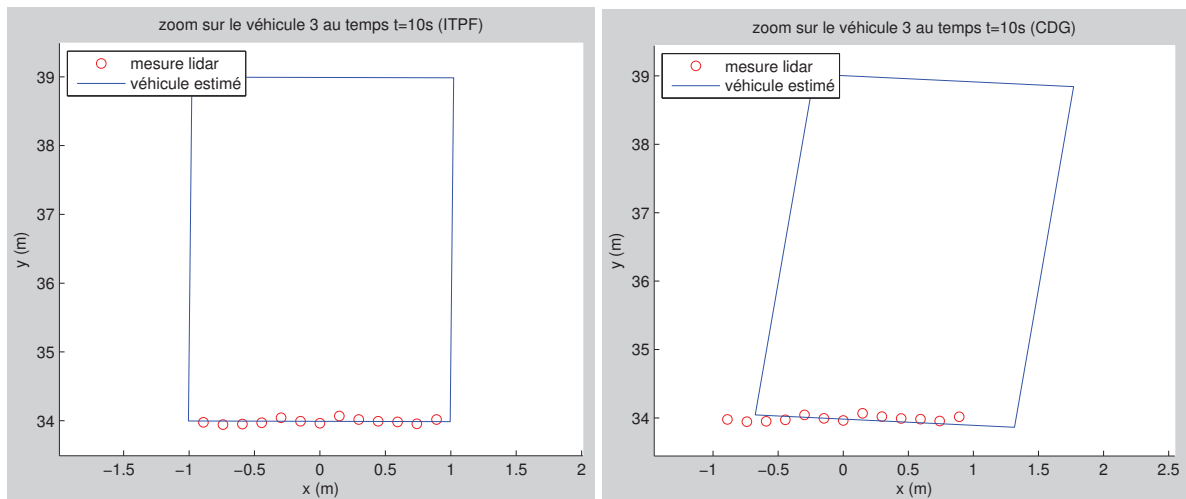


FIGURE III.24. Agrandissement du véhicule de la voie centrale, au temps $t = 10s$, de la figure III.21 : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse

Les figures III.25, III.26 et III.27 présentent les estimations des vitesses (en haut) et accélérations (en bas) de chacun des véhicules, pour notre méthode ITPF à gauche et CDG à droite.

On remarque sur les figures de gauche, de l'approche ITPF, que les estimations de la vitesse et de l'accélération selon les directions x (en bleu) et y (en noir) semblent se superposer aux vitesses et accélérations de référence, et ceci sur la majorité du temps de simulation. Lorsque la vitesse de référence selon x est modifiée (pour atteindre 1ms^{-1}) entre environ $t = 5\text{s}$ et $t = 10\text{s}$, on observe un petit décalage entre la vitesse estimée et celle de référence. On retrouve ce même phénomène avec l'accélération. Le changement d'accélération et de vitesse selon x du véhicule n'est pas instantanément pris en compte par le filtre, cela provoque une perturbation de l'estimé qui après un temps de convergence se repositionne correctement sur les courbes de référence.

Sur les figures de droite, de l'approche CDG, les estimations de la vitesse et de l'accélération selon la direction y (en noir) semblent se superposer aux courbes de référence. Par contre, l'estimation selon la direction x (en bleu) varie autour des courbes de référence. L'estimation est instable et imprécise. Elle ne semble pas converger vers les vitesses et accélérations de référence comparativement à la méthode ITPF.

En résumé, la méthode ITPF se révèle donc bien plus stable que la méthode CDG. Nous avons pu observer que les véhicules reconstruits issus de la méthode ITPF sont mieux accolés aux mesures que ceux de la méthode CDG. En influant sur la rotation du véhicule, la vitesse et l'accélération contribuent à la précision de la position obtenue du véhicule reconstruit par rapport aux mesures. Nous avons également pu observer que la période de convergence est nettement marquée sur les courbes de la méthode ITPF, période après laquelle le filtre est stabilisé autour de la position de référence (la durée de cette période est de $0,5\text{s}$ à 1s selon les simulations ce qui correspond à un nombre de 10 à 20 balayages lidar). Lors de changement brutal de l'accélération en x (changement de voie par exemple), le filtre ne décroche pas même si l'estimation de la vitesse et de l'accélération en x accuse alors un léger retard sur les courbes de référence. On constate en quelque sorte un lissage des observations d'accélération. Pour la méthode CDG, le constat est très différent. Les courbes de vitesse et d'accélération en x fluctuent fortement tout au long de la simulation et on ne peut distinguer de période de convergence sur la durée de la simulation. Les estimations de vitesse et d'accélération de ce filtre résistent donc mal aux variations de mouvements latéraux des véhicules (porteurs et cibles).

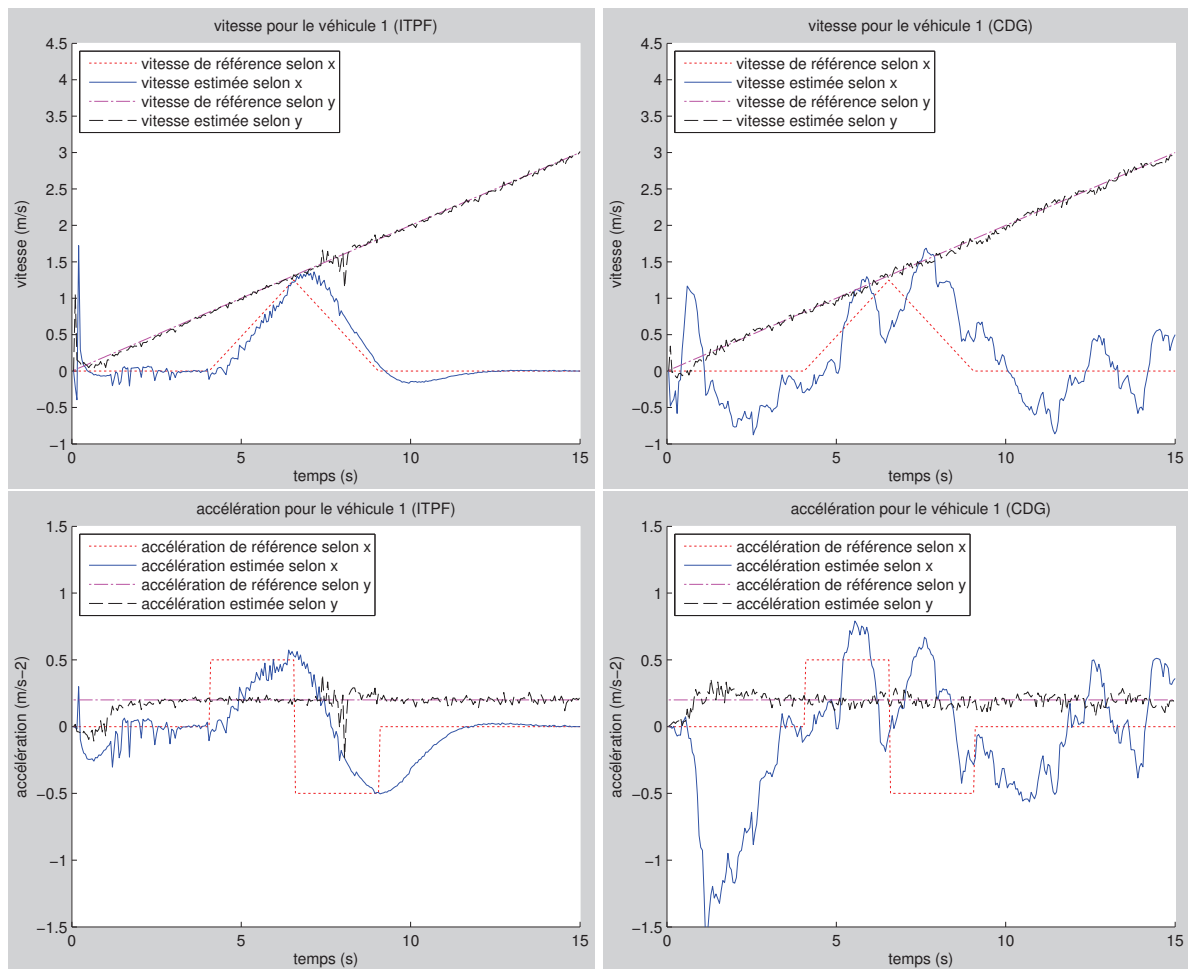


FIGURE III.25. Courbes des Vitesses et accélérations de référence et estimées, du véhicule de la voie de droite en fonction du temps : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse

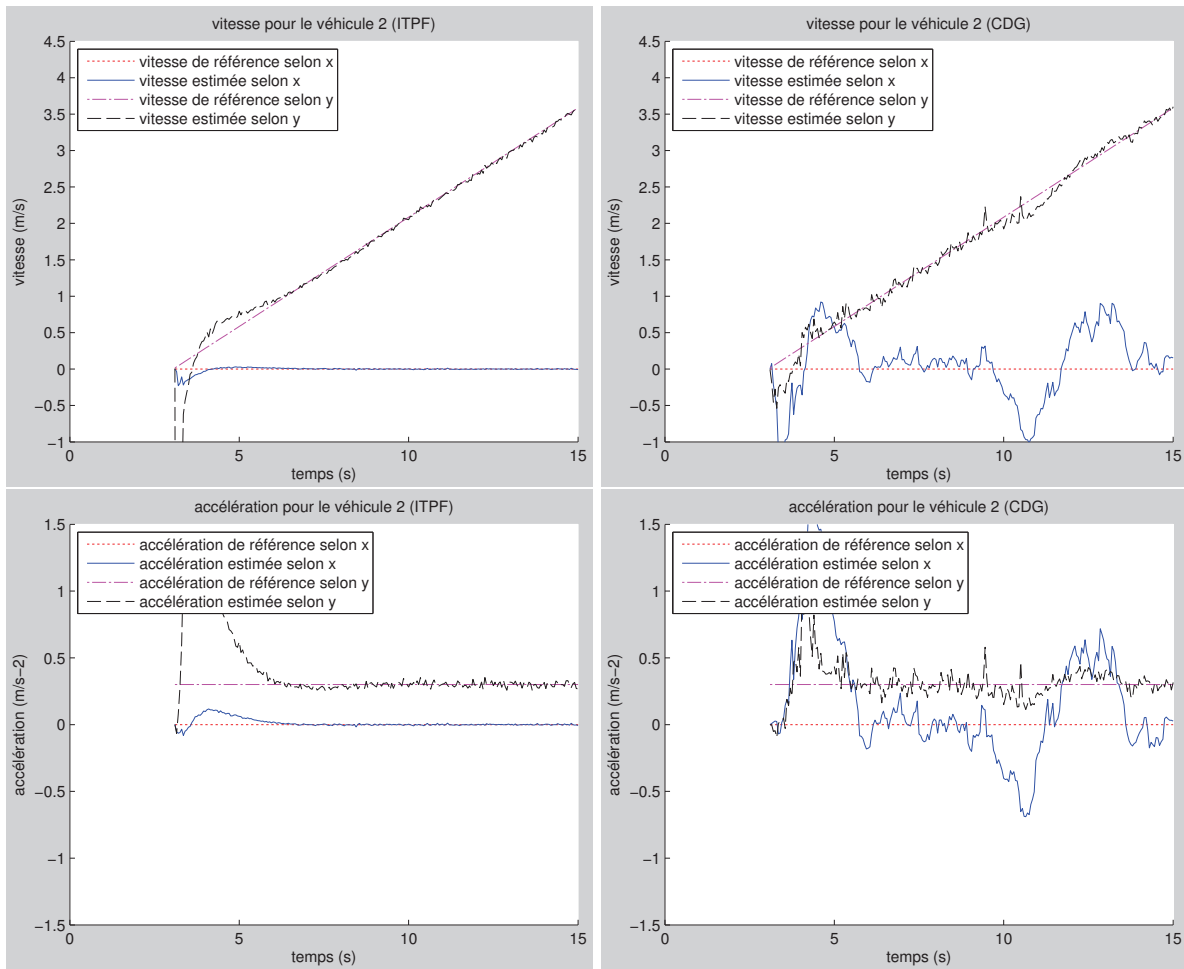


FIGURE III.26. Courbes des Vitesses et accélérations de référence et estimées, du véhicule de la voie de gauche en fonction du temps : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse

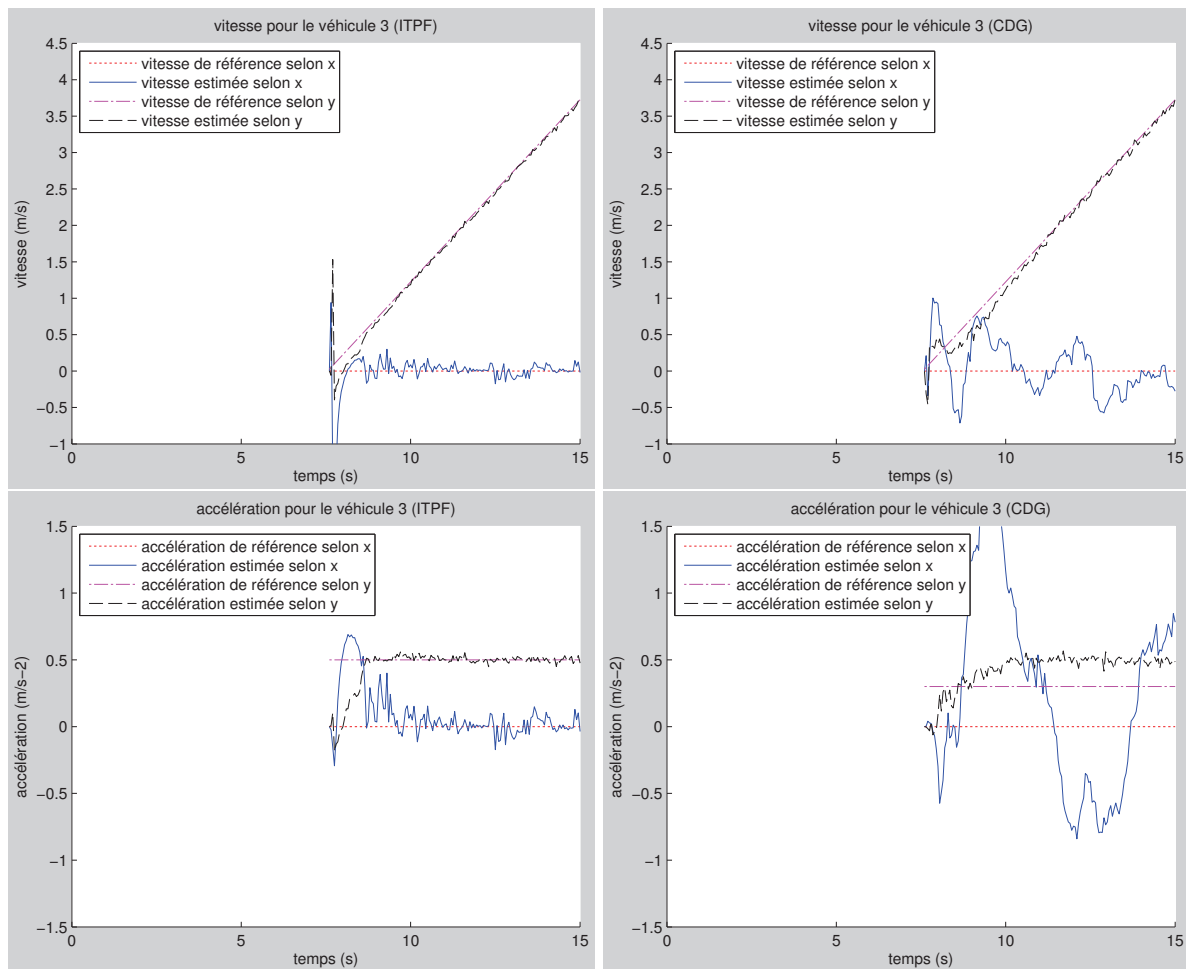


FIGURE III.27. Courbes des Vitesses et accélérations de référence et estimées, du véhicule de la voie centrale en fonction du temps : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse

Nous nous intéressons maintenant aux courbes d'erreur des méthodes ITPF et CDG sur la position (figure III.28 à III.30) des véhicules au cours du temps et aux écarts types des estimations sur les positions, vitesses et accélérations (tableau III.3) calculés sur une période où la convergence est effective.

Pour la méthode ITPF, après l'écoulement de la durée de convergence (marquée par un ou plusieurs pics), l'erreur se stabilise entre 0,1m et $-0,1m$. Pour les figures III.29 et III.30, les véhicules apparaissent respectivement à environ $t = 3s$ et $t = 7,5s$, ce qui explique l'absence d'erreur au préalable et le pic observé au moment de leur détection en tant que véhicule potentiellement à suivre. On constate que l'erreur sur la position reste faible, une fois la période de convergence finalisée. En complément, le tableau III.3 colonne de gauche donne une estimation de l'écart type de l'erreur en position, en vitesse et en accélération calculée hors temps de convergence. Cette estimation est inférieure au bruit de mesure.

Pour la méthode CDG, l'erreur est bornée entre $-0,5m$ et $0,5m$. Comme pour la méthode ITPF, le tableau III.3 colonne de droite donne une estimation de l'écart type de l'erreur en position, en vitesse et en accélération calculée hors temps de convergence. L'écart-type est sensiblement plus élevée que pour la méthode ITPF mais reste faible.

L'étude de ces trois figures et du tableau confirme que la méthode ITPF est plus stable et plus précise que la méthode CDG dite classique.

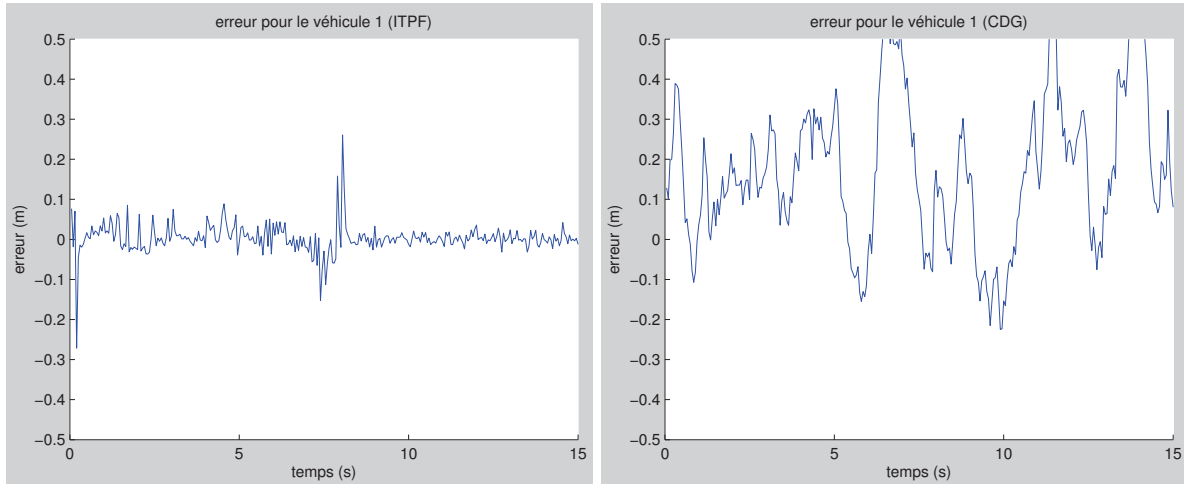


FIGURE III.28. Erreur sur la position du véhicule de la voie de droite en fonction du temps : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse

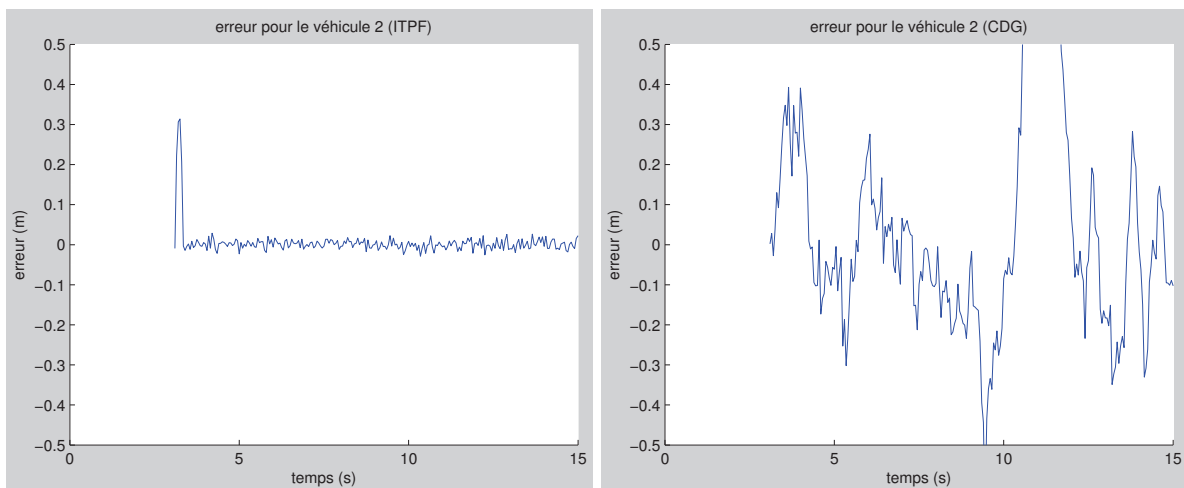


FIGURE III.29. Erreur sur la position du véhicule de la voie de gauche en fonction du temps : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse

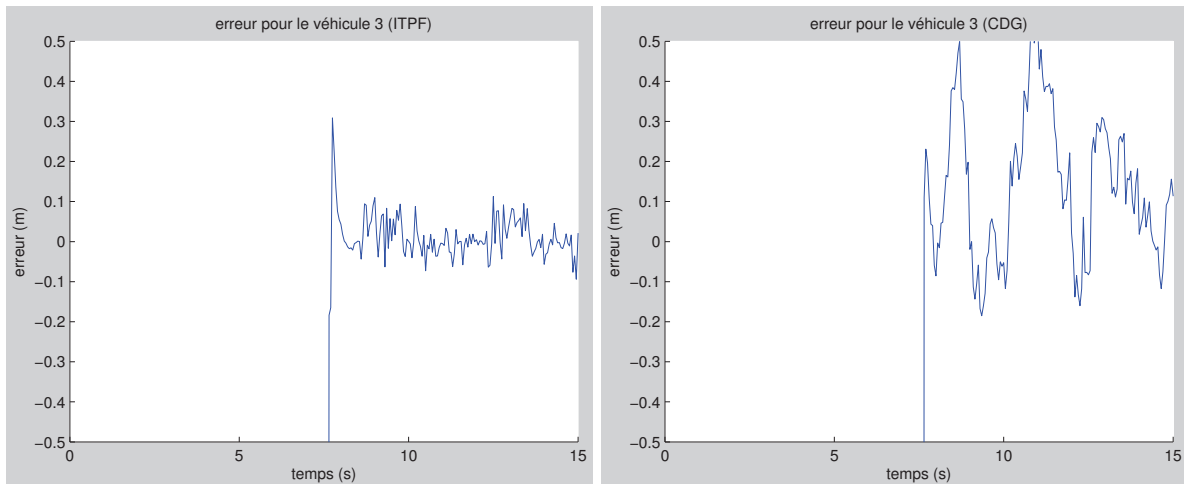


FIGURE III.30. Erreur sur la position du véhicule cible de la voie centrale en fonction du temps : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur de la thèse

Écart type \	Méthode ITPF	Méthode CDG
Position (m)	$1,20 \cdot 10^{-2}$	$3,38 \cdot 10^{-2}$
Vitesse ($\text{m} \cdot \text{s}^{-1}$)	$2,29 \cdot 10^{-2}$	$5,72 \cdot 10^{-2}$
Accélération ($\text{m} \cdot \text{s}^{-2}$)	$2,20 \cdot 10^{-2}$	$9,38 \cdot 10^{-2}$

Tableau III.3. Écart-type de l'erreur d'estimation des positions, vitesses et accélérations pour les véhicules cibles sur une période de convergence de les méthodes ITPF et CDG appliquées aux données du simulateur de la thèse

III.7.2 Résultats obtenus avec les données du logiciel SiVIC

III.7.2.1 Scénario et caractéristiques du capteur

Dans cet exemple, nous utilisons le simulateur de trajectoire SiVIC (Livic, 2011). Ce simulateur présente, pour le capteur lidar notamment, des données de scène routière ayant un comportement très proche de la réalité (statistiques des bruits, comportement dynamique, modélisation des objets ...). Il permet également de prendre en compte l'environnement complet d'un véhicule avec une grande fidélité : bords de route, glissières de sécurité, arbres, piles de pont, éclairages, talus, etc.

Pour notre expérimentation, les caractéristiques du capteur sont définies par une ouverture angulaire de $(-80^\circ, 80^\circ)$ avec une résolution angulaire de $\Delta\theta = 0,25^\circ$. Les plages de visibilité du capteur sont de 10m pour la distance minimale et de 180m pour sa portée maximale. Ces données sont résumées dans le tableau III.4.

	Notation	Valeur
Capteur	(V_x, V_y) en ms^{-1}	(0, 20)
	Num_voie	2
	$(\theta_{min}, \theta_{max})$ en $^{\circ}C$	(-80, 80)
	(d_{min}, d_{max}) (m)	(10, 180)
	$(\Delta\theta, \Delta t)$ en $^{\circ}C$ et s	(0,25, 0,05)
	tir_infini en m	180

Tableau III.4. Paramètres de simulation du capteur positionné relativement à la route (données SiVIC)

On s'intéresse au scénario suivant : la route est constituée de trois voies de circulation. Le véhicule porteur évolue sur la voie centrale durant tout le scénario (sa vitesse selon la direction x est nulle). Sa vitesse selon la direction y est de $20ms^{-1}$. La scène est composée de quatre véhicules. Les trajectoires résultantes des véhicules sont présentées en figure III.31. La durée de la simulation est fixée à 15s avec un échantillonnage temporel de $\Delta t = 0,05s$.

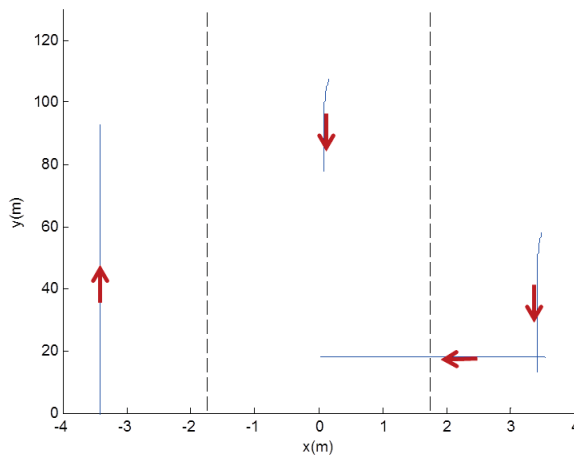


FIGURE III.31. Illustration des trajectoires des véhicules relativement au véhicule porteur du capteur pour le scénario établi au tableau III.4

La figure III.32 montre des images de la scène, présentée en figure III.31, à différents instants de simulation $t = 0s, t = 2,5s, t = 5s$ et $t = 10s$. On remarque ainsi l'apparition d'un véhicule à $t = 2,5s$, le déplacement d'un véhicule de la voie de droite à la voie centrale qui a pour conséquence l'occultation du véhicule voie centrale.

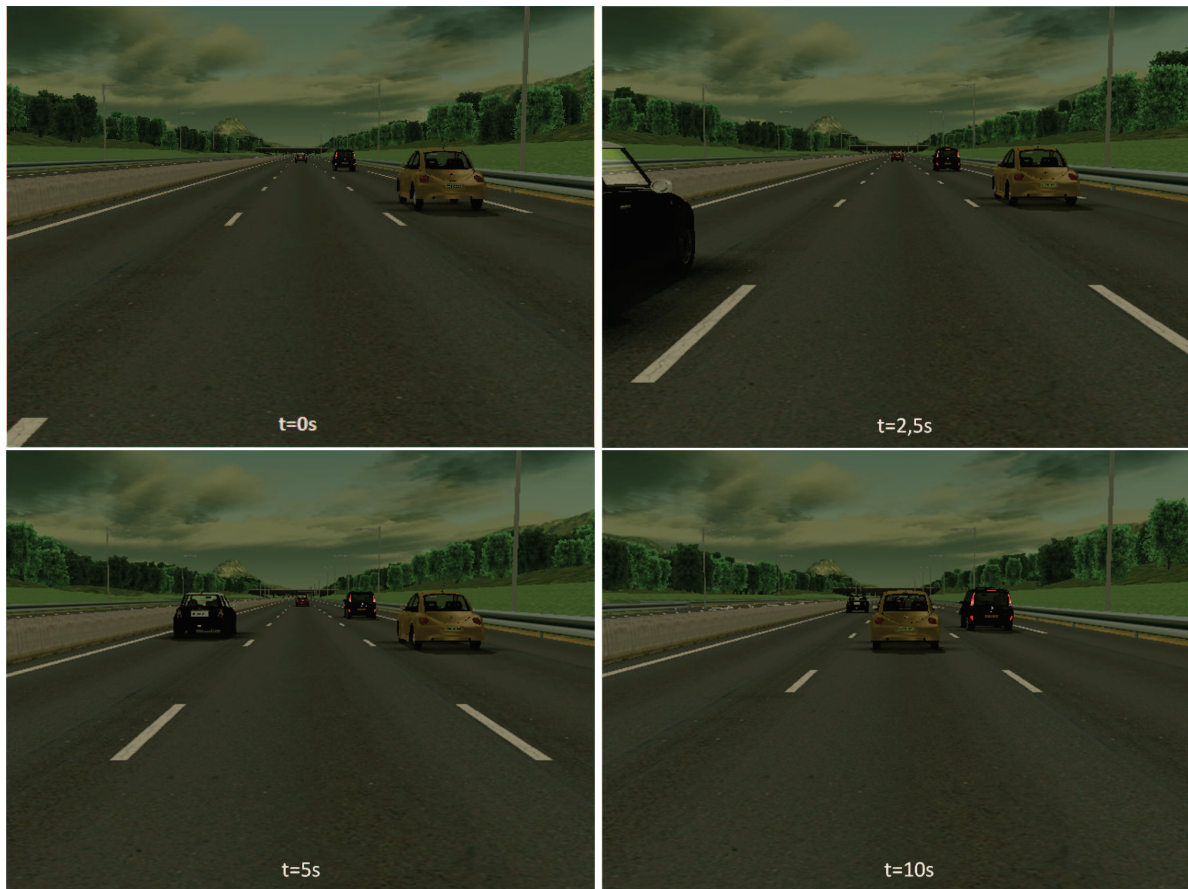


FIGURE III.32. Quatre images de la scène au temps $t = 0s$, $t = 2,5s$, $t = 5s$ et $t = 10s$

III.7.2.2 Comparaison de l'approche proposée ITPF à une approche classique (CDG) sur des données du simulateur SiVIC

Dans cette section, nous présentons une comparaison de notre méthode intégrée de détection et de suivi d'objet (ITPF) avec une méthode classique basée sur l'utilisation du centre de gravité (CDG) des objets et un filtre particulaire. Nous nous plaçons donc dans les mêmes conditions algorithmiques que précédemment (voir le paragraphe III.7.1.2), mais nous appliquons ces algorithmes sur des données plus réalistes. Les paramètres de simulation sont identiques pour chacun des deux filtres. Les résultats sont présentés en vis-à-vis. Ceux de la méthode ITPF sont sur les figures de gauche et ceux de la méthode CDG sur celles de droite.

Nous vérifions ci-après le nombre de pistes créées par les algorithmes en fonction du nombre de véhicules présents à chaque instant (figure III.33), ensuite nous vérifions la modélisation des véhicules estimés en fonction des mesures (figure III.36 à III.42), puis nous comparons les courbes d'estimations des vitesses et accélérations obtenues par les deux méthodes (figure III.43 à III.46).

Les schémas de la figure III.33 présentent le nombre de pistes créées et le nombre de véhicules présents pour les méthodes ITPF à gauche et CDG à droite en fonction du temps.

Contrairement aux résultats présentés à partir des données du simulateur de la thèse (section III.7.1), on remarque que le nombre de véhicules détectés ne correspond pas toujours au nombre de véhicules présents. Faisons un rappel de la structure de traitement détaillée précédemment. Lorsqu'une piste n'est plus détectée

par le module de surveillance, le filtre de Rao-Blackwell peut continuer à fonctionner dès lors qu'il peut identifier des mesures compatibles avec la piste associée (c'est le cas notamment aux environs de l'instant $t = 6s$). De plus, le module de surveillance élimine une piste uniquement après trois itérations (équivalent à 0,15s) sans mesures pour mettre à jour la piste, donc sans détection du véhicule. On peut ainsi observer une continuité du suivi aux environs des instants $t = 6s$ et $t = 12s$. Ainsi le filtre de la méthode ITPF continue à suivre le véhicule alors que le module de surveillance a cessé de le détecter (en raison soit d'un nombre d'impacts trop faible, soit d'une absence d'impacts). Dans le cas de la méthode CDG (figure de droite), le constat est le même. La méthode CDG se basant sur une agrégation de la mesure issue de la détection des objets, l'élimination de la piste se fait uniquement en fonction de la détection. Ainsi la piste est éliminée par le module de surveillance après trois itérations sans détection de l'objet en cours de suivi. C'est pourquoi sur la figure III.33 à $t \simeq 6s$ la piste continue d'être suivie alors qu'elle n'est plus détectée et ce pendant un laps de temps plus long pour la méthode ITPF.

On remarque également un pic à l'instant $t \simeq 8s$ qui est dû au cas d'une apparition et d'une disparition de deux véhicules (quasi) simultanément sur la scène. À cet instant, le véhicule se déplaçant de la voie 3 à la voie 2 occulte le véhicule situé en voie centrale alors que le véhicule de la voie 3 (précédemment occulté par ce déplacement) réapparaît. La configuration du filtre a été choisie de telle sorte que le suivi puisse continuer environ 0,15s après disparition totale des mesures, d'où ce pic à $t \simeq 8s$ de 0,15s. Cette configuration a été définie afin de pouvoir prendre en compte la possibilité qu'un véhicule ne soit pas mesuré par le lidar à un instant donné (phénomène observé en pratique).

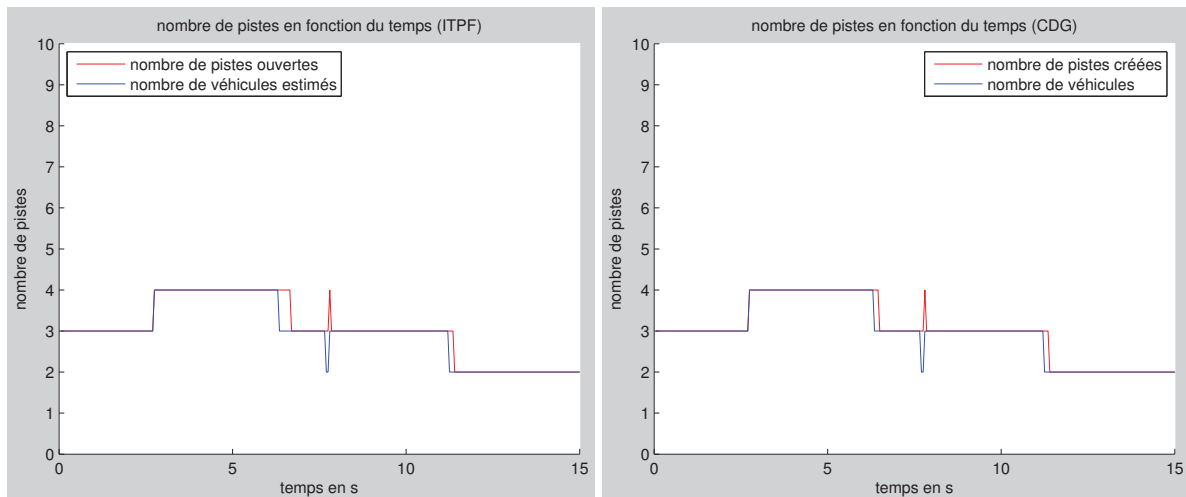


FIGURE III.33. Comparaison du nombre de pistes créées avec le nombre de véhicules détectés pour les méthodes ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC

Cas de la perte de détection par masquage

Dans cette section, nous présentons plus en détail la continuité du suivi dans le cas de la perte de détection. Le détecteur d'objets est construit tel qu'un véhicule n'est pas considéré comme véhicule à suivre dès lors que le nombre d'impacts le caractérisant est inférieur à quatre ou que sa largeur détectée est inférieure à 50cm.

Pour illustrer la perte d'un véhicule par masquage de celui-ci nous étudions plus en détail la scène aux alentours de l'instant $t = 6,3s$. En effet, à cet instant, sur la figure III.34, on observe la diminution du nombre de véhicules détectés par le module de surveillance. Le suivi du véhicule continue cependant, durant encore quelques instants.

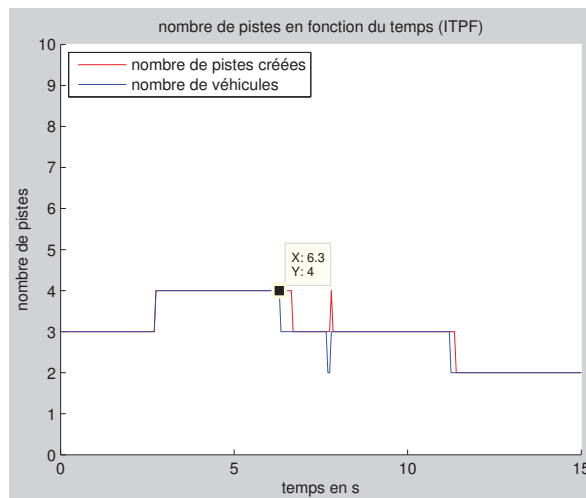


FIGURE III.34. Nombre de pistes créées avec le nombre de véhicules détectés pour la méthode ITPF appliquées aux données du simulateur SiVIC

La figure III.35 montre comment le filtre ITPF agit lors de la disparition progressive d'un véhicule par masquage. Durant le scénario présenté ici, un véhicule (le 4) se déplace de la voie de droite à la voie centrale occultant progressivement un véhicule situé sur la voie de droite (le véhicule 3). Nous avons choisi d'associer à chaque image CCD de la scène, le balayage capteur qui lui est associé afin de dissocier la scène réelle (à gauche) et ce que le capteur observe (à droite). Sur l'image capteur, on observe en rouge les mesures et en bleu le modèle objet issu du filtre.

La figure III.34 montre que le dernier instant de détection du véhicule 3, par le module de surveillance, est $t = 6,3s$. C'est pourquoi, nous présentons quelques résultats de détection et de suivi à partir de cet instant. À $t = 6,3s$, le véhicule 3 est déjà partiellement masqué par le véhicule 4, on peut ainsi voir sur la figure III.35 du haut que seuls les impacts du véhicule 3 situés les plus à gauche sont représentés. On remarque que le suivi (en bleu) n'est pas impacté par la diminution du nombre d'impacts, en effet, le modèle objet semble traverser les mesures issues du capteur. Le constat est le même pour la figure III.35 du centre, où le nombre d'impacts a diminué conformément au déplacement du véhicule 4. Enfin, sur la figure III.35 du bas, il n'y a plus d'impacts caractérisant le véhicule 3. Le véhicule est donc reconstruit sur un certain horizon temporel avant de disparaître s'il ne réapparaît pas dans les mesures.

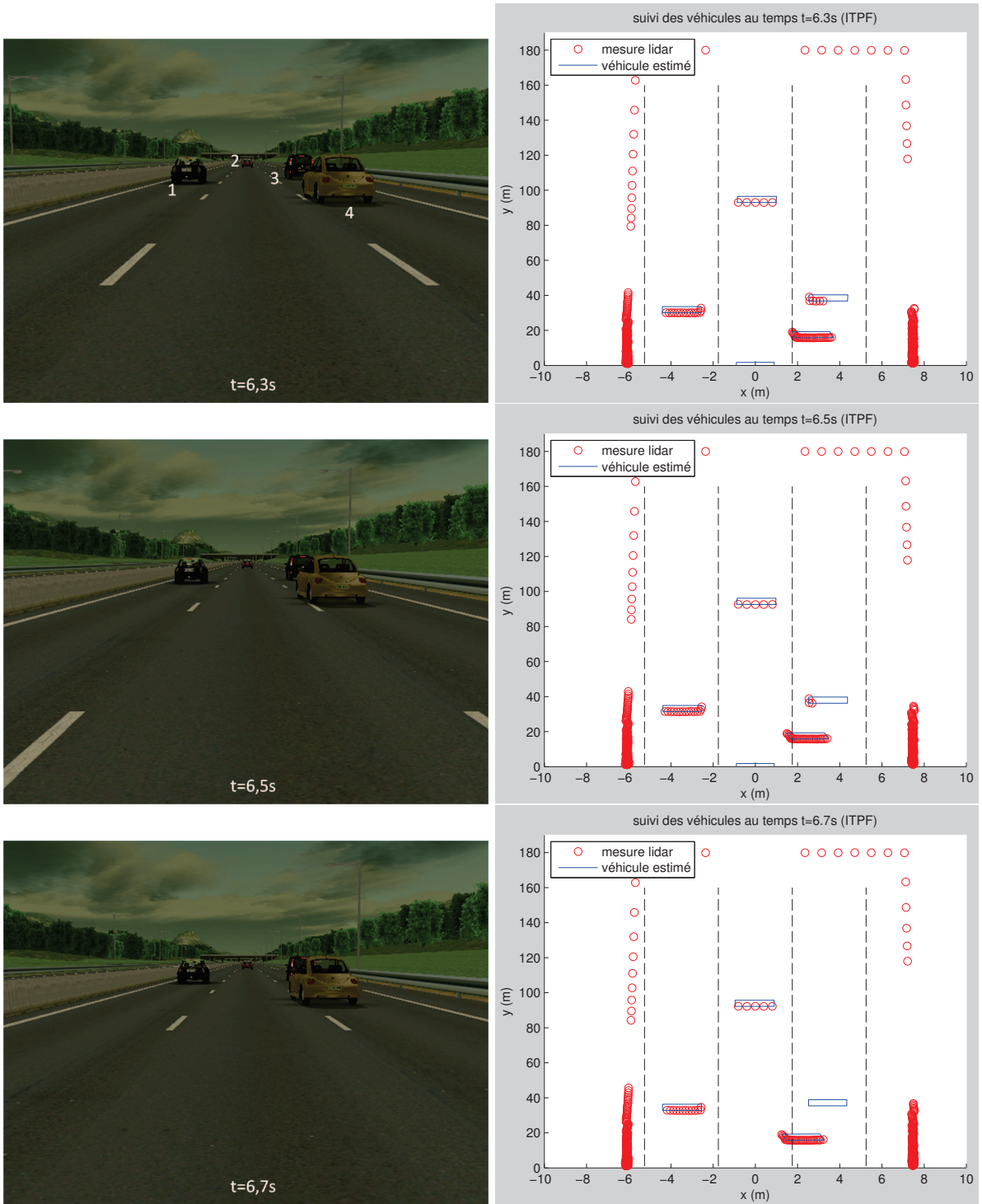


FIGURE III.35. Illustration de la perte d'un véhicule par masquage, image de la scène à $t = 6,3s$, $t = 6,5s$ et $t = 6,7s$

Les figures III.36 et III.41 mettent en correspondance une image CCD de la scène et sa perception par le capteur lidar respectivement à 5 secondes et à 10 secondes. Les figures III.37 à III.40 sont des agrandissements de chacun des véhicules de la figure III.36, à $t = 5$ s. Celles de III.42 sont des agrandissements de chacun des véhicules de la figure III.41, à $t = 10$ s. Pour chacune de ces figures, les mesures du lidar et la boîte englobante, de la position estimée, ont été superposées, aussi bien pour les résultats de l'approche ITPF que ceux de l'approche CDG. On remarque sur ces données réalistes que le suivi CDG est moins précis, notamment en raison d'une délocalisation liée à la courbure du véhicule, ainsi qu'au nombre d'impacts limité sur les côtés.

Cette fois encore, la reconstruction du modèle objet (défini ici par une boîte englobante) correspondant à la position estimée se fait à partir de la position estimée du centre de gravité, de la connaissance *a priori* de la largeur et de la longueur des véhicules (i.e. les dimensions du modèle de véhicule utilisées dans le simulateur) et de celle de l'orientation des véhicules. Cette orientation est obtenue à partir de l'estimation de leur vitesse en x et y . Sa précision est donc aussi impactée par l'estimation des accélérations des véhicules, accélération qui entre dans le calcul des vitesses.

On remarque que pour toutes les reconstructions de véhicule (en bleu) de la méthode ITPF (à gauche sur les figures) le modèle est parfaitement superposé aux mesures (en rouge). Les véhicules sont correctement accrochés, car le modèle se superpose correctement aux mesures. Par contre, pour certains véhicules de la méthode CDG, on observe que le véhicule est incorrectement accroché. Le modèle estimé en bleu ne correspond pas toujours aux mesures en rouge tant au niveau de l'orientation du véhicule que de sa position par rapport aux mesures. L'utilisation de la mesure brute au cours des calculs de la méthode ITPF permet donc de gagner en précision. On retrouve cette constatation sur les estimations de la vitesse et de l'accélération où la méthode ITPF est stable comparativement à la méthode CDG.

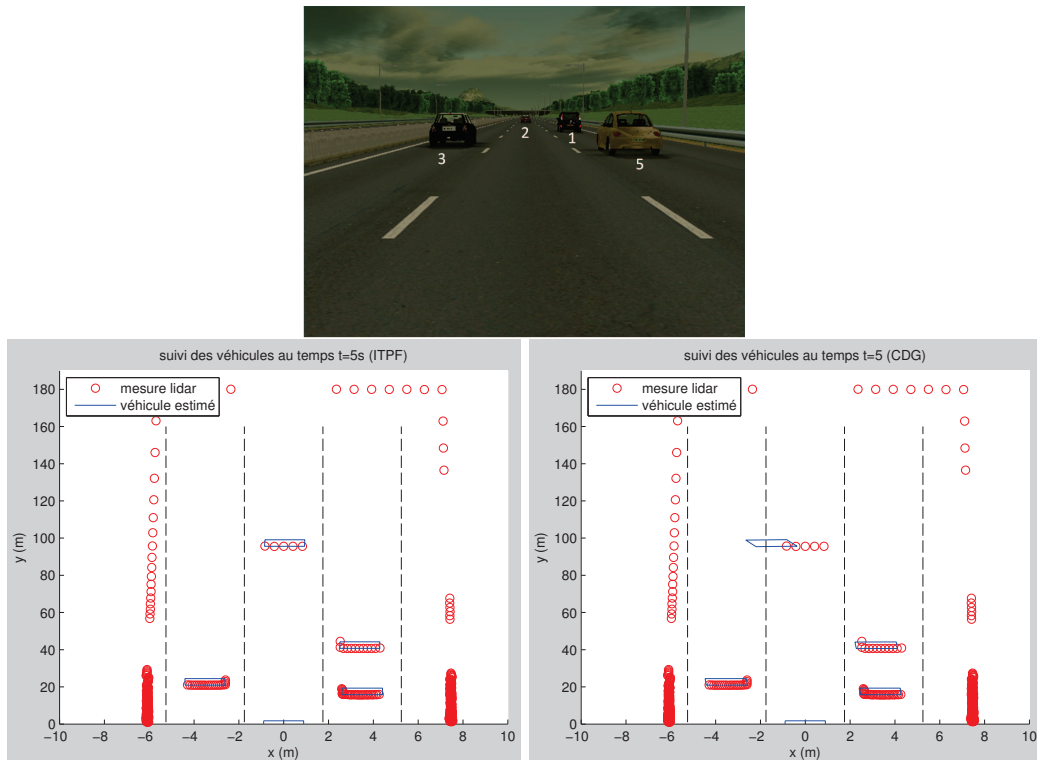


FIGURE III.36. Image CCD de la scène, au temps $t = 5$ s et la même scène perçue par le laser à balayage (en rouge) superposée aux modèles estimés des véhicules (rectangle en bleu) : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC

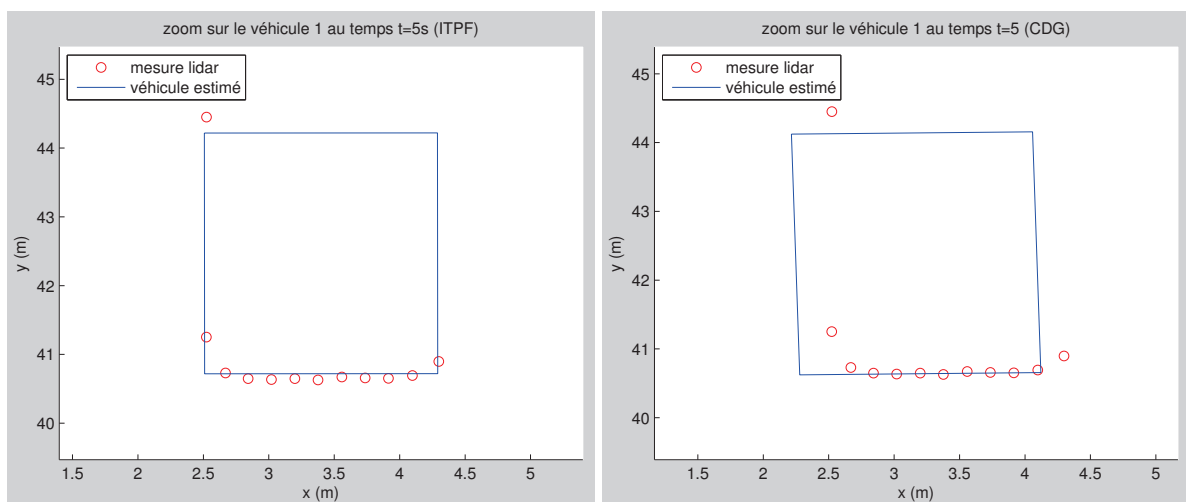


FIGURE III.37. Agrandissement du véhicule de la voie de droite haut, au temps $t = 5$ s, de la figure III.36 : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC

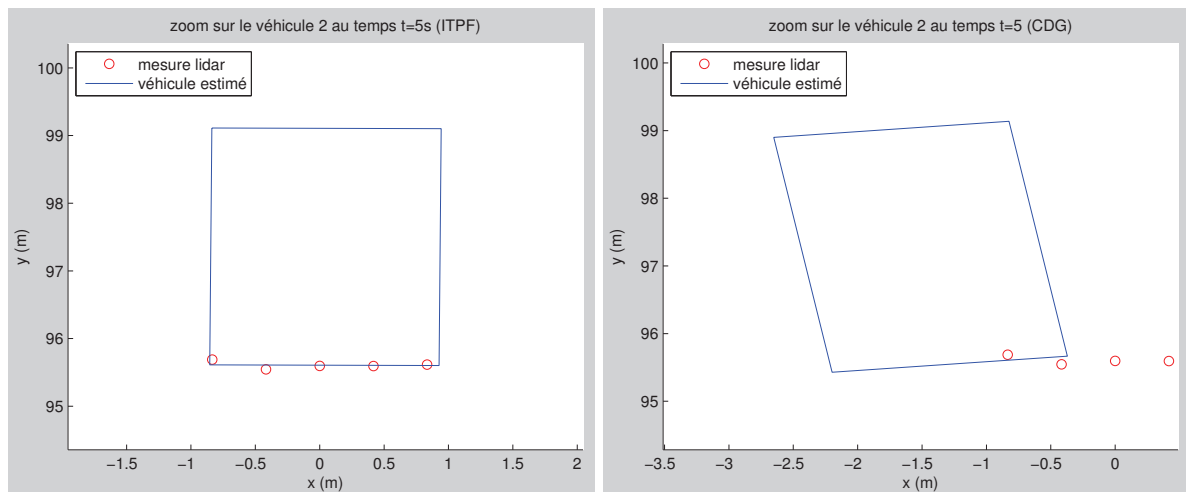


FIGURE III.38. Agrandissement du véhicule de la voie centrale, au temps $t = 5s$, de la figure III.36 : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC

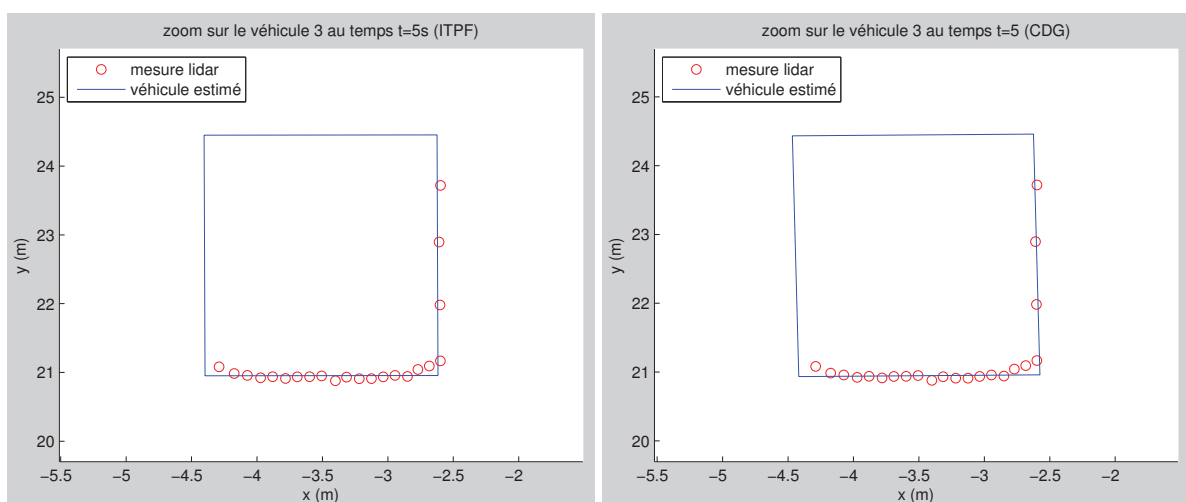


FIGURE III.39. Agrandissement du véhicule de la voie de droite bas, au temps $t = 5s$, de la figure III.36 : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC

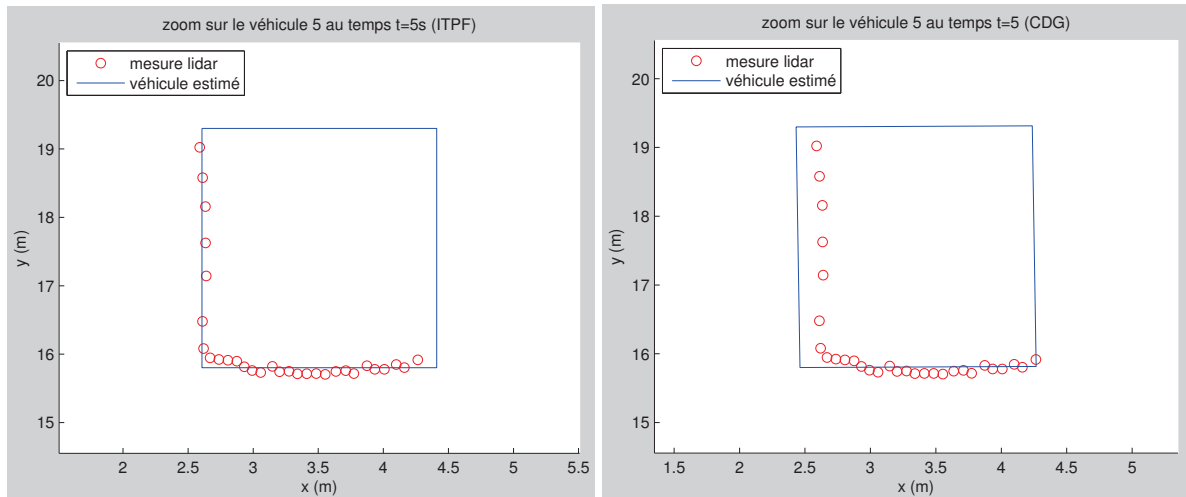


FIGURE III.40. Agrandissement du véhicule de la voie de gauche, au temps $t = 5s$, de la figure III.36 : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC

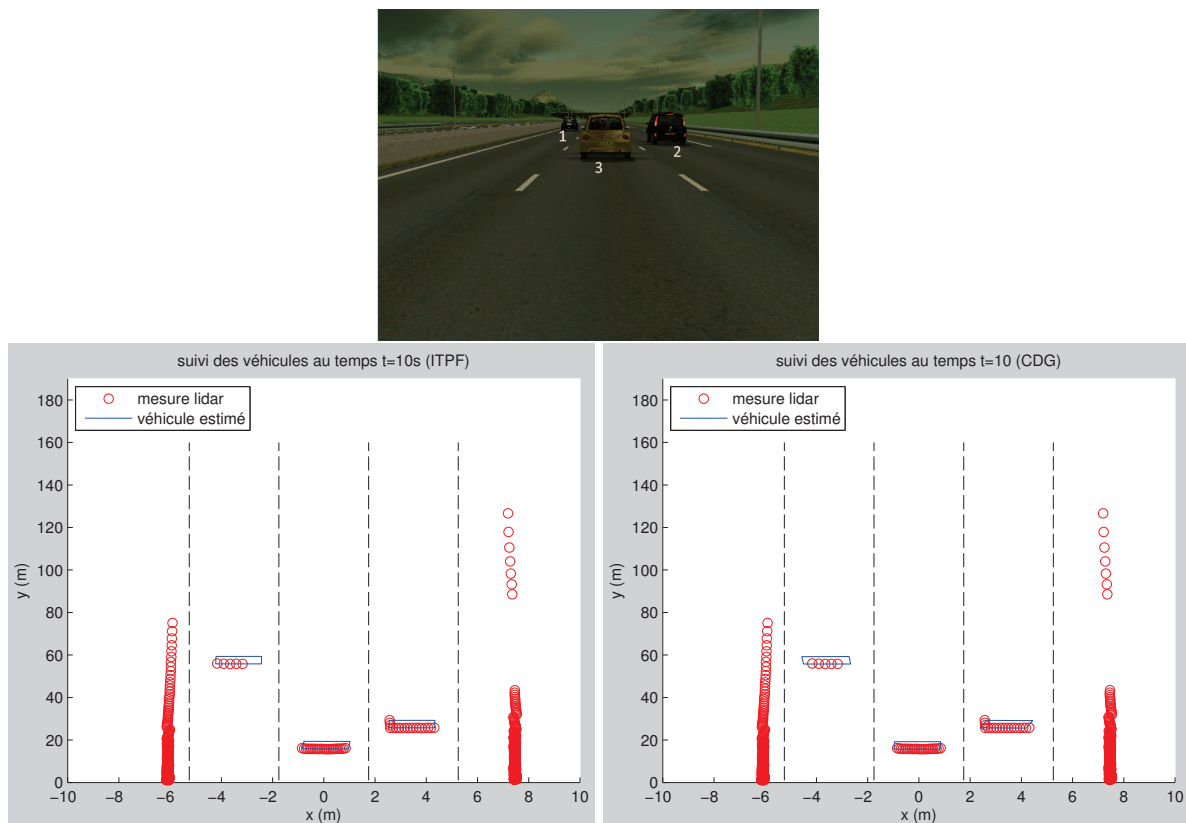


FIGURE III.41. Image CCD de la scène, au temps $t = 10s$ et la même scène perçue par le laser à balayage (en rouge) superposée aux modèles estimés des véhicules (rectangle en bleu) : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC

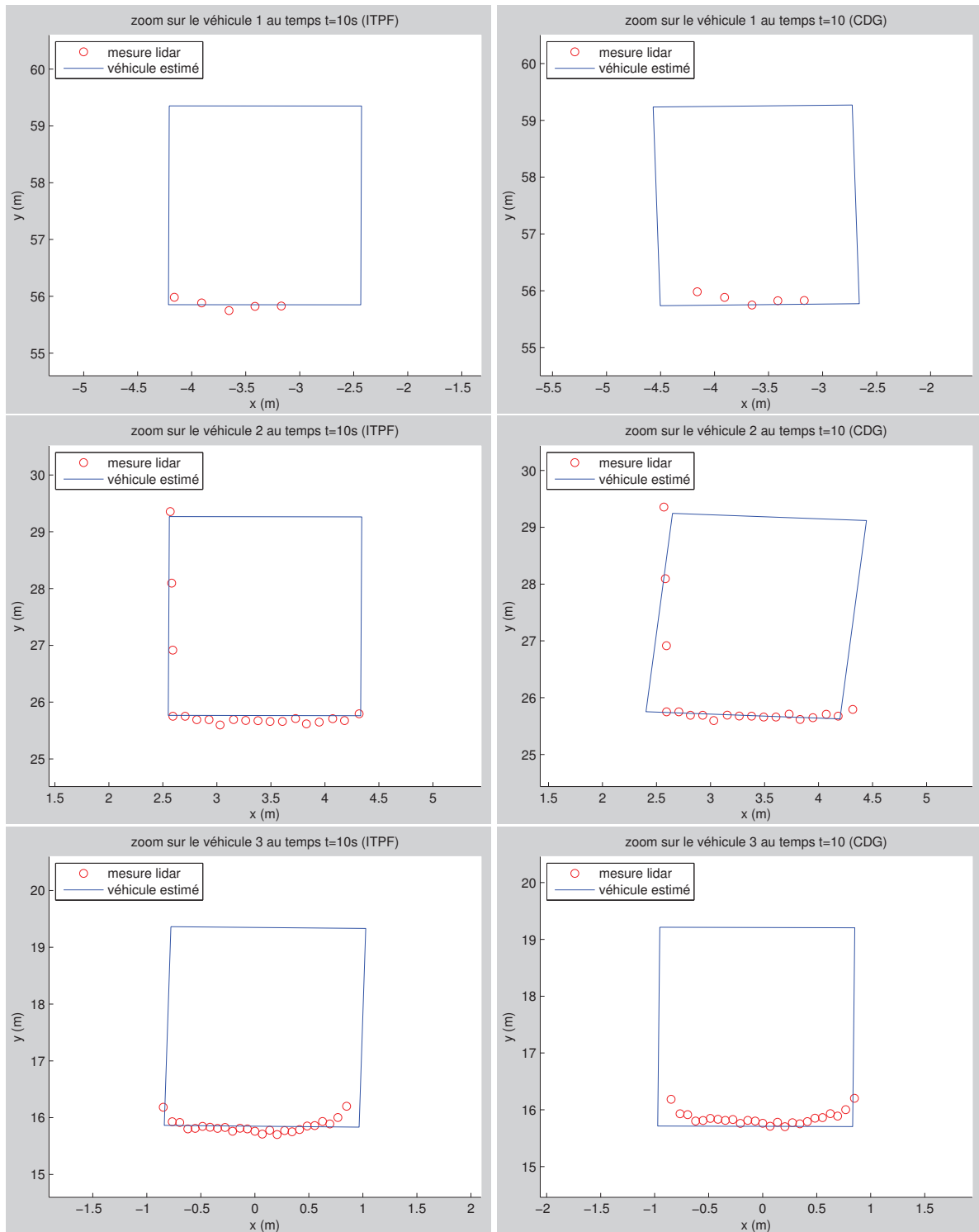


FIGURE III.42. Agrandissement des véhicules de la scène figure III.41, au temps $t = 10s$, véhicule de la voie de gauche en haut, de la voie de droite au centre et de la voie centrale en bas : approches ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC

Les estimations des vitesses et des accélérations de deux véhicules tout au long de la simulation pour les

méthodes ITPF et CDG sont maintenant retracées sur les figures III.43 à III.46.

- Les figures III.43 et III.44 présentent celle d'un véhicule visible du début de la simulation au temps $t = 6s$, puis occulté par un autre véhicule durant environ 2s avant de réapparaître et de rester présent jusqu'à la fin de la simulation. L'occultation, et donc l'interruption de suivi résultante, se traduit par un changement de piste et donc de couleur sur les figures.
- Les figures III.45 et III.46 présentent les vitesses et accélérations d'un véhicule qui change de voie au cours de la simulation, mais reste visible tout au long de celle-ci (pas d'occultation).

On remarque que sur les figures de gauche, de l'approche ITPF, les estimations de la vitesse et de l'accélération semblent ne pas varier fortement et ceci sur la majorité de la simulation. Par contre, sur les figures de droite, de l'approche CDG, les estimations de la vitesse et de l'accélération en x varient plus fortement. L'estimation en x fluctue plus fortement que pour ITPF et semble rester instable. La convergence semble plus difficile. On montre là encore, mais cette fois avec des mesures SiVIC, que le filtre CDG supporte mal les mouvements relatifs latéraux des véhicules cibles, contrairement à la méthode ITPF.

Il est donc important de remarquer que l'on retrouve des performances similaires au cas synthétique pour la méthode ITPF. La précision de la pose est conservée, ainsi que la qualité de détection du nombre de pistes pour des modèles de véhicules dans SiVIC dotés de courbures contrairement aux premiers modèles du simulateur de la thèse. Par ailleurs, on a montré que la prise en compte des données brutes permet de prolonger le suivi lors d'occultation alors que la détection est rendue difficile voire inopérante à ce moment.

Les résultats obtenues sur des données SiVIC sont donc similaires à ceux obtenus avec les données du simulateur de la thèse. En influant sur la rotation du véhicule, la vitesse et l'accélération contribuent à la précision du véhicule reconstruit par rapport aux mesures (cf. figures III.36 à III.42) où l'on a pu observer que les véhicules reconstruits issus de la méthode ITPF assurent une meilleure interprétation des mesures que ceux de la méthode CDG. Ils mettent une fois de plus en exergue la stabilité de la méthode ITPF. Par ailleurs, comme on peut le constater sur les figures III.43 à III.46 la précision en estimation des paramètres cinématiques de la méthode ITPF est accrue par rapport à l'approche CDG, malgré l'usage de paramètres identiques dans les filtres particuliers employés.

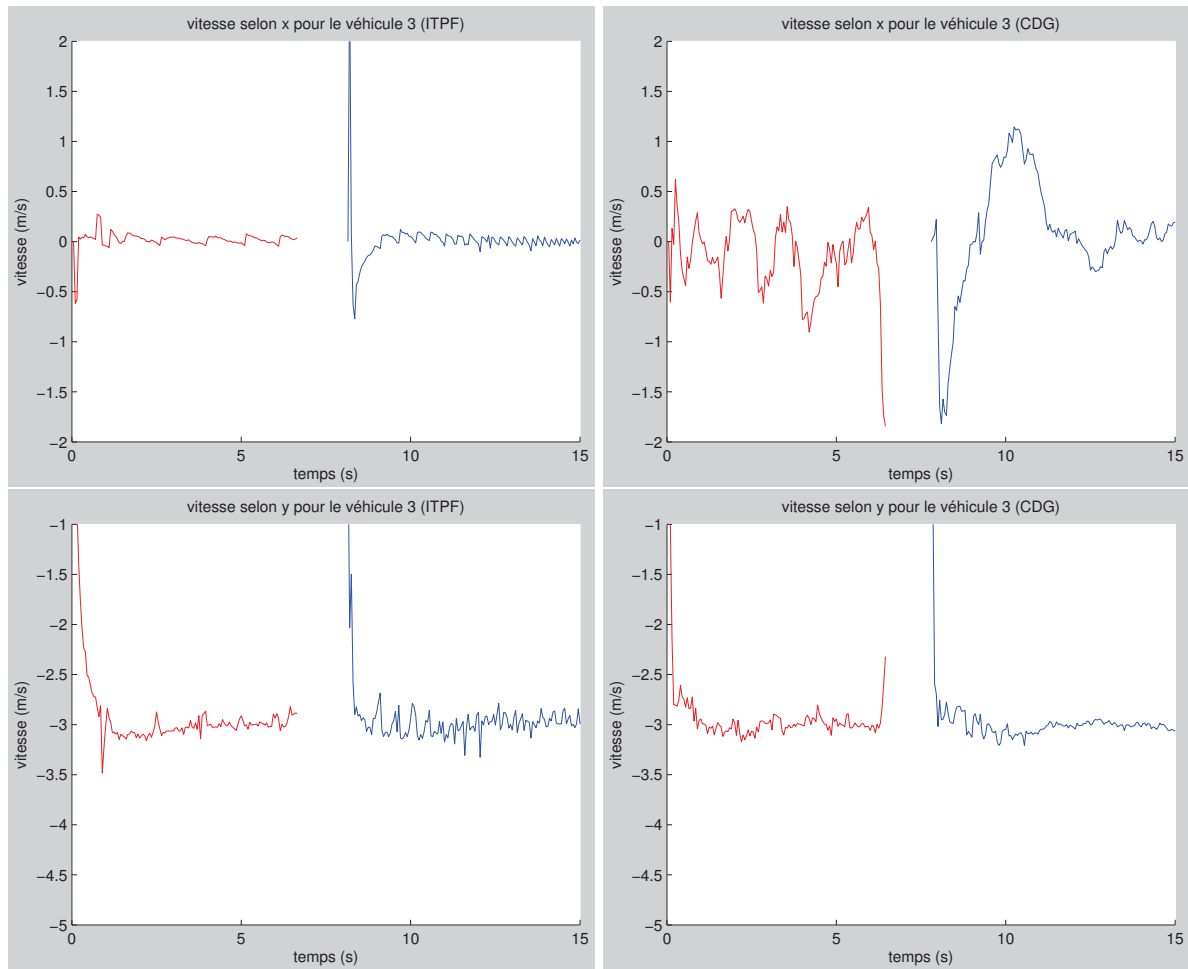


FIGURE III.43. Courbes de vitesse selon x en haut et y estimées du véhicule de la voie de droite en fonction du temps : approches ITPF à gauche et CDG à droite appliquées aux données SiVIC

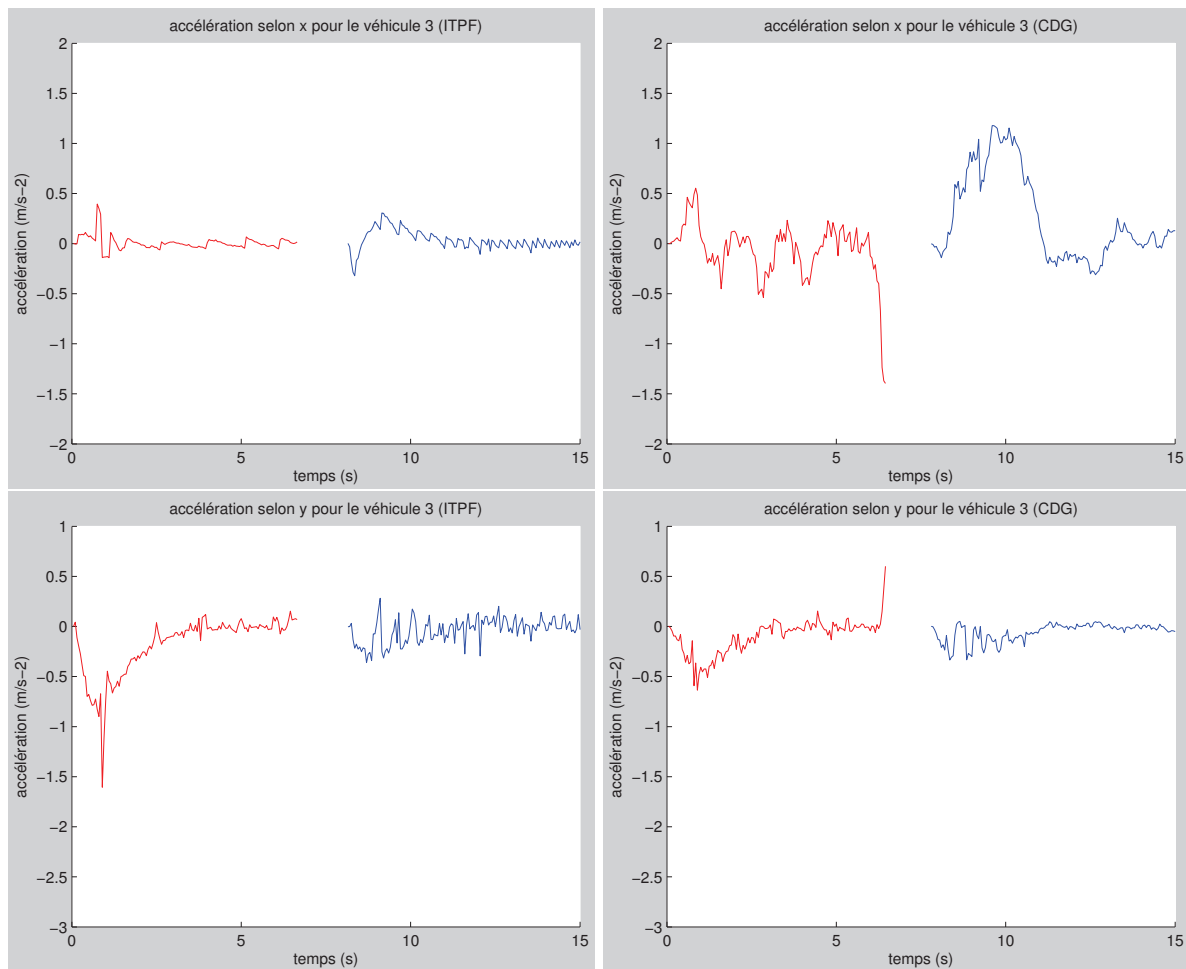


FIGURE III.44. Courbes d'accélération selon x en haut et y estimées du véhicule de la voie de droite en fonction du temps : approches ITPF à gauche et CDG à droite appliquées aux données SiVIC

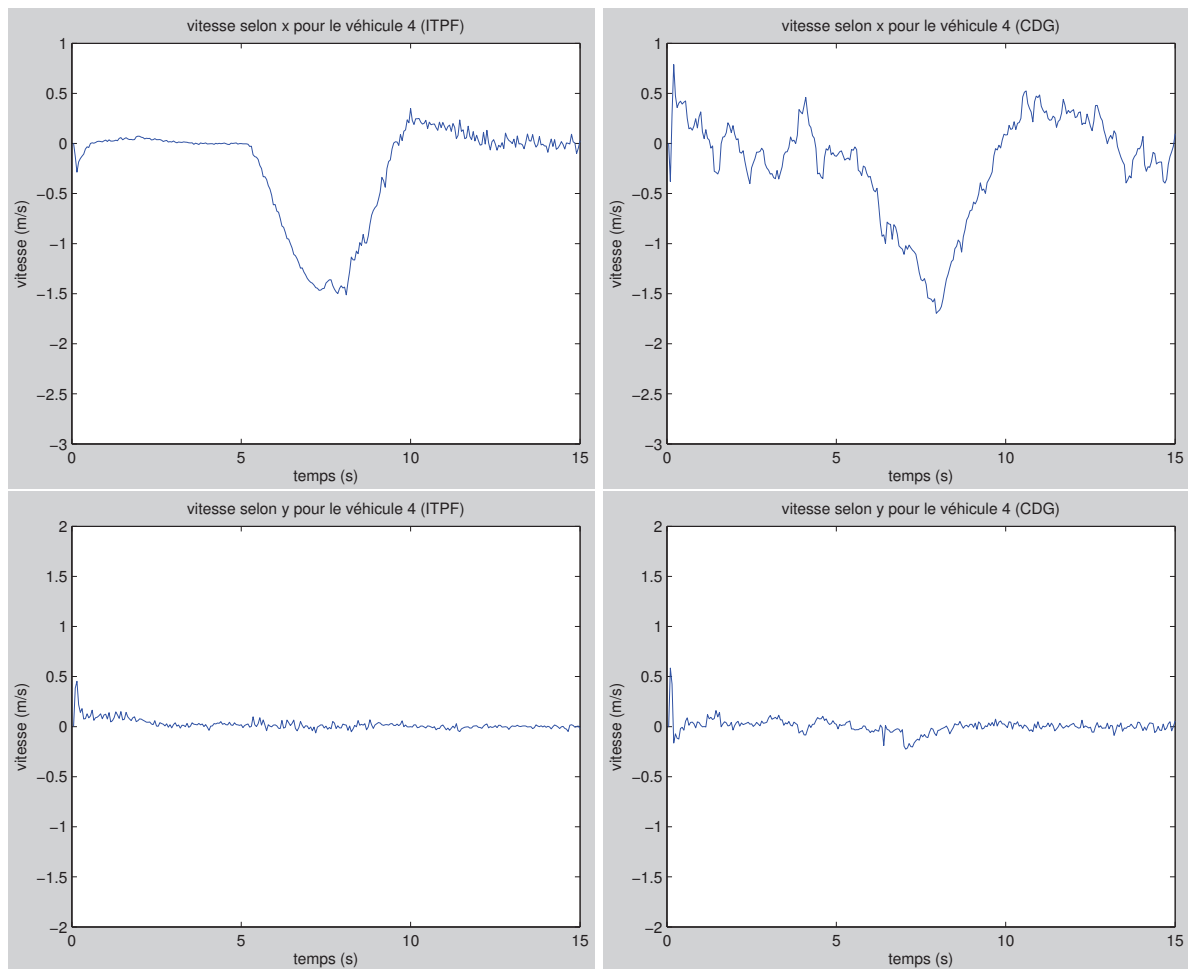


FIGURE III.45. Courbes de vitesse selon x en haut et y estimées du véhicule de la voie centrale en fonction du temps : approches ITPF à gauche et CDG à droite appliquées aux données SiVIC

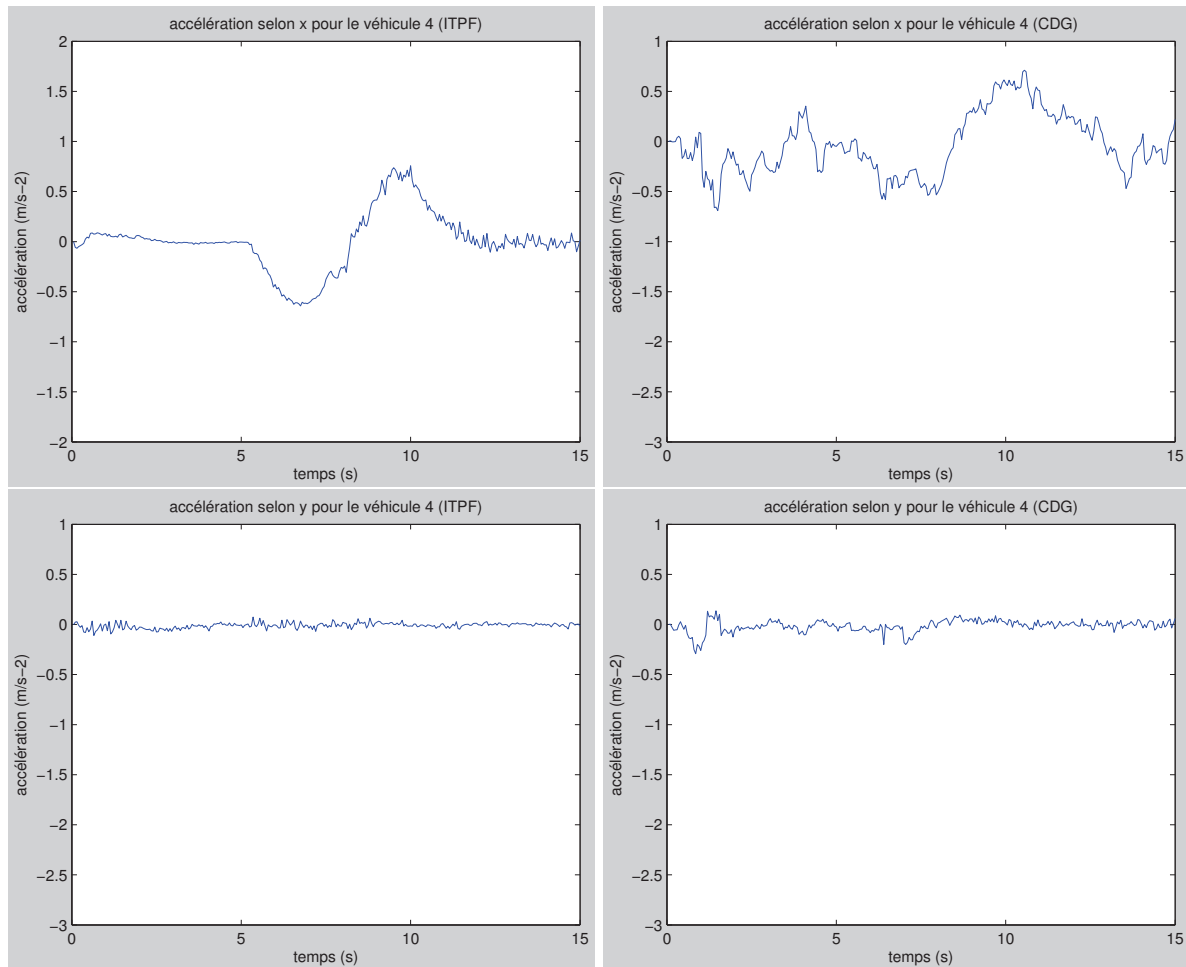


FIGURE III.46. Courbes d'accélération selon x en haut et y estimées du véhicule de la voie centrale en fonction du temps : approches ITPF à gauche et CDG à droite appliquées aux données SiVIC

III.8 Conclusion

Nous avons présenté dans ce chapitre une nouvelle méthode conjointe de détection, association et suivi ITPF basé sur le filtrage particulaire, et plus particulièrement sur une approche de type Rao-Blackwell. La particularité de notre approche intégrée réside dans l'usage d'une modélisation des objets sous forme d'échos. Ces échos ont été obtenus par une modélisation, dans un premier temps, du système sous forme de modèles d'états qui suivent la position (fortement non-linéaire), la vitesse et l'accélération (parties linéaires du modèle) des centres de gravité des objets suivis. Ces centres de gravité servent alors à reconstruire des objets véhicules sous forme de boîtes englobantes. Ces boîtes englobantes sont ensuite utilisées pour reconstruire les échos des véhicules à partir d'un modèle de fonctionnement du capteur lidar. Ce modèle d'objet du véhicule, qui prend en compte l'ensemble des échos des véhicules, améliore l'estimation, car cette dernière tire alors partie de l'ensemble des mesures brutes en sortie du capteur (et non d'une information agrégée de la mesure).

Nous concluons ce chapitre par une comparaison de la méthode intégrée de détection et de suivi ITPF proposée avec une méthode séquentielle de détection, association et suivi, plus classique. Cette comparaison a été réalisée sur des données synthétiques délivrées par le simulateur développé en thèse, dont on maîtrise

le comportement et le simulateur SiVIC qui utilise des modèles de routes, de véhicules, de capteurs, de dynamiques... proches de la réalité. Ceci a permis de mettre en évidence les bonnes performances de notre approche intégrée comparativement à une approche classique. Nous avons montré la stabilité de notre filtre, notamment suite à des changements brutaux en accélération. Nous obtenons une meilleure estimation des positions (plus faibles erreurs). D'autre part, les estimations de la vitesse et de l'accélération convergent avec la méthode ITPF, mais restent fluctuantes avec la méthode CDG.

Par ailleurs, nous avons montré les bonnes performances de la méthode en présence de phénomènes d'occultation. Cette problématique est généralement difficile à traiter par les approches basées sur la détection. Dans notre approche ITPF, l'exploitation des données lidar brutes ont permis de prolonger le suivi lors d'occultations même si le nombre de mesures est insuffisant pour donner lieu à une détection fiable par les méthodes usuelles.

En conclusion, de ce chapitre nous avons montré l'intérêt de la méthode ITPF et son apport vis-à-vis des méthodes usuelles. Pour aller plus loin, nous montrons dans le chapitre suivant comment cette approche peut être étendue au cas des :

- objets de tailles multiples : différents gabarits de voiture, camion, piétons...
- capteur lidar multiplan. En effet, les capteurs lidar monoplan ont montré, sur route, leur vulnérabilité au tangage en particulier. L'usage des capteurs multiplans peut donc s'avérer, dans certain cas, indispensable à un suivi efficace des objets d'intérêt.

La difficulté sera donc d'étendre l'approche proposée à l'étude de ces deux problématiques, critiques dans le contexte routier.

Chapitre IV

Approche conjointe de détection et de suivi d'objets multiples hétérogènes par laser multiplans

IV.1 Introduction

La méthode intégrée de détection et de suivi mise en place a été présentée dans le chapitre III. Nous rappelons ci-après l'architecture fonctionnelle. Elle est composée d'un module de surveillance chargé de détecter puis de gérer les apparitions et disparitions des objets dans le champ de vision du capteur, et d'un module DAT (*Detected, Association and Tracking*) mettant en œuvre des filtres ITPF (*Integrated tracking [method] based on particle filtering*). Ces filtres reposent sur la mise en œuvre d'un filtre de Rao-Blackwell qui traite conjointement la détection, l'association et le suivi des objets. L'efficacité de cette méthode a été montrée pour le cas particulier d'un lidar monoplan et dans le cas d'un modèle de voiture de taille donnée. On a également montré la robustesse de la méthode à une erreur sur les dimensions supposées du véhicule : en effet, dans le chapitre précédent les véhicules ont été modélisés par une boîte englobante unique de taille fixée.

Nous nous intéressons maintenant, dans ce chapitre, à l'extension de notre méthode au cas du suivi multi-objets hétérogènes (voiture, camion...), puis à son extension pour le traitement des données d'un lidar multiplan :

- le cas multi-objets hétérogènes vise à prendre en compte la diversité des véhicules suivis en termes de dimensions notamment. Dans le chapitre précédent le modèle de suivi était supposé identique pour chaque objet d'intérêt. Généralement, la présence d'objets de différentes natures implique l'utilisation d'une étape supplémentaire de classification. Nous choisissons de mettre en place un modèle à dimension variable afin de prendre en compte les différentes caractéristiques des objets, tout en évitant les étapes successives de détection, classification et association.
- le cas multiplan nécessite de prendre en compte les différents plans de tir du capteur lors du traitement des données d'un balayage horizontal de la scène. Le cas multiplan revient ici à une acquisition selon plusieurs élévations d'un balayage complet de la scène. Cela entraîne une augmentation du nombre d'impacts à traiter et surtout une modification de la forme mesurée. Suivant l'élévation du tir laser, il est possible qu'un véhicule ne soit pas atteint par le tir. Un véhicule peut ainsi, à un instant donné, être vu par plusieurs plans et à l'instant d'après par un seul plan. La méthode de détection et de suivi doit être capable de prendre en compte ces différents plans et ce peu importe leur nombre.

Lors de ces extensions, les caractéristiques de notre méthode conjointe de détection et de suivi sont conservées, en particulier, la non utilisation d'une phase de détection des objets de la scène et la non influence

du module de surveillance sur les filtres ITPF. Nous présentons l'extension de la méthode ITPF au cas multi-objets en section IV.2. La section IV.3 est consacrée à la mise en œuvre de notre méthode pour un capteur multiplan.

IV.2 Détection et suivi d'objets multiples hétérogènes (approche VS-ITPF)

IV.2.1 Problématique du cas multi-objets

Le cas multi-objets concerne le suivi de différents types d'objets, objets qui se différencient par leur taille et leur forme en plus de leur dynamique et emplacement.

Précédemment, au chapitre III, nous avons détaillé notre méthode de détection et de suivi qui réalise une reconstruction des objets au moment du calcul du poids. Le calcul du poids de chaque particule générée pour un véhicule suivi est réalisé à partir de cette reconstruction et des mesures brutes du véhicule de l'instant courant. Pour la reconstruction du véhicule, nous utilisons le centre de gravité de la particule estimé (par le filtre), la rotation du véhicule, obtenue à partir de la vitesse estimée (par le filtre), ainsi qu'un modèle objet unique de boîte englobante (rectangulaire) et de dimensions fixes. Ainsi, le module ITPF utilise un modèle de reconstruction des véhicules qui est donc potentiellement inexact si plusieurs objets de différentes tailles coexistent. Utiliser un modèle de dimensions inférieures au gabarit réel de l'objet fonctionne, cependant, ce modèle n'est pas réaliste et peut induire une erreur de suivi. *A contrario*, utiliser un modèle de dimensions supérieures au gabarit réel de l'objet induirait, selon la différence de dimensions, une erreur plus ou moins importante. Il faut ici prendre en compte les particularités des objets à suivre et cela rend le suivi plus complexe.

Classiquement, lorsque différents types de véhicules sont présents dans une image capteur, une étape de classification est mise en œuvre après l'étape de détection. Cependant, dans notre méthode de détection et de suivi intégrée, il n'existe pas d'étape de détection distincte, la classification des objets ne peut donc se faire avant le suivi. De plus, notre module de surveillance n'a pas la capacité d'influer sur les modules ITPF contenus dans le module DAT, il ne fait que transmettre des informations. De ce fait, si on ajoute une étape de classification dans ce module, elle ne serait potentiellement pas prise en compte par le DAT. La difficulté réside désormais dans la prise en compte des dimensions réalistes sinon exactes des véhicules à suivre, sans pour autant mettre en place une phase de classification, ni casser l'autonomie des filtres ITPF, une fois initiés, vis-à-vis du module de surveillance.

Nous présentons ci-après la méthode VS-ITPF (*variable size ITPF*) qui permet, à partir du calcul de différentes longueurs et largeurs de modèles dans le module de surveillance, d'informer le filtre de l'existence d'un modèle plus approprié que celui qu'il utilise, sans pour autant prendre la main sur son fonctionnement. Nous présentons cette méthode au travers de l'exemple d'un camion et plusieurs voitures. Puis nous la validerons sur un scénario réalisé à l'aide du simulateur de la thèse et un scénario du simulateur SiVIC.

IV.2.2 Adaptation du module de surveillance au cas multi-objets hétérogènes

Nous avons choisi l'exemple du camion car ses dimensions imposantes peuvent être source d'erreur de suivi. La difficulté majeure de l'adaptation de la méthode intégrée de détection et de suivi au cas multi-objets réside dans la prise en compte des dimensions des différents objets.

Ce module de surveillance donne une connaissance *a priori* des objets présents sur la scène au module DAT. Il lui fournit en particulier l'information du centre de gravité de l'objet et de sa dimension à chaque nouvelle détection.

La prise en compte d'un modèle de dimensions variables (longueur et largeur) et la translation du centre de gravité en fonction des dimensions de l'objet qui en résulte vont agir sur les étapes de détection et de gestion de pistes de ce module.

Adaptation de l'étape de détection

L'algorithme de détection présenté au chapitre III compte quatre étapes :

1. les impacts du balayage capteur sont, tout d'abord, restreints à une zone d'étude (section III.4.1.1) ;
2. les impacts appartenant à la zone d'étude sont, ensuite, regroupés sous forme de partition, susceptible de représenter un même objet, à l'aide du seuil de Santos (section III.4.1.2) ;
3. les mesures des partitions sont, alors, segmentées pour en extraire des droites, à partir de la théorie de *Split and Merge* (section III.4.1.3) ;
4. enfin, à partir de ces informations sur la partition et d'un modèle objet, le centre de gravité est estimé (section III.4.1.4).

Dans le cas multi-objets, plusieurs modèles de véhicule peuvent maintenant coexister. On peut considérer que plusieurs centres de gravité différents peuvent alors être estimés en fonction des différents modèles. C'est sur cette hypothèse que nous basons notre approche. Nous choisissons de transmettre des informations supplémentaires au module de gestion de piste. Ainsi, en l'absence de connaissance *a priori* de l'objet suivi, pour chaque partition, deux centres de gravité vont être estimés et transmis au gestionnaire de piste ainsi que deux dimensions de modèle :

- le premier centre de gravité sera estimé à partir des impacts de la partition et du modèle de véhicule le plus petit, dans ce cas, les dimensions mises en mémoire seront les dimensions du modèle de véhicule le plus petit ;
- le second centre de gravité sera estimé à partir des impacts de la partition et des dimensions que l'on en extrait, dans ce cas, les dimensions mises en mémoire seront les dimensions des droites extraites de la partition.

Nous choisissons d'illustrer cette démarche à l'aide de l'exemple suivant, soit une partition $\mathcal{P}(n)$ composée de deux droites :

$$\begin{cases} (d_1) & : & X_1 & = & a_1 Y_1 + b_1 \\ (d_2) & : & Y_2 & = & a_2 X_2 + b_2 \end{cases}$$

Les droites (d_1) et (d_2) décrites ci-dessus sont (à titre d'exemple) respectivement parallèles à l'axe des ordonnées et des abscisses. Les couples (X_1, Y_1) et (X_2, Y_2) représentent respectivement les sous-ensembles d'impacts des droites 1 et 2. Ces sous-ensembles d'impacts permettent d'extraire les largeurs (L_{estime}) et longueurs (l_{estime}) réelles de l'objet au temps courant utilisées dans le calcul du centre de gravité. L'ordre d'apparition de ces deux droites dans la partition est fonction de la position relative du véhicule au capteur.

L'orientation du véhicule est donnée par le coefficient directeur « a » (équation III.3) de la droite (d) antérieurement calculée entre le premier et le dernier impact de la partition. Cette connaissance de l'orientation du véhicule nous permet d'estimer le centre de gravité de la partition. Car, elle donne une information de la

position en x du centre de gravité (X_{cdg}) relativement au point de rupture. Suivant la configuration, deux cas sont possibles. Soit on ajoute, à l'ordonnée à l'origine (b_1) la demi largeur de la droite (estimé (L_{estime}) par le détecteur ou du modèle (L_{modele}) de dimensions les plus petites), soit on la soustrait. Par contre, elle est inutile pour la position selon y du centre de gravité (Y_{cdg}), car le côté du véhicule observé est toujours le plus proche du véhicule porteur. Y_{cdg} sera donc toujours positionné une demi longueur de la droite (estimé (L_{estime}) par le détecteur ou du modèle (L_{modele}) de dimensions les plus petites) plus haut que la droite observée.

$$\begin{cases} b_{1_modele} = b_1 - L_{modele}/2 & \text{si } a < 0 \\ b_{1_modele} = b_1 + L_{modele}/2 & \text{si } a > 0 \\ b_{2_modele} = b_2 + l_{modele}/2 \end{cases} \quad \text{ou} \quad \begin{cases} b_{1_estime} = b_1 - L_{estime}/2 & \text{si } a < 0 \\ b_{1_estime} = b_1 + L_{estime}/2 & \text{si } a > 0 \\ b_{2_estime} = b_2 + l_{estime}/2 \end{cases}$$

On pose ($X_{cdg_modele}, Y_{cdg_modele}$) le centre de gravité estimé à partir des dimensions du modèle de véhicule le plus petit et ($X_{cdg_estime}, Y_{cdg_estime}$) le centre de gravité estimé à partir des dimensions réelles de la partition. Ces centres de gravité sont données par :

$$\begin{cases} X_{cdg_modele} = \text{mean}(a_1 * Y_1 + b_{1_modele}) \\ Y_{cdg_modele} = \text{mean}(a_2 * X_2 + b_{2_modele}) \end{cases} \quad \text{ou} \quad \begin{cases} X_{cdg_estime} = \text{mean}(a_1 * Y_1 + b_{1_estime}) \\ Y_{cdg_estime} = \text{mean}(a_2 * X_2 + b_{2_estime}) \end{cases}$$

Le module de détection transmet alors, à l'initialisation, au module de gestion de pistes le nombre de véhicules, les deux centres de gravité estimés (($X_{cdg_modele}, Y_{cdg_modele}$) et ($X_{cdg_estime}, Y_{cdg_estime}$)), et les deux modèles de dimensions qui en découlent.

Ainsi les algorithmes 3 et 4 présentés dans le chapitre III sont remplacés par les algorithmes 6 et 7 qui suivent :

Algorithm 6 Estimation du centre de gravité à partir d'un segment de droite pour le cas multi-objets

Require: $\mathcal{P}(n)$, la partition n ; (X, Y) , les éléments qui la composent, forment un segment de droite (d);

V_{modele_v} , le modèle de voiture, $l_{modele_v} = l_{min}$ sa longueur et $L_{modele_v} = L_{min}$ sa largeur; V_{modele_c} , le modèle de camion, $l_{modele_c} = l_{max}$ sa longueur et $L_{modele_c} = L_{max}$ sa largeur.

Ensure: le centre de gravité $C_{\mathcal{P}(n)}$ du véhicule représenté par ce segment de droite.

- 1: Calculer la largeur du segment de droite : $L_{droite} = |\min(X) - \max(X)|$
 - 2: **if** $L_{droite} < 1$ **then**
 - 3: $L_{droite} = L_{modele_v}$
 - 4: **end if**
 - 5: Calculer la position en X du centre de gravité : $C_{\mathcal{P}(n)}(X) = \min(X) + L_{droite}/2$
 - 6: Calculer la position en Y du centre de gravité modèle voiture : $C_{\mathcal{P}(n)}(Y) = \min(Y) + l_{modele_v}/2$
 - 7: Calculer la position en Y du centre de gravité modèle camion : $C_{\mathcal{P}(n)}(Y_{estime}) = \min(Y) + l_{modele_c}/2$
 - 8: $C_{\mathcal{P}(n)} \leftarrow [C_{\mathcal{P}(n)}(X), C_{\mathcal{P}(n)}(Y), C_{\mathcal{P}(n)}(Y_{estime})]$
-

Algorithm 7 Estimation du centre de gravité à partir de deux segments de droites pour le cas multi-objets 1

Require: $\mathcal{P}(n)$, la partition n ; (X_1, Y_1) et (X_2, Y_2) les éléments qui la composent, forment deux droites (d_1) et (d_2) ; V_{modele_v} , le modèle de voiture, $l_{modele_v} = l_{min}$ sa longueur et $L_{modele_v} = L_{min}$ sa largeur; V_{modele_c} , le modèle de camion, $l_{modele_c} = l_{max}$ sa longueur et $L_{modele_c} = L_{max}$ sa largeur; a , le coefficient directeur du segment de droite entre le premier et le dernier impact de la partition.

Ensure: le centre de gravité $C_{\mathcal{P}(n)}$ du véhicule représenté par ces segments de droites.

```

1: if  $|\min(X_1) - \max(X_1)| < 1$  then
2:   Calculer la largeur de la droite :  $L_{droite} = |\min(X_2) - \max(X_2)|$ 
3:   Calculer la longueur de la droite :  $l_{droite} = |\min(Y_1) - \max(Y_1)|$ 
4:   if  $L_{droite} < 1$  then
5:      $L_{droite} = L_{modele}$ 
6:   end if
7:   Estimer l'équation de la droite formée par les éléments de  $(X_1, Y_1)$  :  $Y_1 = a_1X_1 + b_1$ 
8:   Estimer l'équation de la droite formée par les éléments de  $(X_2, Y_2)$  :  $X_2 = a_2Y_2 + b_2$ 
9:   Translater la droite  $Y_1$  sur le centre de gravité modèle :  $b_{1\_modele} = b_1 + l_{modele_v}/2$ 
10:  Translater la droite  $Y_1$  sur le centre de gravité estimé :  $b_{1\_estime} = b_1 + l_{droite}/2$ 
11:  if  $a < 0$  then
12:    Translater la droite  $X_1$  sur le centre de gravité modèle :  $b_{2\_modele} = b_2 + L_{modele_v}/2$ 
13:    Translater la droite  $X_1$  sur le centre de gravité estimé :  $b_{2\_estime} = b_2 + L_{droite}/2$ 
14:  else
15:    Translater la droite  $X_1$  sur le centre de gravité modèle :  $b_{2\_modele} = b_2 - L_{modele_v}/2$ 
16:    Translater la droite  $X_1$  sur le centre de gravité estimé :  $b_{2\_estime} = b_2 - L_{droite}/2$ 
17:  end if
18:  Calculer la position en  $X$  du centre de gravité modèle :  $C_{\mathcal{P}(n)}(X_{modele}) = \text{mean}(a_1X_1 + b_{1\_modele})$ 
19:  Calculer la position en  $X$  du centre de gravité estimé :  $C_{\mathcal{P}(n)}(X_{estime}) = \text{mean}(a_1X_1 + b_{1\_estime})$ 
20:  Calculer la position en  $Y$  du centre de gravité modèle :  $C_{\mathcal{P}(n)}(Y_{modele}) = \text{mean}(a_2Y_2 + b_{2\_modele})$ 
21:  Calculer la position en  $Y$  du centre de gravité estimé :  $C_{\mathcal{P}(n)}(Y_{estime}) = \text{mean}(a_2Y_2 + b_{2\_estime})$ 
22: else
23:   Calculer la largeur de la droite :  $L_{droite} = |\min(X_1) - \max(X_1)|$ 
24:   if  $L_{droite} < 1$  then
25:      $L_{droite} = L_{modele}$ 
26:   end if
27:   Estimer l'équation de la droite formée par les éléments de  $(X_1, Y_1)$  :  $X_1 = a_1Y_1 + b_1$ 
28:   Estimer l'équation de la droite formée par les éléments de  $(X_2, Y_2)$  :  $Y_2 = a_2X_2 + b_2$ 
29:   Translater la droite  $Y_2$  sur le centre de gravité modèle :  $b_{2\_modele} = b_2 + l_{modele_v}/2$ 
30:   Translater la droite  $Y_2$  sur le centre de gravité estimé :  $b_{2\_estime} = b_2 + l_{droite}/2$ 
31:   if  $a < 0$  then
32:     Translater la droite  $X_1$  sur le centre de gravité modèle :  $b_{1\_modele} = b_1 + L_{modele_v}/2$ 
33:     Translater la droite  $X_1$  sur le centre de gravité estimé :  $b_{1\_estime} = b_1 + L_{droite}/2$ 
34:   else
35:     Translater la droite  $X_1$  sur le centre de gravité modèle :  $b_{1\_modele} = b_1 - L_{modele_v}/2$ 
36:     Translater la droite  $X_1$  sur le centre de gravité estimé :  $b_{1\_estime} = b_1 - L_{droite}/2$ 
37:   end if
38:   Calculer la position en  $X$  du centre de gravité modèle :  $C_{\mathcal{P}(n)}(X_{modele}) = \text{mean}(a_1Y_1 + b_{1\_modele})$ 
39:   Calculer la position en  $X$  du centre de gravité estimé :  $C_{\mathcal{P}(n)}(X_{estime}) = \text{mean}(a_1Y_1 + b_{1\_estime})$ 
40:   Calculer la position en  $Y$  du centre de gravité modèle :  $C_{\mathcal{P}(n)}(Y_{modele}) = \text{mean}(a_2X_2 + b_{2\_modele})$ 
41:   Calculer la position en  $Y$  du centre de gravité estimé :  $C_{\mathcal{P}(n)}(Y_{estime}) = \text{mean}(a_2X_2 + b_{2\_estime})$ 
42: end if
43:  $C_{\mathcal{P}(n)} \leftarrow [C_{\mathcal{P}(n)}(X_{reel}), C_{\mathcal{P}(n)}(Y_{reel}), C_{\mathcal{P}(n)}(X_{modele}), C_{\mathcal{P}(n)}(Y_{modele})]$ 

```

Adaptation de l'étape de gestion de pistes

Le module de gestion de pistes est chargé de gérer les apparitions et disparitions des objets dans le champ de vision du capteur. À cette fin, il associe les centres de gravité détectés (par le détecteur) aux centres de gravité estimés (par les filtres, du module DAT). Cependant, dans le cas multi-objets, deux centres de gravité sont estimés et transmis au module de gestion de pistes. Ce double calcul va permettre au module de gestion de pistes de gérer les changements de dimensions réalisés par le filtre VS-ITPF.

Pour pallier les difficultés dues à des dimensions de véhicule très importantes, la prise en compte du modèle doit avoir lieu après un temps de convergence du suivi. Cela implique que dans un premier temps, le véhicule qui apparaît, dans le champ de vision du capteur, prend les dimensions du modèle de véhicule le plus petit et par la suite, les dimensions sont ré-estimées aux dimensions observées (c'est-à-dire les dimensions extraites par le détecteur d'objet). Ainsi, entre deux instants $t - 1$ et t , il peut y avoir ré-estimation des dimensions impliquant une translation du centre de gravité estimé. C'est ce changement de dimensions qui doit être pris en compte par le gestionnaire de pistes au niveau de la phase d'association.

Dans le filtre VS-ITPF, la modification des dimensions du modèle intervient après un temps de convergence connu, dans notre cas 0,5s. Une variable booléenne, appelé « flag », permet de savoir si les dimensions du modèle ont été modifiées dans le filtre VS-ITPF. Ainsi, dans le module de gestion de pistes un test sur ces deux paramètres permet de connaître le centre de gravité (estimé à partir du modèle de dimensions les plus petites ou du modèle de dimensions réelles) à utiliser pour réaliser au mieux la phase d'association. Dans notre méthode, tant que la modification des dimensions n'a pas eu lieu, l'association est réalisée à partir des dimensions du modèle de véhicule le plus petit. Lorsque les dimensions du modèle ont été mises à jour une fois, le calcul de l'association se fera, jusqu'à la disparition du véhicule, à partir des dimensions du modèle réel. Nous avons choisi de comparer la distance euclidienne (algorithme 8) entre les deux centres de gravité (estimés par le DAT et celui du détecteur) à une valeur seuil afin de réaliser l'association.

Algorithm 8 Estimation de la distance Euclidienne pour la gestion des pistes

Require: $(X_{cdg_modele}, Y_{cdg_modele})$ le centre de gravité du modèle le plus petit ; $(X_{cdg_estime}, Y_{cdg_estime})$ le centre de gravité estimé lors de l'étape de détection ; $(X_{cdg_ITPF}, Y_{cdg_ITPF})$ le centre de gravité estimé par le filtre ITPF et flag une variable booléenne égale à 0 si la dimension de l'objet est inchangée et 1 si la dimension est changée.

Ensure: Distance Euclidienne ($d_{euclidien}$)

- 1: **if** temps de convergence de la piste > 0,5s et flag= 1 **then**
 - 2: $d_{euclidien} = \sqrt{(X_{cdg_ITPF} - X_{cdg_estime})^2 + (Y_{cdg_ITPF} - Y_{cdg_estime})^2}$
 - 3: **else**
 - 4: $d_{euclidien} = \sqrt{(X_{cdg_ITPF} - X_{cdg_modele})^2 + (Y_{cdg_ITPF} - Y_{cdg_modele})^2}$
 - 5: **end if**
-

La suite du programme reste identique à la méthode présentée chapitre III, avec création ou élimination de pistes suivant les résultats de la phase d'association.

IV.2.3 Résumé des algorithmes du module de surveillance dans le cas multi-objets hétérogènes

Nous synthétisons les différents algorithmes à réaliser dans le module de surveillance en prenant comme référence le chapitre III.

- L'algorithme 1 de partitionnement de données reste inchangé.
- L'algorithme 2 de segmentation de données voit ses procédures 3 et 4 respectivement remplacées par les procédures 6 et 7.
- L'algorithme 5 de gestion de pistes est identique (le calcul de $d_{euclidien}$ étant présenté algorithme 8).

IV.2.4 Adaptation du module DAT au cas multi-objets hétérogènes

Le module DAT est composé d'un ensemble de filtres VS-ITPF réalisant chacun le suivi d'un objet. Le filtre VS-ITPF, à partir des informations en provenance du module de surveillance, autorise ou non la modification de la longueur : il autorise uniquement son augmentation de sorte à éviter les pertes d'informations dues à des occultations. Cette modification de longueur va entraîner une translation du centre de gravité.

La modification de la longueur interviendra avant le calcul du poids du filtre VS-ITPF, car ce calcul nécessite une connaissance fiable du modèle. En effet, lors de cette étape les « véhicules particuliers » sont construits à partir de leur centre de gravité, de la rotation du véhicule (obtenue à partir de la vitesse estimé de celui-ci) et du modèle de véhicule. Le filtre effectuera un test avant de décider de la longueur du modèle :

- Si la longueur du modèle actif est inférieure à la longueur du modèle de dimensions réelles (en sortie du détecteur) alors la longueur du modèle peut être modifiée. À l'inverse si la longueur du modèle actif est supérieure à la longueur du modèle de dimensions réelles alors la longueur n'est pas modifiée pour éviter les pertes d'information dues à des cas de masquage.

Si cette condition est remplie alors la longueur est modifiée. La modification de la longueur entraîne une translation du centre de gravité qui est fonction des dimensions du véhicule. Le principe est le même pour la largeur du véhicule. Le schéma IV.1 montre la translation du centre de gravité en fonction de la variation de la longueur et du sens de rotation du véhicule. Dans notre cas, le véhicule effectue une rotation horaire. L'algorithme est détaillé ci-après (algorithme 9).

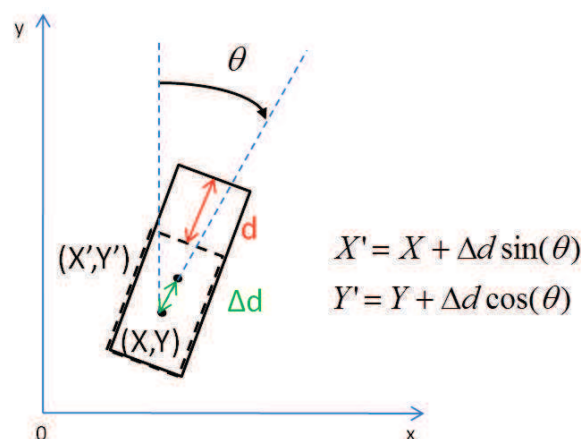


FIGURE IV.1. Calcul du nouveau centre de gravité après modification du modèle véhicule

Algorithm 9 Translation du centre de gravité de l'objet lors d'un changement de dimension

Require: $(X_{cdg_ITPF}, Y_{cdg_ITPF})$ le centre de gravité estimé par le filtre VS-ITPF ; flag une variable booléenne égale à 0 si la dimension de l'objet est inchangée et 1 si la dimension est changée ; dimension courante de l'objet ; dimension réelle de l'objet transmise par le détecteur d'objet ; θ l'angle de rotation du véhicule ; Δd la longueur de déplacement du centre de gravité

Ensure: Centre de gravité du filtre VS-ITPF $(X_{cdg_ITPF}, Y_{cdg_ITPF})$ recalé après changement de dimension de l'objet

- 1: **if** dimensions courantes < dimensions réelles **then**
- 2: dimensions courantes = dimensions réelles
- 3: $X_{cdg_ITPF} = X_{cdg_ITPF} + \Delta d \sin(\theta)$
- 4: $Y_{cdg_ITPF} = Y_{cdg_ITPF} + \Delta d \cos(\theta)$
- 5: flag = 1
- 6: **end if**

Ensuite, le filtre conserve la même structure que dans le chapitre III (reconstruction des particules, calcul du poids, calcul de l'estimé, rééchantillonnage et enfin prédiction du Kalman).

IV.2.5 Résultats expérimentaux

Il s'agit ici de montrer les performances de notre méthode de détection et de suivi d'objets mise en place, sur un exemple multi-objets hétérogènes. Dans la section IV.2.5.1, les mesures sont issues du simulateur de trajectoire de la thèse (cf. annexe A.2). Ce simulateur ne prend pas en compte l'environnement extérieur aux voies de circulation. Ainsi, seuls les objets à suivre (dans notre cas des véhicules types camion ou voiture) sont à l'origine de mesures. Ce simulateur permet de calibrer précisément les performances en suivi et en estimation. Nous avons simulé des mesures en plaçant des objets, dont les dimensions moyennes sont équivalentes à celles de véhicule de types voiture, de largeur 1,8m et de longueur 2,6m, ou camion, de largeur 2,6m et de longueur 13m, dans le champ de vision du capteur. Les mesures sont ensuite bruitées, par une erreur d'écart-type de 0,03m. Dans la section IV.2.5.3, les mesures sont issues du simulateur SiVIC (cf. annexe A.2) où les mesures capteur sont très proche de la réalité. Dans cet exemple, la simulation est réalisée à partir de deux modèles de voiture : une Mini et un Espace.

————— Simulateur de la thèse —————

IV.2.5.1 Scénario et caractéristiques du capteur pour le simulateur de la thèse

Pour notre expérimentation, les caractéristiques du capteur sont définies par une ouverture angulaire de $(-80^\circ, 80^\circ)$, avec une résolution angulaire de $\Delta\theta = 0,25^\circ$. La plage de visibilité du capteur est de 10m pour la portée minimale et de 110m pour la portée maximale.

Nous nous intéressons au scénario suivant : la route est constituée de trois voies de circulation. Le véhicule porteur du capteur évolue sur la voie centrale durant tout le scénario (sa vitesse selon la direction x est nulle). Sa vitesse selon la direction y est de $15\text{m}\cdot\text{s}^{-1}$. La durée de la simulation est fixée à 15s avec un échantillonnage temporel de $\Delta t = 0,05\text{s}$. Ces données sont résumées dans le tableau IV.1.

Le scénario est tel que trois véhicules apparaissent dans le champ de vision du véhicule porteur du capteur à des instants différents. À tout moment, à partir de la première apparition, au moins un véhicule est présent dans ce champ. Le véhicule situé sur la voie de gauche apparaît au temps $t = 6\text{s}$, c'est une voiture qui a une trajectoire rectiligne. Les deux autres véhicules changent de voie au cours du scénario. Le véhicule le plus à droite est un camion qui apparaît à $t = 70\text{s}$. L'autre véhicule est une voiture qui apparaît à $t = 3\text{s}$. Chacun de ces véhicules a une vitesse égale sinon supérieure à celle du véhicule porteur du capteur. De

	Notation	Valeur
Capteur	(V_x, V_y) en ms^{-1}	(0, 15)
	Num_voie	2
	$(\theta_{min}, \theta_{max})$ en $^{\circ}\text{C}$	(-80, 80)
	(d_{min}, d_{max}) (m)	(10, 110)
	$(\Delta\theta, \Delta t)$ en $^{\circ}\text{C}$ et s	(0,25, 0,05)
	tir_infini en m	250

Tableau IV.1. Paramètres de simulation

ce fait, au cours de la simulation, ils s'éloignent du véhicule porteur. Les paramètres d'initialisation de la simulation sont résumés dans le tableau IV.2. Durant la simulation, ces paramètres évoluent pour initier des changements de voies, des accélérations et autres, ces changements ne figurent pas dans le tableau. Sur la figure IV.2, les trajectoires des véhicules de ce scénario sont représentées en bleu. Les mesures pour chacun des véhicules sont présentées en noir aux instants $t = 0,5\text{s}$, $t = 5\text{s}$, $t = 10\text{s}$ et $t = 14,5\text{s}$. On remarque que pour les deux véhicules situés les plus à gauche tous les instants de mesures ne sont pas représentés. Cela est dû à leur apparition tardive, en effet, ils apparaissent respectivement à 6s et 3s (de gauche à droite). De même le véhicule le plus à gauche est occulté par le second, ce qui explique que les seules mesures lui faisant référence sont celles de l'instant $t = 14,5\text{s}$.

	Notation	Valeur
Route	Nb_voie	3
	$largeur_voie$ en m	3,5
Cible 1	Num_voie	1
	$dateapparition$ en s	3
	(V_x, V_y) en ms^{-1}	(0, 0)
	$(a_x, a_y, Bruit_sur_acc)$ en ms^{-2}	(0, 0,3, 0,03)
	$d_vehicule_capteur$ en m	20
Cible 2	Num_voie	2
	$dateapparition$ en s	0
	(V_x, V_y) en ms^{-1}	(0, 0)
	$(a_x, a_y, Bruit_sur_acc)$ en ms^{-2}	(0, 0,2, 0,03)
	$d_vehicule_capteur$ en m	20
Cible 3	Num_voie	1
	$dateapparition$ en s	6
	(V_x, V_y) en ms^{-1}	(0, 0)
	$(a_x, a_y, Bruit_sur_acc)$ en ms^{-2}	(0, 0,5, 0,03)
	$d_vehicule_capteur$ en m	40

Tableau IV.2. Paramètres de simulation initiaux des véhicules cibles relativement à la route et au véhicule porteur

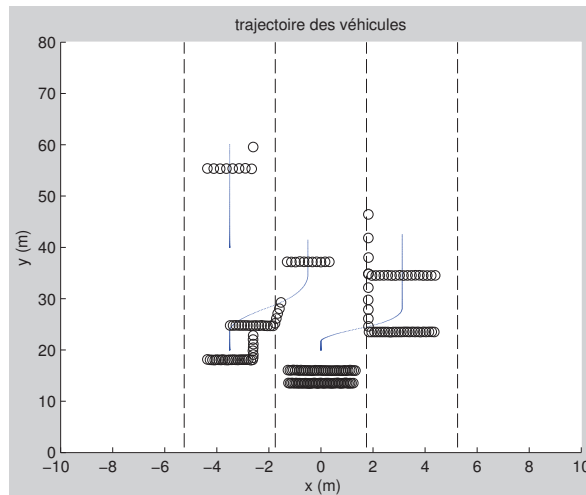


FIGURE IV.2. Illustration des trajectoires relatives au véhicule porteur pour le scénario (défini par les tableaux IV.1 et IV.2), constitué d'un camion et de deux voitures

IV.2.5.2 Résultat de l'approche multi-objets pour les données du simulateur de la thèse

La figure IV.3 présente une comparaison du nombre de pistes actives avec le nombre de véhicules détectés à chaque instant. Nous constatons une continuité du suivi aux environs de l'instant $t = 9$ s. Cela est dû à la structure de traitement mise en place. Le filtre associé à une piste continue de fonctionner tant qu'il y a des mesures pour l'alimenter. De ce fait, même si une piste n'est plus détectée par le module de surveillance (cas de masquage partiel, disparition progressive du champ de vision du capteur...), le filtre peut continuer à fonctionner dès lors qu'il identifie des mesures compatibles avec la piste associée. C'est pourquoi, à l'instant $t = 9$ s, un véhicule continue à être suivi alors qu'il n'est plus détecté. C'est le véhicule situé en haut de la voie de gauche sur la figure IV.4 (pour les temps $t = 7,5$ s et $t = 14$ s) qui disparaît et réapparaît, au cours de la simulation.

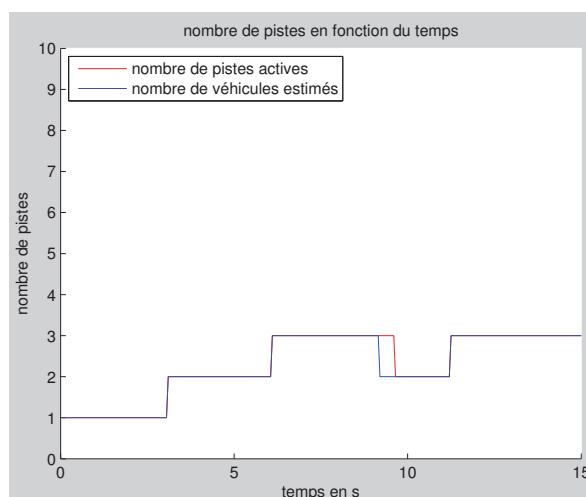


FIGURE IV.3. Comparaison du nombre de pistes actives avec le nombre de véhicules estimé par le détecteur au cours de la simulation

La figure IV.4 montre la perception de la scène par le lidar respectivement aux temps $t = 5s$, $t = 7,5s$, $t = 11s$ et $t = 14s$. Pour chacun de ces temps, les mesures lidar en rouge ont été superposées à l'estimation faite par les filtres VS-ITPF associés à chaque véhicule détecté. Cette estimation est représentée par le modèle objet : une boîte englobante, bleue, aux dimensions du véhicule suivi. Ce modèle objet est obtenu à partir de l'estimation, par le filtre VS-ITPF, du centre de gravité et des vitesses de l'objet suivi, ainsi que par ces dimensions. On remarque que les mesures sont traversées par le modèle objet aussi bien pour le camion (véhicule le plus à droite sur les figures) que pour les deux voitures (les deux véhicules de gauche). L'orientation du modèle objet est identique à celle des mesures pour le camion et les voitures. L'orientation du véhicule étant obtenue à l'aide de la vitesse propre au véhicule. Nous pouvons ainsi dire que l'estimation faite par le filtre fonctionne aussi bien pour un camion que pour une voiture. Nous confirmons cette tendance avec les courbes de vitesse, d'accélération et d'erreur de chacun des véhicules de la simulation présentées ci-dessous.

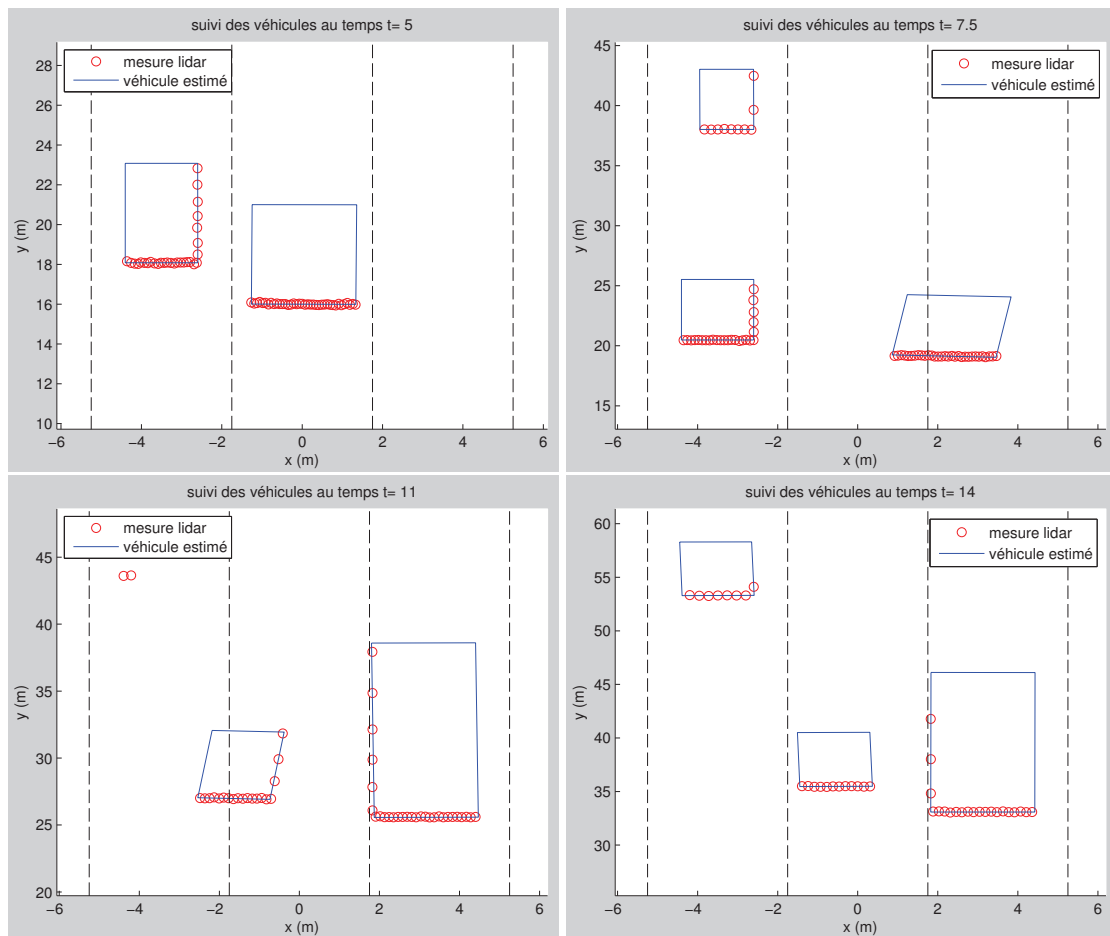


FIGURE IV.4. Perception du lidar aux temps $t = 5s$ (figure haut gauche), $t = 7,5s$ (figure haut droite), $t = 11s$ (figure bas gauche) et $t = 14s$ (figure bas droite) superposée aux modèles de véhicules estimés par le filtre ITPF, pour un cas multi-objets

La figure IV.5 présente les vitesses à gauche et accélérations à droite de chacun des véhicules de la simulation. Après un temps de convergence court pouvant aller de 0,5s à 1s, les vitesses et accélérations convergent vers celles de référence. Par la suite, les variations observées semblent faibles autour des vitesses et accélérations de référence. Un léger décalage de l'estimation est, cependant, observé lorsque l'accélération est brutalement modifiée. Cela est dû au fait que les changements de vitesse et d'accélération ne sont pas instantanément pris en compte par le filtre, ce qui provoque une perturbation de l'estimé, qui après un temps de convergence se repositionne correctement sur les courbes de référence. Ce changement d'accélération a un impact sur la vitesse de l'objet, cependant, l'estimation en est moins perturbée, elle est presque collée aux courbes de référence. Après convergence de l'estimation, on ne remarque pas de décrochage du suivi. Nous pouvons conclure que la méthode est stable et ce peu importe le type de véhicule suivi. Les bons résultats au niveau du suivi des véhicules et de l'accélération se retrouvent sur la figure IV.4 où les modèles objets sont parfaitement superposés aux mesures. Ils sont par ailleurs confirmés par les courbes d'erreur de chaque véhicule présentées figure IV.7. Dans le tableau IV.3, nous présentons les écart-types moyens de chacun des paramètres (position, vitesse et accélération) sur une période où le suivi a convergé.

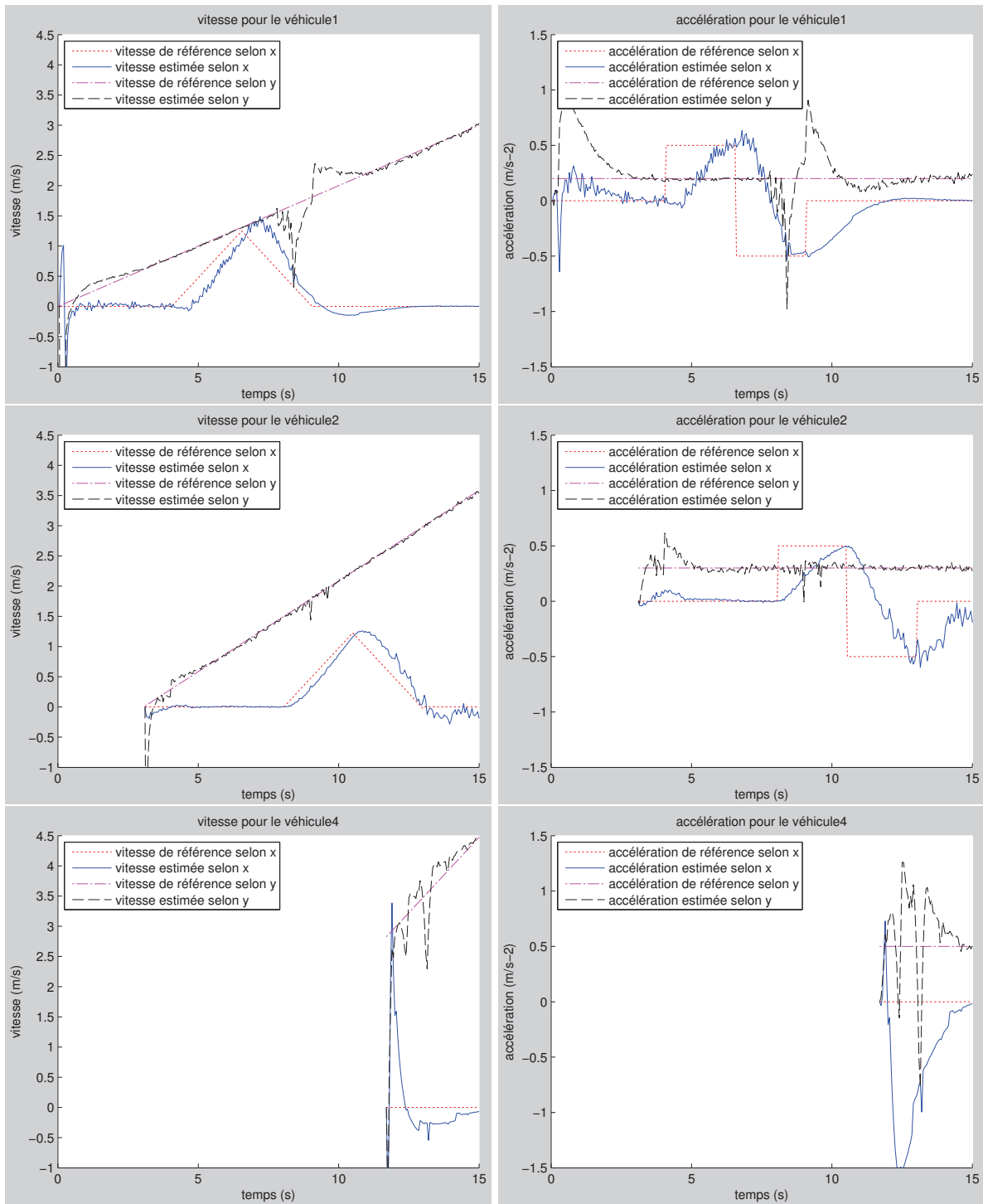


FIGURE IV.5. Courbes des vitesses à gauche et accélération à droite en fonction du temps, du camion situé à droite (figure du haut), du véhicule central (figure du centre) et du véhicule de gauche (figure du bas), pour un cas multi-objets

La figure IV.6 présente l'évolution de la largeur et longueur estimées des boîtes englobantes pour chacun des véhicules présents dans la scène. On remarque que sur la figure IV.6 en haut, le véhicule a une longueur estimée de 5m de $t = 0s$ à $t = 8s$, puis progressivement sa longueur évolue jusqu'à atteindre 13m. Sa largeur, quant à elle, est directement initialisée à 2,3m, largeur réelle du véhicule en raison de sa forte observabilité dans ce scénario. Les figures IV.6 en bas à gauche et à droite, correspondent aux deux voitures de la simulation. Les deux véhicules n'étant pas visibles en début de simulation leurs largeur et longueur ne sont pas estimées dans un premier temps. Par la suite, dès que la voiture apparaît les dimensions correctes sont rapidement estimées. En raison de la rotation des véhicules dont les résultats sont présentés en bas de la figure, la longueur estimée des boîtes englobantes de ces véhicules est supérieure à leur longueur réelle.

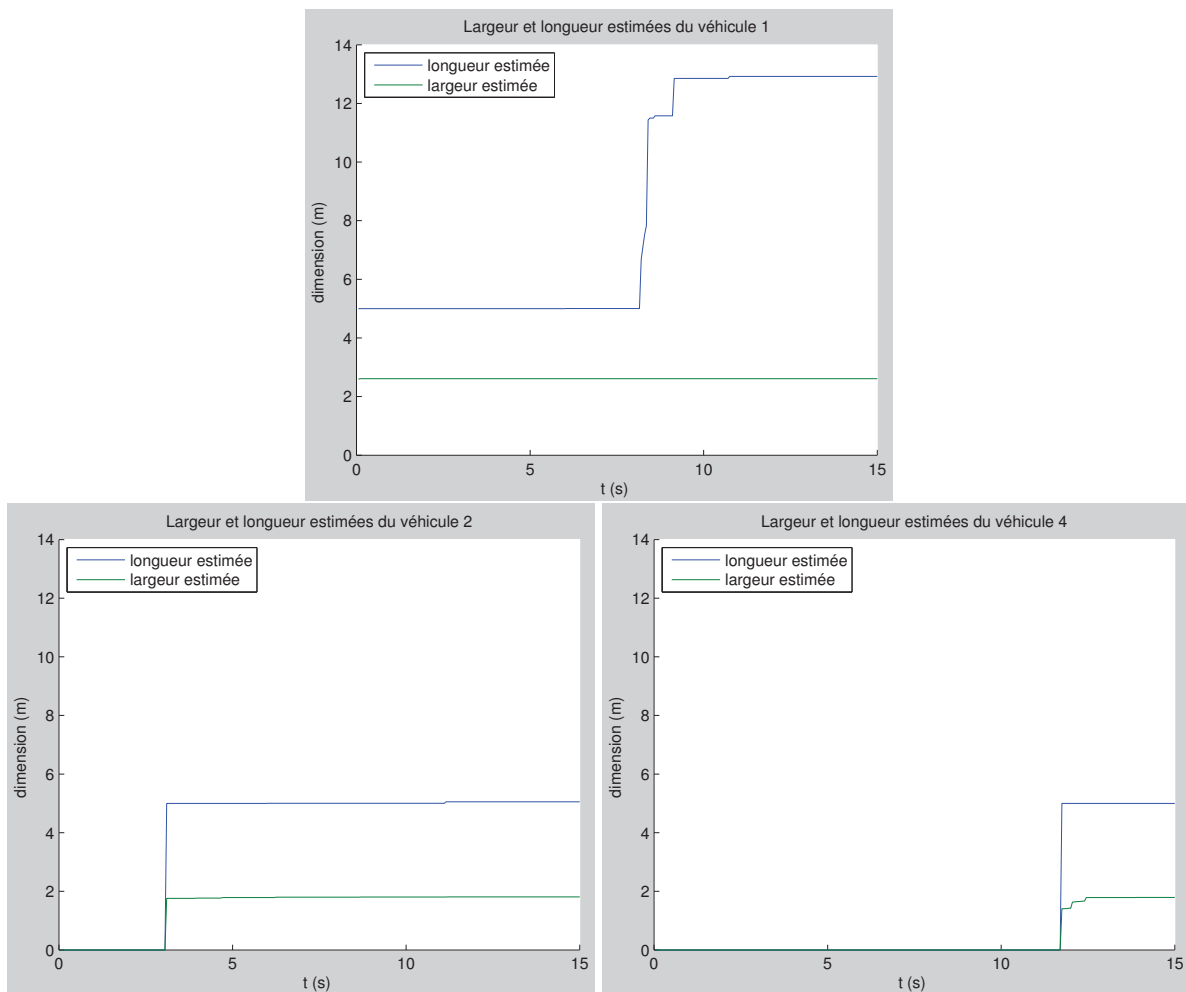


FIGURE IV.6. Longueur et largeur estimées des boîtes englobantes des véhicules de la simulation en fonction du temps, en haut le camion et en bas les deux voitures

La figure IV.7 présente l'erreur sur la position des véhicules cibles au cours du temps. Dans le cas de notre suivi multi-objets hétérogènes, le filtre ne connaît pas toujours la taille exacte des objets suivis. Dans ces circonstances, calculer l'erreur en position des objets, à partir de leur centre de gravité n'a aucun sens, malgré la connaissance fiable de cette position, pour chaque véhicule. En effet, les dimensions réelles du véhicule sont prises en compte après un certain temps de convergence. Dans le cas de cette simulation, le camion débute sa course dans la voie centrale, seul le bas du véhicule est alors visible. Lorsqu'il change de voie, deux côtés sont visibles et c'est à cet instant que la longueur réelle est prise en compte. Ainsi, le centre de gravité estimé au début de la simulation ne correspond pas au centre de gravité réel (de référence). Les résultats du suivi ne sont pas impactés par cette différence, car, pour ce véhicule, le modèle en début de simulation (c'est le modèle aux dimensions plus petites) est différent de celui en fin de simulation (le modèle à dimensions visibles du véhicule). Réaliser le calcul de l'erreur en fonction des centres de gravité ne serait donc pas réaliste dans ce cas.

Par conséquent, le calcul de l'erreur est réalisé à partir des mesures brutes appartenant au véhicule (sans le bruit d'échantillonnage spatial du capteur). Pour cela, une moyenne des mesures brutes appartenant à chaque véhicule est réalisée. Le centre de gravité estimé par le filtre ITPF, la vitesse et les dimensions du véhicule permettent de construire une estimation des impacts sur le véhicule. Une moyenne des impacts du véhicule estimé est alors réalisée. L'erreur est calculée à partir de ces moyennes.

La figure IV.7 montre les courbes d'erreur sur la position obtenue selon ce mode opératoire. La courbe en haut à gauche représente l'erreur du camion (véhicule situé le plus à droite figures IV.4). La courbe en haut à droite représente l'erreur d'une voiture en situation de changement de voie (véhicule situé au centre figures IV.4). Enfin, la courbe du bas représente l'erreur d'une voiture ayant une trajectoire rectiligne (véhicule situé le plus à gauche figures IV.4). L'erreur sur chacun des véhicules est faible, après convergence, elle est bornée à $\pm 0,1\text{m}$. On remarque sur la figure en haut à gauche, concernant le camion, que peu de temps après le début du changement de trajectoire (de $t = 4\text{s}$ à $t = 9\text{s}$), il y a une augmentation rapide de l'erreur. Après quelques secondes, l'erreur a de nouveau convergé. Cette augmentation est liée au changement important de dimension du modèle. Malgré la compensation de l'erreur liée à la translation du centre de gravité de l'objet, il y a une forte augmentation de l'erreur lors du changement de dimension. Cette augmentation est liée aux vitesse et accélération de l'objet. En effet, la vitesse de l'objet s'est adaptée pour compenser la dimension erronée du modèle. Lors du changement de dimension, la position du centre de gravité de l'objet a été translatée pour éviter une augmentation de l'erreur (due à une mauvaise position du centre de gravité) de l'objet suivi, mais l'augmentation très importante de taille a malgré tout un impact sur l'estimation de la position de l'objet. Cette erreur tend à diminuer à nouveau lorsque le filtre VS-ITPF a de nouveau estimé correctement les dimensions du véhicule. Sur la figure de droite, où un changement de trajectoire a également lieu, l'augmentation de l'erreur est moins importante. Enfin, pour le véhicule du bas l'augmentation de la période de convergence est la conséquence du nombre d'impacts faible. D'où une période de convergence un peu plus longue que pour les deux autres véhicules. Dans ce cas, le véhicule étant masqué, il apparaît progressivement au cours du temps. Le faible nombre de mesures se ressent alors sur le suivi.

Le tableau IV.3 montre une estimation de l'écart-type sur la position, la vitesse et l'accélération pour la méthode VS-ITPF, sur une période où le suivi a convergé, dans le cas d'un suivi multi-objets hétérogènes. La méthode VS-ITPF présente une forte stabilité et l'erreur reste faible en présence d'objets de différentes tailles. On constate en effet qu'elle est peu sensible à la taille des objets à suivre.

Écart type	Méthode VS-ITPF
Position	$6,36.10^{-2}m$
Vitesse	$1,48.10^{-1}m.s^{-1}$
Accélération	$1,52.10^{-1}m.s^{-2}$

Tableau IV.3. Écart-type de l'erreur d'estimation des positions, vitesses et accélérations pour les véhicules cibles sur une période de convergence de la méthode VS-ITPF appliquées aux données du simulateur de la thèse, dans un cas multi-objets

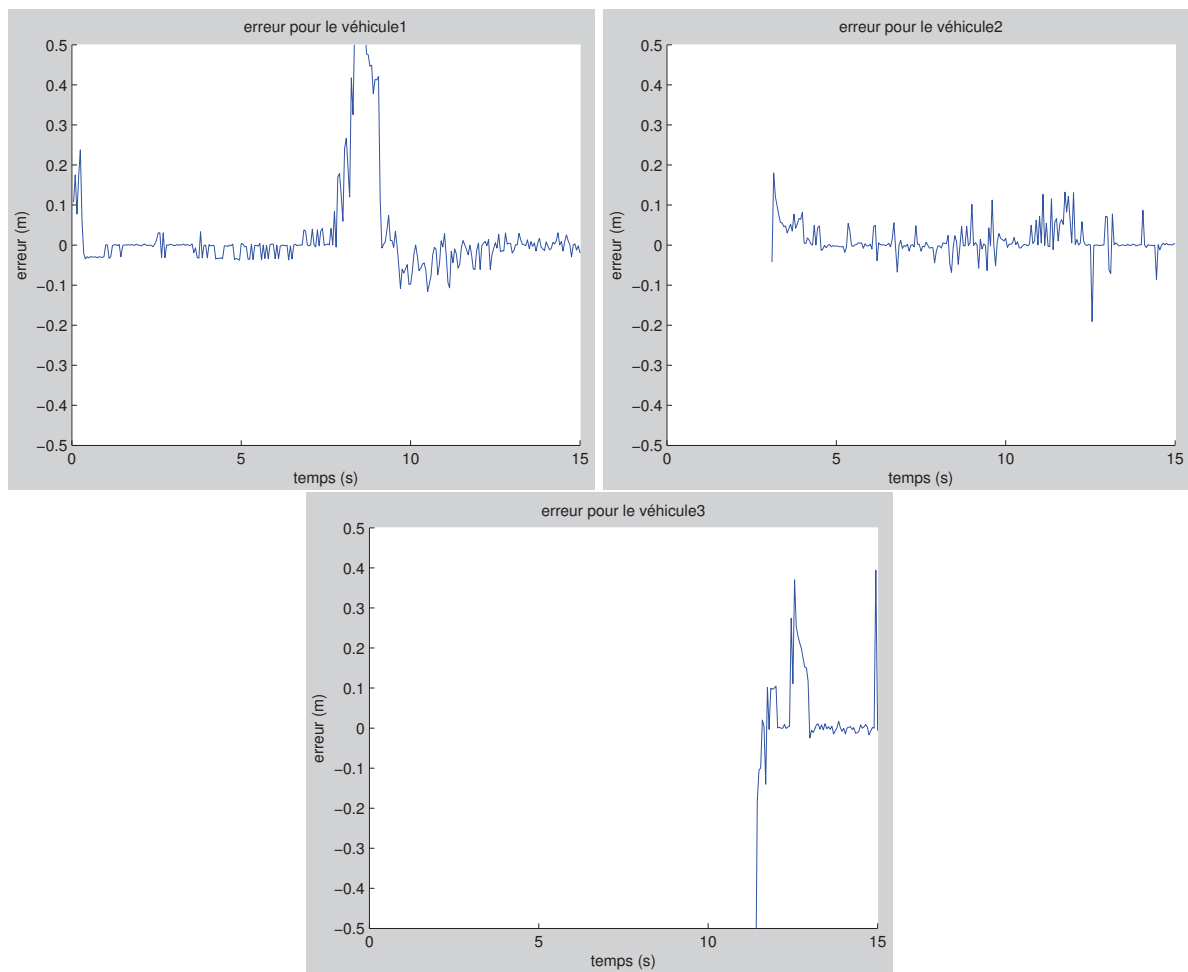


FIGURE IV.7. Courbes d'erreur sur la position en fonction du temps, du camion situé voie de droite (figure haut gauche), pour la voiture voie centrale (figure haut droite) et pour la voiture voie de gauche (figure du bas)

IV.2.5.3 Scénario et caractéristiques du capteur pour le simulateur SiVIC

Pour notre expérimentation, les caractéristiques du capteur sont définies par une ouverture angulaire de $(-80^\circ, 80^\circ)$, avec une résolution angulaire de $\Delta\theta = 0,25^\circ$. La plage de visibilité du capteur est de 10m pour la portée minimale et de 180m pour la portée maximale.

Nous nous intéressons au scénario suivant : la route est constituée de trois voies de circulation. Le véhicule porteur évolue sur la voie centrale durant tout le scénario (sa vitesse selon la direction x est nulle). Sa vitesse selon la direction y est de $20\text{m}\cdot\text{s}^{-1}$. La durée de la simulation est fixée à 15s avec un échantillonnage temporel $\Delta t = 0,05\text{s}$. Ces données sont résumées dans le tableau IV.4.

	Notation	Valeur
Capteur	(V_x, V_y) en ms^{-1}	(0, 20)
	Num_voie	2
	$(\theta_{min}, \theta_{max})$ en $^\circ\text{C}$	(-80, 80)
	(d_{min}, d_{max}) (m)	(10, 180)
	$(\Delta\theta, \Delta t)$ en $^\circ\text{C}$ et s	(0,25, 0,05)
	tir_infini en m	180

Tableau IV.4. Paramètres de simulation

Le scénario est tel que deux véhicules (un Espace et une Mini) apparaissent à des instants différents, et au cours de celui-ci, au moins un véhicule est toujours présent dans la scène. Le véhicule situé sur la voie gauche (la Mini) apparaît au temps $t \simeq 3\text{s}$, selon une trajectoire rectiligne. Le second véhicule (l'Espace) est situé sur la voie droite et change de voie au cours de la simulation. Ces trajectoires sont représentées en bleu sur la figure IV.8. Les figures IV.9 décrivent le scénario figure IV.8 à l'aide d'image de la scène aux instants $t = 1\text{s}$, $t = 4\text{s}$, $t = 7,5\text{s}$ et $t = 10\text{s}$. Elles illustrent également les exemples présentés figures IV.10 et IV.12.

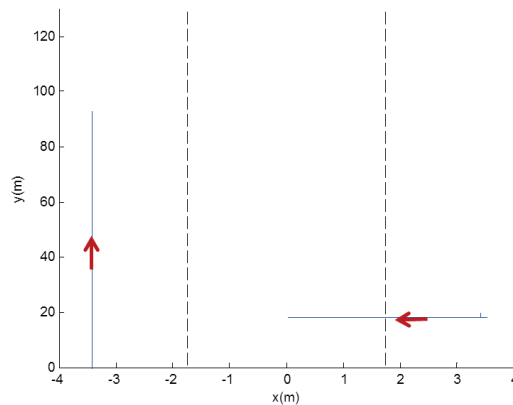


FIGURE IV.8. Illustration des trajectoires relatives au véhicule porteur pour le scénario du simulateur SiVIC



FIGURE IV.9. Quatre images de la scène au temps $t = 1s$, $t = 4s$, $t = 7,5s$ et $t = 10s$

IV.2.5.4 Résultat de l'approche VS-ITPF pour les données du simulateur SiVIC

La figure IV.10 montre la perception de la scène par le lidar respectivement aux temps $t = 1s$, $t = 4s$, $t = 7,5s$ et $t = 10s$. Sur chacune de ces figures, les mesures lidar en rouge ont été superposées à l'estimation faite par les filtres VS-ITPF associés à chaque véhicule détecté. Cette estimation est représentée par le modèle objet, une boîte englobante, aux dimensions du véhicule suivi, en bleu sur les figures. Ce modèle objet est obtenu à partir de l'estimation, par le filtre VS-ITPF, du centre de gravité, des vitesses de l'objet suivi, ainsi que par ses dimensions. On remarque la description correcte des mesures par le modèle aussi bien pour l'Espace (véhicule le plus à droite sur les figures) que pour la Mini (située à gauche). L'orientation du modèle objet est cohérente avec les mesures pour les deux véhicules et notamment pour l'Espace, dont l'orientation est modifiée par le changement de trajectoire (figure IV.10 en bas à gauche où la boîte englobante est parfaitement superposée aux mesures). Sur la figure IV.10 en bas à droite, on remarque que la largeur de la boîte englobante représentant l'estimation de la Mini en haut à gauche, est plus grande que la largeur des mesures. Cela provient de deux causes, la première est le masquage partiel du véhicule par l'Espace (ayant changé de voie), la seconde est l'incapacité du filtre à diminuer les dimensions du modèle (par construction) afin d'éviter toute perte d'informations (notamment dues à la sortie du champ de vision du capteur). Nous pouvons ainsi constater que l'estimation faite par le filtre fonctionne sur des modèles de type différent.

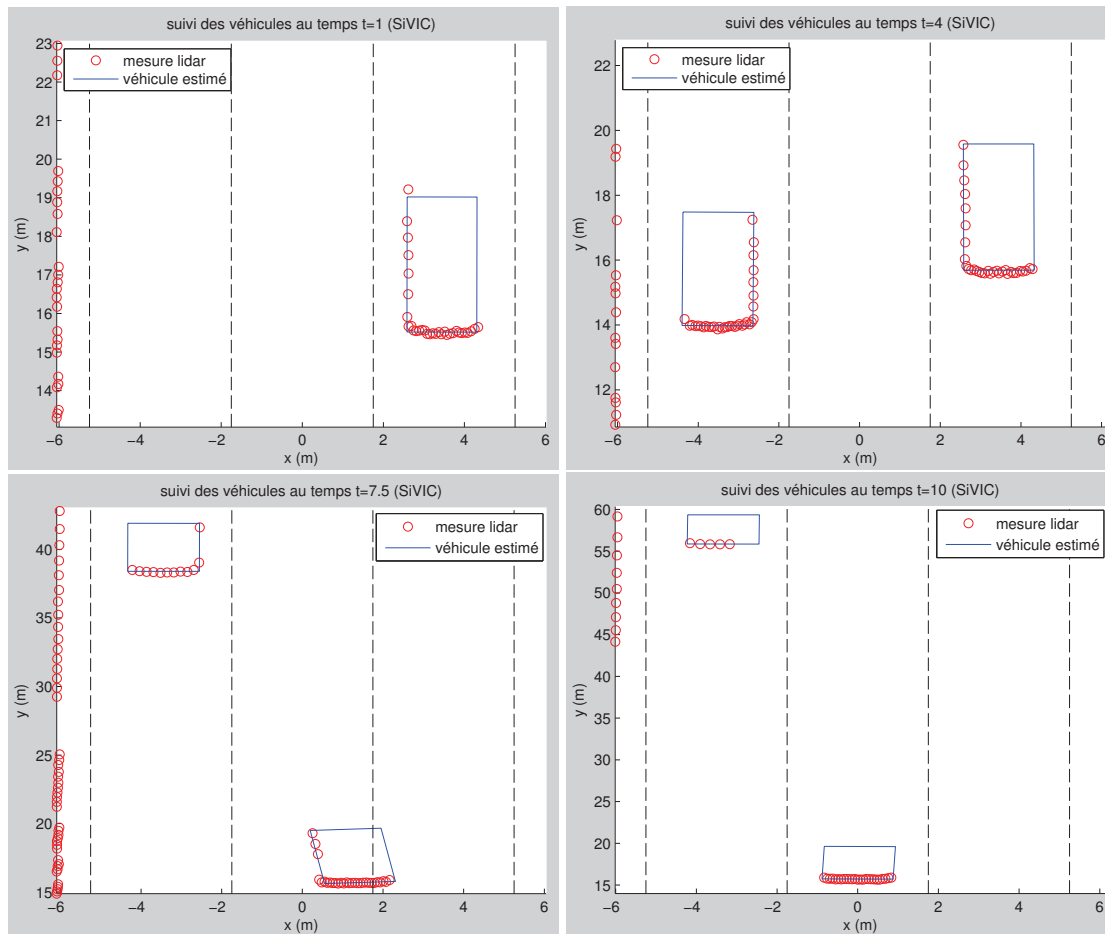


FIGURE IV.10. Scènes aux instants $t = 1$ s (figure haut gauche), $t = 4$ s (figure haut droite), $t = 7,5$ s (figure bas gauche) et $t = 10$ s (figure bas droite) et véhicules estimés par le filtre VS-ITPF

La figure IV.11 présente l'évolution de la largeur et longueur de chacun des véhicules présents. On remarque plusieurs sauts modifiant la largeur ou la longueur des véhicules. Le véhicule figure IV.11 de gauche est l'Espace. Sa longueur maximale estimée est de 3,9m et sa largeur est de 1,79m, ces valeurs sont en deçà des dimensions réelles d'un espace. Cela s'explique en particulier par le manque de mesures décrivant l'objet (mesures distribuées). Ce phénomène n'influence, cependant, pas les performances de la méthode, car dans le cas d'un système anti-collision on s'intéresse avant tout aux distance et vitesse relatives. On remarque ainsi figure IV.12 de gauche que malgré cette connaissance des dimensions de l'objet erronée, les impacts du bas du véhicule sont parfaitement superposés au modèle de boîte englobante. Si on s'intéresse maintenant au cas de la Mini (figure IV.11 de droite). Sa longueur maximale estimée est de 3,5m et sa largeur est de 1,77m. Comme pour l'espace, sa longueur est sous estimée, cela est due au même constat. Sa largeur quant à elle est sur estimée, cela est dû au détecteur d'objet et à une mauvaise segmentation de la partition : le détecteur prend en compte quelques impacts appartenant au côté latéral de l'objet dans son estimation de la largeur, la rendant plus grande.

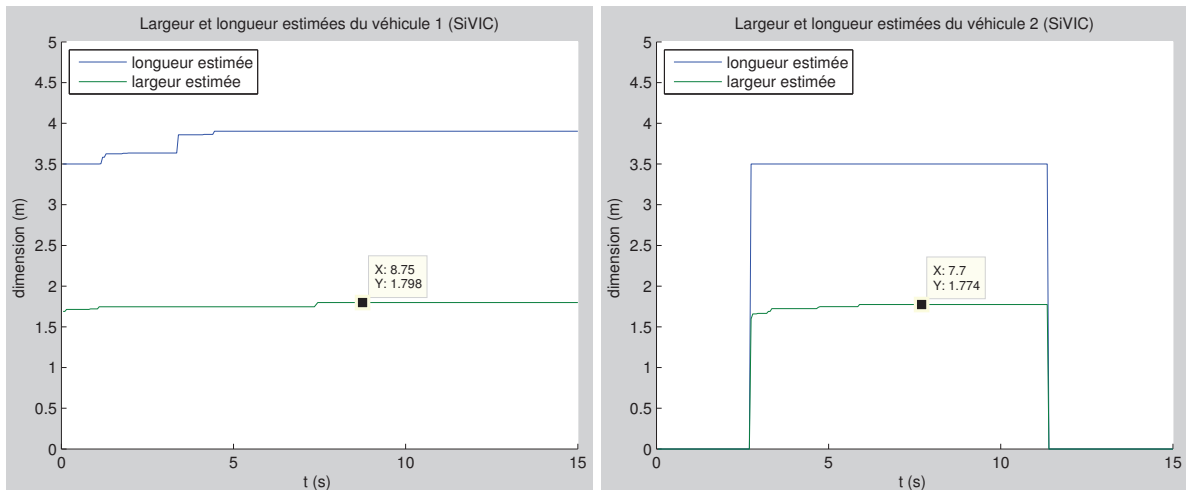


FIGURE IV.11. Longueur et largeur estimées des véhicules en fonction du temps à gauche l'Espace et à droite la Mini

La figure IV.12 présente un agrandissement des mesures observées sur l'Espace, aux temps $t = 1s$ à gauche et $t = 4s$ à droite. On remarque sur la figure de gauche que les mesures ne sont pas entièrement prises en compte dans la boîte englobante, par contre sur la figure de droite, toutes les mesures sont englobées. Si on se réfère à la figure IV.11 de gauche, on constate qu'entre $t = 1s$ et $t = 4s$ les dimensions du modèle de boîte englobante subissent quelques mises à jour.

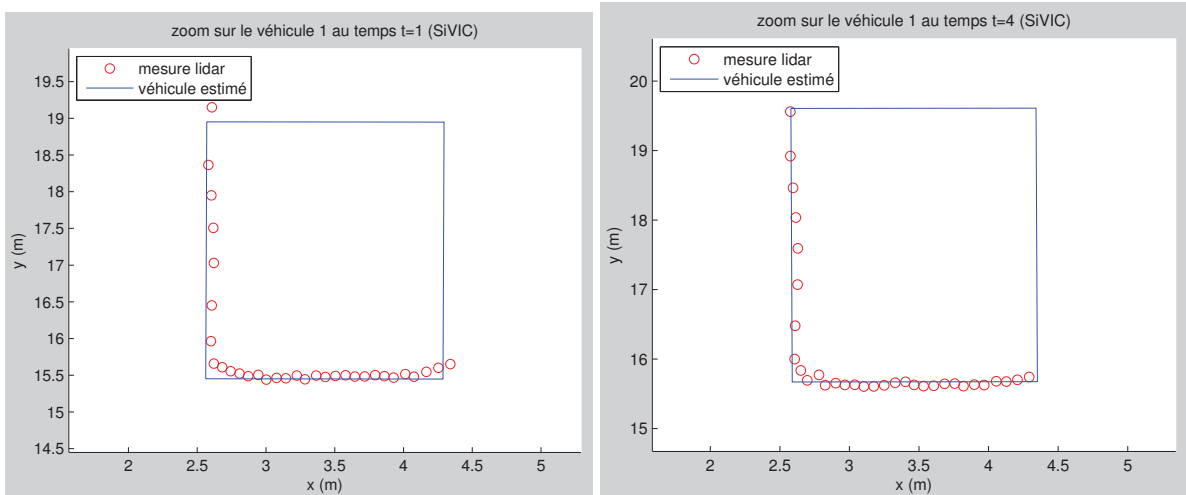


FIGURE IV.12. Représentation de l'Espace avant à $t = 1s$ (à gauche) et après à $t = 4s$ (à droite) mise à jour des dimensions

IV.2.6 Conclusion

Dans cette section, nous avons présenté l'extension de notre méthode de détection et de suivi au cas multi-objets hétérogènes. La présence d'objets multiples hétérogènes implique de devoir reconsidérer l'ensemble de la solution ITPF afin d'intégrer cette variabilité possible des paramètres dimensions. En effet, même si nous avons montré la robustesse de la méthode à une erreur sur les dimensions estimés du véhicule, la prise en compte de cette variabilité permet d'envisager une meilleure qualité du suivi et de gestion des situations pathogènes (occultation par exemple). Une des difficultés pour résoudre ce problème, provient de la structure même de notre méthode. En effet, l'absence de phase de détection et de suivi distincte ne permet pas la présence d'une phase de classification avant le suivi des objets. Notre méthode est cependant constituée d'un module de surveillance qui agit en parallèle des filtres, sans intervenir dans leur fonctionnement. Dans ce cas, il est parfois possible de considérer la présence d'un objet de grande taille alors que la scène réelle est constituée de deux objets de petites tailles mais très proches. Avec ces connaissances, le module de surveillance est alors incapable de dissocier les deux objets et considère qu'il est en présence d'un objet de grande taille, engendrant par la suite le changement de dimensions du modèle. L'apport d'une connaissance supplémentaire, provenant d'un GPS (*Global Positionning System*) ou d'autres véhicules pourrait permettre d'éviter ce genre de confusion. La méthode mise en place aurait dans ce cas été différente pour prendre en compte ces informations supplémentaires. Deux possibilités seraient envisageables, soit maintenir le changement des dimensions du modèle dans le filtre, soit faire cette mise à jour dans le module de surveillance à l'aide de la connaissance de la scène et de ces informations extérieures. Dans le premier cas, le module de surveillance transmettrait une information supplémentaire sur l'objet obtenue après traitement des données extérieures. Dans le second cas, le module de surveillance transmettrait les nouvelles dimensions et les nouveaux paramètres du centre de gravité. Pour maintenir le parallélisme de la méthode, un module supplémentaire contenant les paramètres modifiés pourrait être créé où les modules de surveillance et DAT viendraient récupérer les informations dont ils ont besoin. La figure IV.13 montre ce que deviendrait l'architecture fonctionnelle du système dans ce dernier cas.

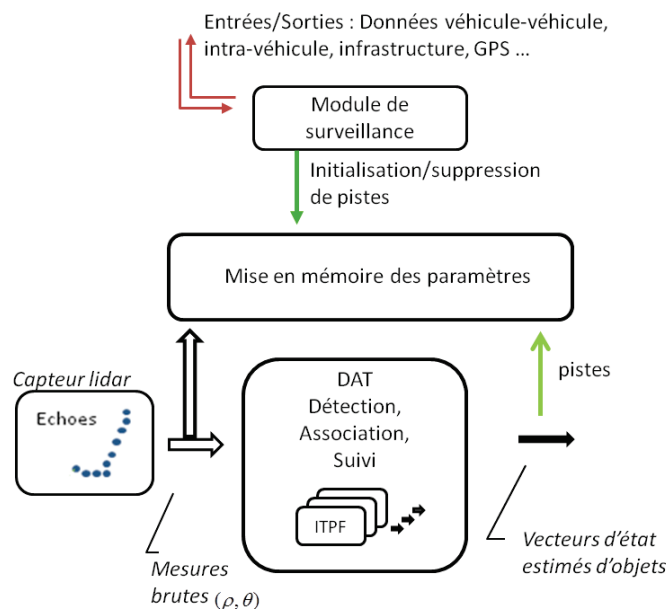


FIGURE IV.13. Architecture fonctionnelle de l'approche VS-ITPF modifiée

Nous avons donc réalisé un choix pour la mise à jour des dimensions des différents modèles du scénario, tout en gardant à l'esprit que l'apport d'informations supplémentaires pourrait modifier notre méthode. Notre choix est fortement dépendant des informations prises en compte et de la structure du filtre. En conclusion, nous avons montré la robustesse de la méthode vis-à-vis de véhicules de tailles différentes (voiture, camion). L'extension au cas de cibles de tailles plus petites (piéton, cycliste) est envisageable et reposerait sur une formulation de la fonction de pondération, étape cruciale de notre approche.

IV.3 Détection et suivi d'obstacles multiples homogènes avec un lidar multiplan (approche ML-ITPF)

IV.3.1 Problématique et mesures d'un capteur multiplan

Une des difficultés connues du lidar (liées notamment à sa directivité) est sa sensibilité forte au mouvement du véhicule porteur. En effet, les mouvements de roulis et de tangage peuvent donner lieu à des fausses alarmes (échos factices dus à la route par exemple) ou à des non-détection (absence d'échos par une visée à l'infini). L'utilisation d'un capteur télémétrique multiplan permet de remédier partiellement à cette difficulté en donnant une vision 3D partielle de la scène.

Ce type de capteur comporte plusieurs plans de mesure. Un même objet peut alors être détecté sur un nombre de ces plans de tir variable, au cours du temps ou en fonction de la distance relative au véhicule porteur. La mesure z_θ d'un lidar multiplan correspond donc, à un instant et pour un angle de tir donnés, à l'ensemble des valeurs de ρ_θ^m obtenues, sachant que m va de 1 au nombre de plans de tir (Nb_plan). On peut alors écrire :

$$z_\theta = (\{\rho_\theta^m; m = 1 : Nb_plan\}, \theta)$$

où $\rho_\theta^m < \rho_{inf}$ en cas de retour d'un écho au tir laser, du plan de tir m , pour l'angle θ (détection) ou encore $\rho_\theta^m = \rho_{inf}$ en cas de non retour d'un écho pour ce tir (non détection).

La figure IV.14 illustre les mesures obtenues, à la date $t = 5s$, pour deux véhicules situés à différentes distances du véhicule capteur. Les données SiVIC utilisées ont été obtenues avec quatre plans de tir dont les élévations respectives sont 0° , $0,4^\circ$, $0,8^\circ$ et $1,2^\circ$ et un capteur lidar positionné sur le véhicule à une hauteur de 80cm du sol. Les échos obtenus pour chaque plan de tir sont représentés en bleu (plan à 0° d'élévation), rouge ($0,4^\circ$), vert ($0,8^\circ$) et cyan ($1,2^\circ$). L'agrandissement autour de chacun des véhicules détectés masque les valeurs de ρ_θ^m qui pourraient être à l'infini. On remarque, figure de gauche, quatre plans de mesure visible et, figure de droite, seulement deux correspondants aux plans de tir de 0° et $0,4^\circ$ d'élévation.

Il existe plusieurs causes à ces variations du nombre de plans. Si un véhicule occultant est petit et le véhicule occulté grand, un tir élevé peut atteindre ce dernier sans toucher le véhicule occultant les autres plans. Cette situation est celle d'un masquage inter-véhicule. D'autre part, l'élévation relative du tir à la route est fonction de la distance au capteur. Enfin, selon les fluctuations de hauteur relative du capteur à la route (dus au tangage et au roulis), lorsqu'un tir est orienté trop bas ou trop haut, il peut respectivement créer des échos factices ou ne pas atteindre le véhicule. La figure IV.14 permet d'observer cette variabilité du nombre de plans de mesures. On constate notamment sur la figure IV.14 du bas à gauche, concernant la Newbeetle en bas à droite, que quatre plans de mesures sont observés. Par contre sur la figure IV.14 du bas à droite, concernant la Mini rouge voie centrale, seulement deux plans de mesures sont observés.

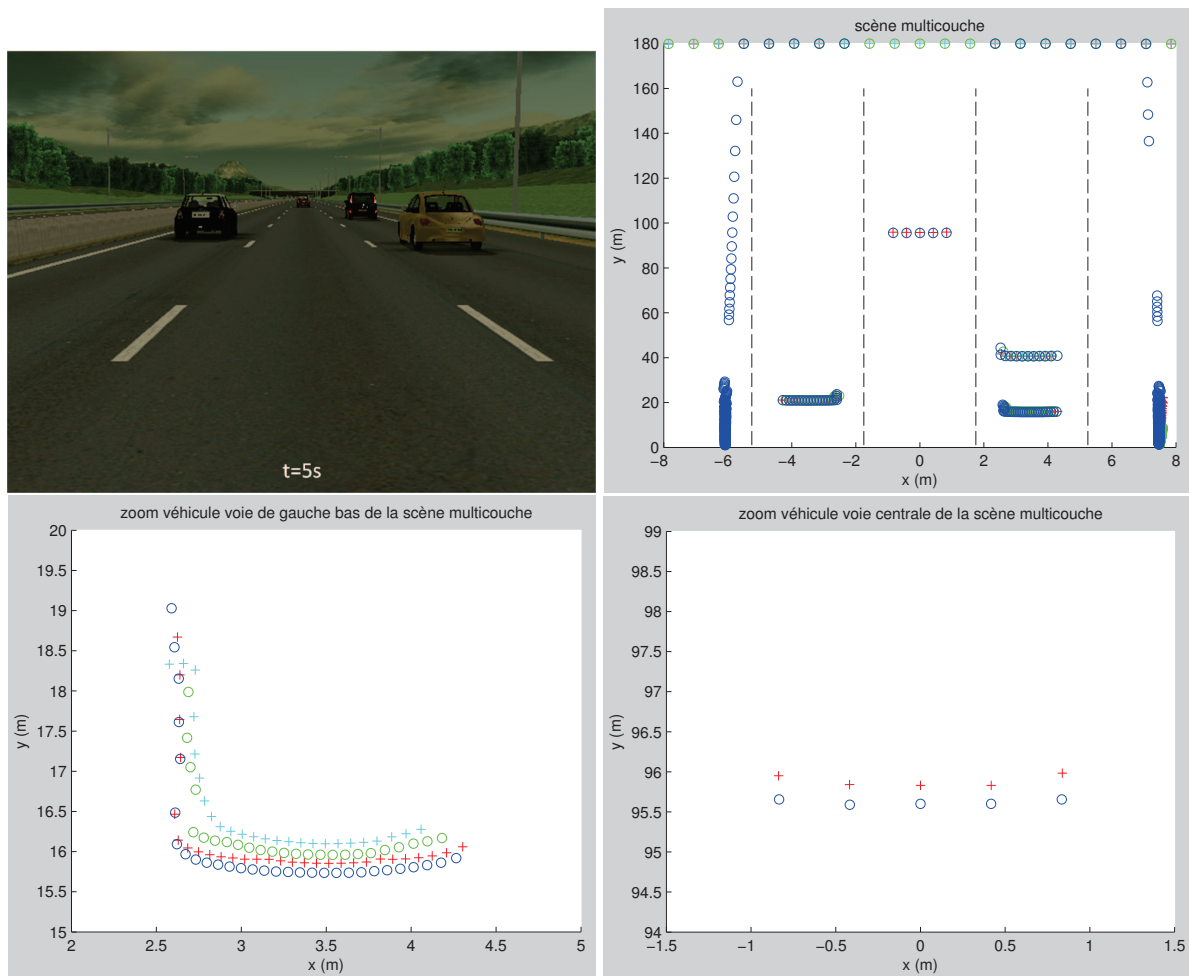


FIGURE IV.14. Exemple de deux véhicules vus par un capteur lidar multiplan au temps $t = 5s$ (en bleu tir à 0° d'élévation, en rouge tir à $0,4^\circ$ d'élévation, en vert tir à $0,8^\circ$ d'élévation et en cyan tir à $1,2^\circ$ d'élévation)

La difficulté de l'adaptation de notre méthode au cas du lidar multiplan que nous désignons par ML-ITPF (*multilayer* ITPF) réside dans la prise en compte des différents plans de tir pour lesquels on observera ou non, en fonction de la position du véhicule aux objets cibles, des impacts sur ces objets cibles. Nous garderons, pour cette adaptation, l'usage d'un modèle de boîte englobante, car si les échos sont maintenant en trois dimensions l'image détectée ne l'est pas. Le modèle du véhicule et l'estimation de son centre de gravité prendront cependant en compte l'ensemble des plans. Le modèle objet joue un rôle au niveau du détecteur d'objet du module de surveillance et au niveau de la pondération dans le filtre ITPF. C'est donc dans ces deux parties que des modifications sont à prévoir. Il est évident que la méthode globale reste similaire à celle présentée au chapitre III. Pour éviter les redondances, nous ne détaillons pas la méthode présentée dans le chapitre III, mais seulement les points soumis à adaptation, le lecteur pourra consulter le chapitre III pour les détails de la méthode. Par conséquent, nous présentons dans les paragraphes suivants les nouvelles notations utilisées pour prendre en compte les différents plans de tir (paragraphe IV.3.2), l'adaptation du détecteur d'objet du module de surveillance (paragraphe IV.3.3) et celle des filtres ML-ITPF au lidar multiplan (paragraphe IV.3.4). Enfin, nous montrons les résultats obtenus avec un lidar à quatre plans de tir (d'élévation 0° , $0,4^\circ$, $0,8^\circ$ et $1,2^\circ$) positionné sur le véhicule de sorte à être à $80cm$ du sol (paragraphe IV.3.5).

IV.3.2 Notation : extension au cas multiplan

Dans le cas de données multiplan, les notations présentées précédemment (section II.2) doivent être adaptées :

- Z_t l'ensemble des mesures d'un balayage t (sur l'ensemble des plans) ;
- $Z_t = \left\{ \bigcup_{m=1}^l z_{\theta}^m ; \theta^m \in \{\theta_{min}, \theta_{min} + \Delta\theta, \dots, \theta_{max}\}^m \right\}$ avec $m \in [1, Nb_plan]$;
- \mathcal{P}_t l'ensemble des partitions de Z_t ;
- $\mathcal{P}_t = \left\{ \mathcal{P}(n) ; n \in [0, N_{max}], N_{max} \leq \frac{\theta_{max} - \theta_{min}}{\Delta\theta} + 1, \mathcal{P}(0) = \emptyset \right\}$;
- N_{max} le nombre maximal d'objets dans l'image à t d'au moins un point
- $\mathcal{P}(n) = f(\mathcal{P}^m(n), m \in [1, Nb_plan])$
- $\mathcal{P}^m(n) = \left\{ z_{\theta}^m ; \theta \in \{\theta_a, \theta_a + \Delta\theta, \dots, \theta_{a+b}\}^m \right\}$, avec $m \in [1, Nb_plan]$, b le nombre de points de $\mathcal{P}(n)$ et $\mathcal{P}(1) \cap \dots \cap \mathcal{P}(n) = \emptyset$;
- $z_{\theta}^m = (\rho_{\theta}, \theta)^m$ où $m \in [1, Nb_plan]$ lorsque la mesure est exprimée en coordonnées polaires ;
- $z_{\theta}^m = (x_{\theta}, y_{\theta})^m$ où $m \in [1, Nb_plan]$ lorsqu'elle l'est en coordonnées cartésiennes ;
- $C_{\mathcal{P}^m(n)}$ l'ensemble des centres de gravité du plan m , $\forall m \in [1, Nb_plan]$;
- $C_{\mathcal{P}(n)}$ l'ensemble des centres de gravité calculés à partir des $C_{\mathcal{P}^m(n)}$, $\forall m \in [1, Nb_plan]$ pour un balayage capteur.
- m l'indice du plan de tir
- Nb_plan le nombre de plan

IV.3.3 Adaptation du détecteur d'objet du module de surveillance au cas multiplan

La mise en œuvre des filtres de suivi est liée à la création des pistes. Cela passe par l'utilisation d'un module de détection qui à partir des mesures brutes va extraire les objets à suivre. On sait que dans la méthode multiplan, le nombre d'objets sur chacun des plans peut être différent, comme l'illustre, la figure IV.15, pour les plans de tir de 0° et $0,8^\circ$. Deux possibilités s'offrent à nous, soit on effectue la détection à partir d'un unique plan, soit on effectue la détection à partir de l'ensemble des plans de tir.

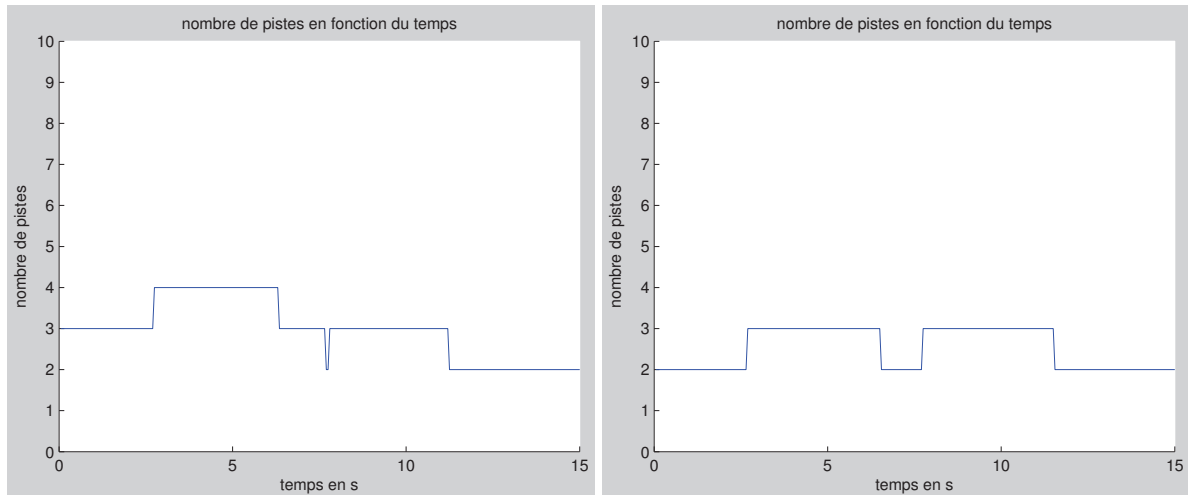


FIGURE IV.15. Nombre de véhicules détectés en fonction du temps pour les plans 1 (à gauche) et 3 (à droite)

L'utilisation d'un unique plan pour la détection implique d'avoir une connaissance *a priori* des différents plans. En effet, si le plan est haut, il est possible que certains véhicules ne soient pas touchés par le tir laser. Ainsi, tous les véhicules présents sur la scène, à cet instant, ne seraient pas détectés, ce qui n'est pas acceptable d'un point de vue applicatif. Utiliser un unique plan implique donc de connaître le plan qui admet l'ensemble des véhicules cibles à chaque instant. Le plan n'admettant aucune élévation est généralement le plus représentatif de la scène. Il existe une autre difficulté, si la distance entre les échos du plan le plus bas et ceux du plan le plus haut est importante, cela peut engendrer un biais entre le véhicule détecté sur un unique plan et l'estimation du véhicule suivi (à partir de l'ensemble des plans de tir) provoquant alors des erreurs d'association au niveau du module de surveillance. Au vu de ces difficultés, nous choisissons d'utiliser l'ensemble des plans pour la détection. Cela implique de gérer le nombre variable de véhicules par rapport au plan de tir et ce à chaque instant.

Le modèle objet de boîte englobante doit être au plus près des différents plans présents : la position du modèle sera la moyenne des positions de l'objet sur chaque plan. La méthode de détection mise en œuvre est inchangée par rapport à celle présentée algorithmes 1, 2, 3 et 4 du chapitre III. Elle est appliquée à chaque plan du balayage, pour en extraire l'ensemble des obstacles ($\mathcal{P}(n)$), mis en mémoire plan par plan ($\mathcal{P}^m(n)$). Les objets détectés sur chacun des plans sont mis en correspondance de sorte à calculer la position moyenne relative de chacun des obstacles perçus par le capteur. L'algorithme 10 résume cette mise en correspondance et la figure IV.16 illustre le résultat obtenu, l'objet détecté et son centre de gravité sont représentés, superposés aux mesures du lidar, par un rectangle et une étoile noire.

Algorithm 10 Algorithme de surveillance pour le cas multiplan

Require: Nb_plan le nombre de plans ; T la matrice contenant pour tout plan m l'ensemble de ses centres de gravité $C_{\mathcal{P}^m(n)}$ des partitions $\mathcal{P}^m(n)$; V le vecteur contenant la somme des centres de gravité détectés pour un même objet sur 1 à Nb_plan plan.

- 1: **for** $m = 1 : Nb_plan$ **do**
 - 2: Exécuter l'algorithme 1 de partitionnement de données (section III.4.1.2)
 - 3: Exécuter l'algorithme 2 de segmentation de données (section III.4.1.3) en sortie les $C_{\mathcal{P}^m(n)}$
 - 4: $T(m, :) \leftarrow C_{\mathcal{P}^m(n)}$
 - 5: **end for**
 - 6: Chercher la ligne g tel que le nombre d'éléments de $T(g, :)$ soit maximal
 - 7: $nb_element \leftarrow$ nombre d'élément de $T(g, :)$
 - 8: Initialiser V tel que $V(1 : nb_element) \leftarrow T(g, :)$
 - 9: **for** $b = 1 : nb_element$ **do**
 - 10: Chercher $\{T(m,a); T(g,b) \Leftrightarrow T(m,a)\} \forall a \in [1, nb_element]; m \neq g; m \in [1, Nb_plan]$
 - 11: $V(m) = V(m) + T(m,a)$
 - 12: $Nb_obj_somme(b) \leftarrow$ nombre des objets sommés à $T(g,b)$
 - 13: $C_{\mathcal{P}(n)} \leftarrow V(b)/Nb_obj_somme(b)$
 - 14: **end for**
 - 15: Exécuter l'algorithme 5 de surveillance (section III.4.2)
-

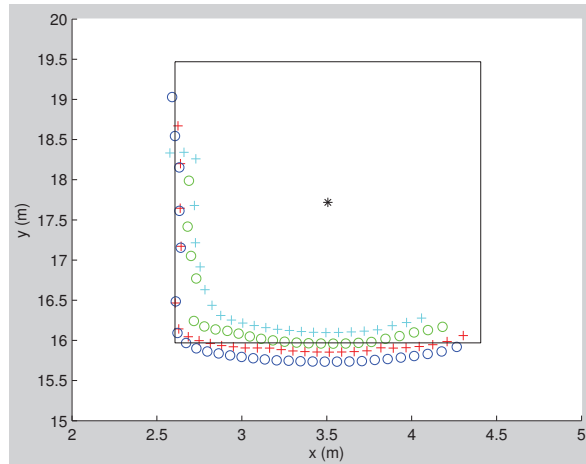


FIGURE IV.16. Illustration d'un véhicule vu par quatre plans capteur et la détection qui en découle

IV.3.4 L'approche ML-ITPF

Le module DAT est composé d'un ensemble de filtre ITPF permettant le suivi des objets dont il a la charge. À chaque filtre est associé un véhicule. Dans le filtre ITPF, c'est le calcul du poids qui nécessite une connaissance fiable du modèle. En effet, lors de cette étape les « véhicules particuliers » sont construits à partir de leur centre de gravité, de la rotation du véhicule (obtenue à partir de la vitesse estimée de celui-ci) et du modèle de véhicule. C'est donc juste avant cette étape que la prise en compte des différents plans doit avoir lieu. Dans un premier temps, nous présentons la modélisation du problème multiplan (section IV.3.4.1). Puis nous présentons les différentes étapes de la prise en compte des différents plans de mesures (section IV.3.4.3).

IV.3.4.1 Modélisation du problème

Dans le cas multiplan, la mesure est constituée des éléments du balayage capteur de chaque plan qui sont réalisés successivement. Ainsi, la quantité d'information à traiter est plus importante et peut être redondante. Dans le cas monoplan, les impacts liés au véhicule suivi ont été modélisés ainsi :

$$Z_t = \begin{pmatrix} \rho_{\theta_{min},t} \\ \vdots \\ \rho_{\theta_{max},t} \end{pmatrix} = h_t(X_t) + n_t$$

où les termes $\rho_{\theta_j,t}$ représentent les mesures de distance de tout le balayage, c'est-à-dire pour chaque angle θ_j de $\{\theta_{min}, \theta_{max}\}$ équidistant d'un pas angulaire $\Delta\theta$. n_t est un bruit blanc gaussien de dimension d et de moyenne nulle où $d = (\theta_{max} - \theta_{min})/\Delta\theta$.

On notera donc l'extension au cas multiplan de l'équation précédente de la manière suivante, pour le plan m :

$$Z_t^m = \begin{pmatrix} \rho_{\theta_{min},t}^m \\ \vdots \\ \rho_{\theta_{max},t}^m \end{pmatrix}$$

la modélisation des impacts de l'ensemble des plans Nb_plan du balayage capteur devient alors :

$$Z_t = \begin{pmatrix} Z_t^1 \\ \vdots \\ Z_t^{Nb_plan} \end{pmatrix} \quad (IV.1)$$

La modélisation dynamique reste quant à elle identique au cas monoplan. On définit le vecteur d'état $X_t = (x, V_x, a_x, y, V_y, a_y)_t^T$ où (x, y) est la position du centre de gravité de l'objet en fonction des différents plans, (V_x, V_y) est la vitesse et (a_x, a_y) est l'accélération. Il s'agit ici des positions, vitesses et accélérations relatives au véhicule porteur. Le problème d'estimation peut donc s'écrire de manière similaire au III.5 :

$$\begin{cases} X_t = f_t(X_{t-1}) + v_t \\ Z_t = h_t(X_t) + n_t \end{cases} \quad (IV.2)$$

avec f_t la fonction de dynamique, h_t la fonction de mesure, v_t le bruit de modèle et n_t le bruit de mesure. $v_t \sim \mathcal{N}(0, Q_t)$ et $n_t \sim \mathcal{N}(0, R_t)$ sont des bruits blancs gaussiens additifs de moyenne nulle. Q_t la matrice de covariance du bruit de modèle. $R_t = \sigma_\rho^2 I_n$ la matrice de covariance du bruit de mesure (I_n est la matrice identité de dimension n représentant le nombre de mesures de distance dans une analyse).

IV.3.4.2 Filtrage de Rao-Blackwell

Comme précédemment la solution proposée est basée sur l'exploitation d'un filtre particulière de Rao-Blackwell. Les particules $X^{(i)} \forall i \in \{1, \dots, N\}$ caractérisent N hypothèses dans l'espace d'état. L'équation dynamique (IV.2a) est alors séparée en deux parties (une partie non-linéaire et une partie linéaire) et devient IV.3 :

$$\begin{cases} X_{t+1}^P = f^P(X_t^P) + F_t^P X_t^K + G^P v_t^P \\ X_{t+1}^K = F_t^K X_t^K + G^K v_t^K \end{cases} \quad (IV.3)$$

Dans ce contexte, une autre équation de mesure associée à X^K doit être ajoutée au modèle d'état :

$$Z_t^K = X_{t+1}^P - f^P(X_t^P) = F_t^P X_t^K + G^P v_t^P \quad (IV.4)$$

Seule la partie non-linéaire (équations IV.1 et IV.3 a) est traitée par un filtre particulière, la partie linéaire (équations IV.3b et IV.4) est résolue par un filtre de Kalman (optimal dans ce cas). Le point central de notre approche est la modélisation d'état du problème étudié et plus particulièrement le modèle de mesure qui conduit au calcul des poids des particules.

Dans le cas multiplan, l'initialisation et l'évolution des particules restent identiques au cas monoplan. Nous ne réécrivons donc pas les équations décrites en III.6.2.1 et III.6.2.2 du chapitre III. Par contre, la pondération des particules doit prendre en compte les différents plans, nous en détaillons les étapes ci-dessous.

IV.3.4.3 Évolution des particules

Notre solution repose sur la construction de l'ensemble des échos $Z_t^{(i)}$, pour chaque particules (i) . C'est-à-dire, nous assurons une reconstruction des échos associés à la particule (i) sur l'ouverture angulaire $[\theta_{min}, \theta_{max}]$ d'un balayage. Or, plusieurs plans sont à prendre en compte ici.

Dans les systèmes lidar multiplans, le décalage spatial observé sur les différents plans est principalement lié à la forme du véhicule. L'élévation du balayage ne représente qu'une contribution non significative au décalage observé. On peut observer ce phénomène pour les véhicules étudiés sur la figure IV.14. Nous avons donc choisi d'en tenir compte dans notre approche particulière, et en particulier dans l'étape de reconstruction particulière du véhicule (que nous avons désigné par « véhicule particulière »).

Ainsi, pour chaque particule une reconstruction des échos d'un balayage capteur est réalisé. La zone de restriction ROI définie à partir des particules est calculée pour un plan. L'estimation de la ROI est donc réalisée de la même manière que pour le cas monoplan. Elle est présentée partie III.6.2.3.2 de la section III.6.2.3 du chapitre III, en considérant ici que la position de chaque particule est identique sur les différents plans (extension 3D du modèle de boîte englobante).

Par contre, la seconde restriction de la région d'intérêt, appelée ROI^+ , qui permet d'éviter que le calcul du poids soit réalisé aussi bien sur des mesures ρ_θ appartenant au véhicule que sur des mesures ρ_θ correspondant à l'infini, est définie plan par plan. Elle est créée à partir des mesures brutes de la zone ROI dans le plan, appartenant au véhicule suivi. Pour chaque plan m , les mesures brutes du capteur restant dans cette zone ROI^+ sont celles de la région ROI allégées de celle placées à l'infini (figure IV.17).

La nouvelle région de calcul du poids ROI^+ , d'un plan m ($\forall m \in [1, Nb_plan]$) est restreinte à :

$$\xi^{m,ROI^+} = \{\rho_{\theta,t}^m; \theta \in \{\theta_{min}^{ROI}, \theta_{min}^{ROI} + \Delta\theta, \dots, \theta_{max}^{ROI}\}; \rho_{\theta,t}^m < \rho_{inf}^m\} \quad (IV.5)$$

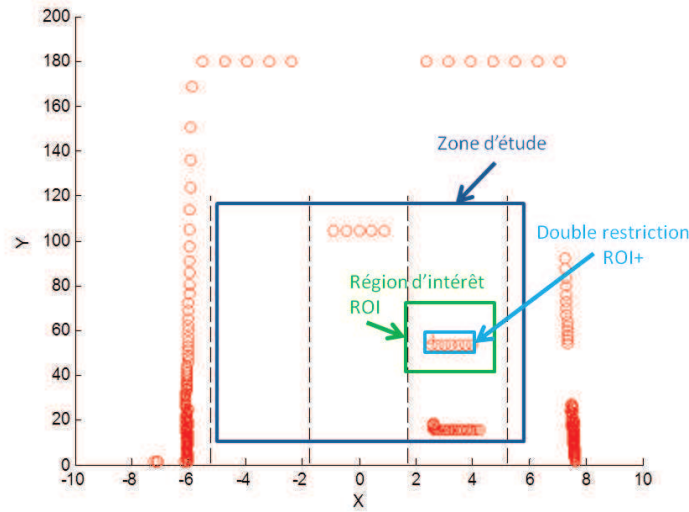


FIGURE IV.17. Double restriction, ROI^+ d'une région d'intérêt ROI de la zone d'étude considérée

Dans le problème multiplan, une des difficultés concerne la prise en compte de l'ensemble des plans de mesure. Toutefois, l'augmentation de la dimensionnalité de la mesure est directement prise en compte par le filtre particulaire, au travers du calcul du poids. La structure de pondération retenue présente une certaine adaptabilité qui permet de tenir compte de la variabilité du nombre de plan de mesure.

La structure globale du poids d'une particule s'écrit donc :

$$w_t^{(i)} = \prod_{m=1}^{Nb_plan} P(\rho_{\theta,t}^m | \rho_{\theta,t}^{m,(i)}) w_{t-1}^{(i)} \quad \text{avec } \theta \in \xi^{m,ROI^+} \quad (IV.6)$$

soit :

$$w_t^{(i)} = \frac{w_{t-1}^{(i)}}{\sqrt{\det(R)2\pi}} \exp \frac{-1}{2\sigma_p^2} \sum_{m=1}^{Nb_plan} \sum_{\theta \in \xi_{\sigma}^{m,ROI^+}} (\rho_{\theta,t}^m - \rho_{\theta,t}^{m,(i)})^2$$

et le poids devient :

$$w_t^{(i)} = \frac{\prod_{m=1}^{Nb_plan} P(\rho_{\theta,t}^m | \rho_{\theta,t}^{m,(i)})}{\sum_{j=1}^N \prod_{m=1}^{Nb_plan} P(\rho_{\theta,t}^m | \rho_{\theta,t}^{m,(j)})} w_{t-1}^{(i)} \quad (IV.7)$$

Il est ensuite normalisé selon l'équation II.11.

En résumé, la mise en place d'une double zone de suivi (ROI et ROI^+) permet de réaliser la procédure d'association entre les « véhicules particuliers » reconstruits et les mesures brutes appartenant à un véhicule. Elle permet d'éviter, tout d'abord, qu'un autre véhicule proche du véhicule étudié, attire les particules vers lui (par élimination des échos lui appartenant), et ensuite, de créer une attirance à l'infini.

Ensuite, le filtre présente une structure identique à la méthode présentée chapitre III (rééchantillonnage, prédiction du Kalman et calcul de l'estimé).

IV.3.5 Résultats expérimentaux

Il s'agit ici de montrer les performances de notre méthode de détection et de suivi d'objets mise en place, sur un exemple multiplan, sur les données du simulateur de trajectoires SiVIC (cf. annexe A.2).

IV.3.5.1 Scénario et caractéristiques du capteur

Nous choisissons de travailler à partir du scénario présenté au paragraphe III.7.2 et dont les caractéristiques sont rappelées dans le tableau IV.5. La scène est composée de quatre véhicules. Chaque véhicule apparaît à un instant t différent. Les trajectoires résultantes des véhicules sont présentées figure IV.18.

	Notation	Valeur
Capteur	(V_x, V_y) en ms^{-1}	(0, 20)
	Num_voie	2
	$(\theta_{min}, \theta_{max})$ en $^\circ\text{C}$	(-80, 80)
	(d_{min}, d_{max}) (m)	(10, 180)
	$(\Delta\theta, \Delta t)$ en $^\circ\text{C}$ et s	(0,25, 0,05)

Tableau IV.5. Paramètres du capteur positionné relativement à la route

La figure IV.19 montre des images CCD de la scène (présentée figure IV.18) à différents instants de simulation $t = 0\text{s}$, $t = 2,5\text{s}$, $t = 5\text{s}$ et $t = 10\text{s}$. On remarque ainsi l'apparition d'un véhicule à $t = 2,5\text{s}$, le déplacement d'un véhicule de la voie de droite à la voie centrale qui a pour conséquence l'occultation du véhicule voie centrale.

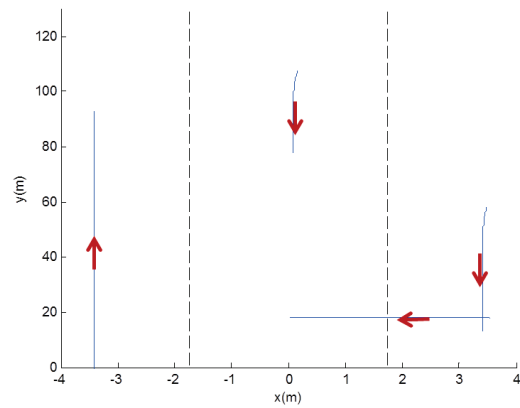


FIGURE IV.18. Trajectoires relatives au véhicule porteur de l'ensemble des véhicules de la scène

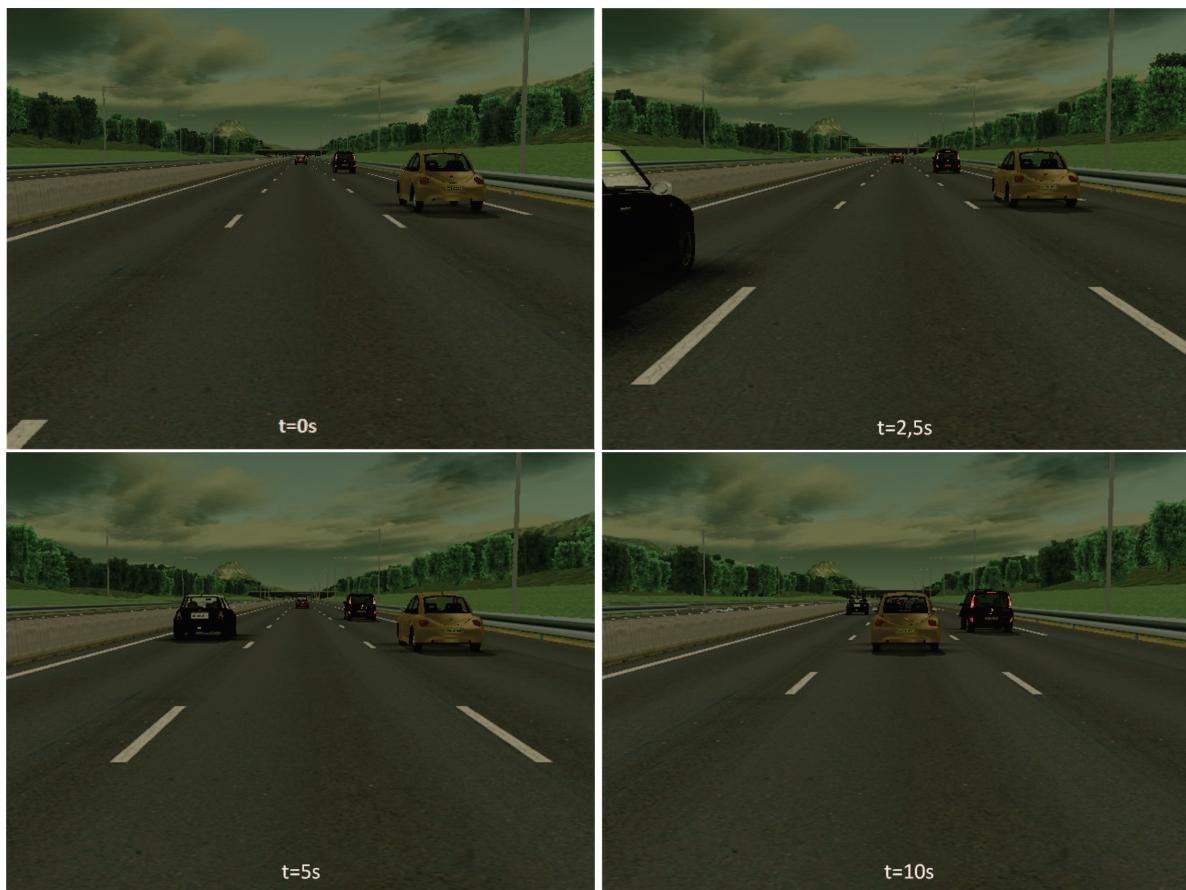


FIGURE IV.19. Quatre images de la scène au temps $t = 0s$, $t = 2,5s$, $t = 5s$ et $t = 10s$

IV.3.5.2 Résultats de l'approche ML-ITPF

Nous présentons une comparaison de notre méthode intégrée de détection et de suivi d'objets ML-ITPF à une méthode classique basée sur l'utilisation du centre de gravité (CDG) des objets à suivre à partir d'un filtre

de Rao-Blackwell, pour un exemple d'application multiplan. Les paramètres de simulation sont identiques pour chacun des deux filtres.

Nous vérifions ci-après le nombre de pistes créé par les algorithmes en fonction du nombre de véhicules présents à chaque instant (figure IV.20), ensuite nous vérifions la modélisation des véhicules estimés en fonction des mesures (figure IV.21 à IV.28), puis nous comparons les courbes d'estimations des vitesses et accélérations obtenues par les deux méthodes (figure IV.29 à IV.32).

Les schémas en figure IV.20 présentent le nombre de pistes créées et le nombre de véhicules présents pour les méthodes ML-ITPF à gauche et CDG à droite en fonction du temps.

Comme précédemment, les résultats obtenus avec les méthodes ML-ITPF et CDG sont similaires. Cela est dû au fait que nous utilisons comme algorithme de détection les mêmes algorithmes de Santos et de *Split and Merge* pour les deux méthodes ML-ITPF et CDG, de la même façon l'algorithme d'association de la méthode CDG est le même que l'algorithme de gestion de piste du module de surveillance de la méthode ML-ITPF.

Cependant, on remarque que le nombre de véhicules détectés ne correspond pas toujours au nombre de véhicules présents. Cela est dû à la structure de traitement détaillée précédemment. Lorsqu'une piste n'est plus détectée par le module de surveillance, le filtre de Rao-Blackwell peut continuer à fonctionner dès lors qu'il peut identifier des mesures compatibles avec la piste associée (c'est le cas notamment aux environs de l'instant $t = 6s$). De plus, le module de surveillance élimine une piste uniquement après trois itérations (équivalent à 0,15s) sans mesures pour mettre à jour la piste, donc sans détection du véhicule. On peut ainsi observer une continuité du suivi aux environs des instants $t = 6s$ et $t = 12s$. Ainsi le filtre de la méthode ML-ITPF continue à suivre le véhicule alors que le module de surveillance a cessé de le détecter (en raison soit d'un nombre d'impacts trop faible, soit d'une absence d'impacts). Dans le cas de la méthode CDG (figure de droite), le constat est le même. La méthode CDG se basant sur une agrégation de la mesure issue de la détection des objets, l'élimination de la piste se fait uniquement en fonction de la détection. Ainsi la piste est éliminée par le module de surveillance après trois itérations sans détection de l'objet en cours de suivi. C'est pourquoi sur la figure IV.20 à $t \simeq 6s$ la piste continue d'être suivi alors qu'elle n'est plus détectée et ce pendant un laps de temps plus important pour la méthode ML-ITPF.

On remarque également un pic à l'instant $t \simeq 8s$ qui est dû au cas d'une apparition et d'une disparition de deux véhicules (quasi) simultanément sur la scène. À cet instant, le véhicule se déplaçant de la voie 3 à la voie 2 occulte le véhicule situé en voie centrale alors que le véhicule de la voie 3 (précédemment occulté par ce déplacement) réapparaît. La configuration du filtre a été choisie de telle sorte que le suivi puisse continuer environ 0,15s après disparition totale des mesures, d'où le pic à $t \simeq 8s$ de 0,15s. Cette configuration a été définie afin de pouvoir prendre en compte la possibilité qu'un véhicule ne soit pas mesuré par le lidar à un instant donné (phénomène observé en pratique).

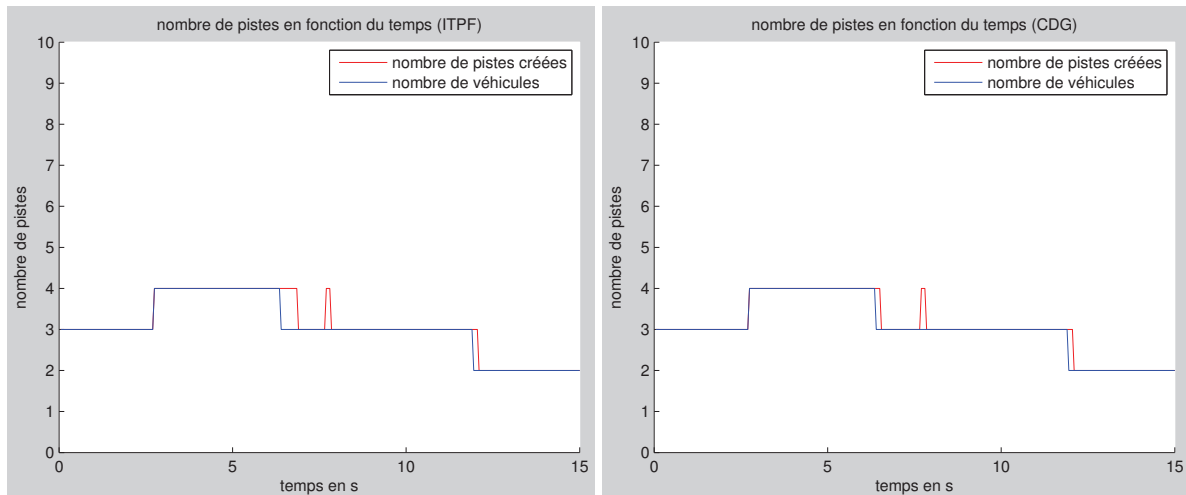


FIGURE IV.20. Comparaison du nombre de pistes créées avec le nombre de véhicule estimé par le détecteur présent au cours de la simulation, pour les méthodes ML-ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC

Les figures IV.21 et IV.25 suivantes montrent une image CCD de la scène et la perception de celle-ci par le capteur lidar respectivement au temps de 5 secondes puis à celui de 10 secondes. Les figures IV.22 à IV.24 sont des agrandissements de chacun des véhicules de la figure IV.21, à $t = 5s$. Celles de IV.26 à IV.28 sont des agrandissements de chacun des véhicules de la figure IV.25, à $t = 10s$. Pour chacune de ces figures, les mesures du lidar et la position estimée, représentée par un rectangle au dimension de l'objet suivi, ont été superposées, aussi bien pour les résultats de l'approche ML-ITPF que CDG.

Sur chacune de ces figures, les mesures lidar en rouges ont été superposées à l'estimation, faite par les filtres ML-ITPF (à gauche) ou par la méthode CDG (à droite), associés à chaque véhicule détecté. Cette estimation est représentée par le modèle objet (un rectangle) aux dimensions du véhicule suivi, en bleu sur les figures. La reconstruction du modèle objet correspondant à la position estimée se fait à partir de la position estimée du centre de gravité, de la connaissance *a priori* de la largeur et de la longueur des véhicules (i.e. les dimensions du modèle de véhicule utilisées dans le simulateur) et de celle de l'orientation des véhicules. L'orientation du véhicule est obtenue à l'aide de la vitesse propre de chaque véhicule.

L'approche proposée permet de prendre en compte de manière naturelle l'ensemble des plans de mesure et le fait qu'ils puissent être en nombre variable à chaque instant (par exemple : absence de véhicule dans un plan de mesure).

On remarque que pour les reconstructions de véhicule (en bleu), de la méthode ML-ITPF (à gauche sur les figures), le modèle traverse les différents plans de mesures (en rouge) en les moyennant. Sur certaines figures IV.22 et IV.28, il est légèrement décalé des mesures cependant, son orientation reste correcte. Par contre, pour la méthode CDG, on remarque que plusieurs véhicules reconstruits sont mal positionnés ou mal orientés. Le modèle s'écarte de façon importante de la mesure et les différents plans de mesures ne sont pas moyennés par l'estimé.

L'utilisation de la mesure brute dans la méthode ITPF permet donc de gagner en précision, aussi bien sur des données monoplan (observé dans les résultats du chapitre III) que multiplan. L'utilisation des multiples plans de balayage influe peu, voire pas du tout, sur les performances de notre méthode. On retrouve cette constatation sur les estimations de la vitesse et de l'accélération où la méthode ML-ITPF est stable comparée à la méthode CDG.

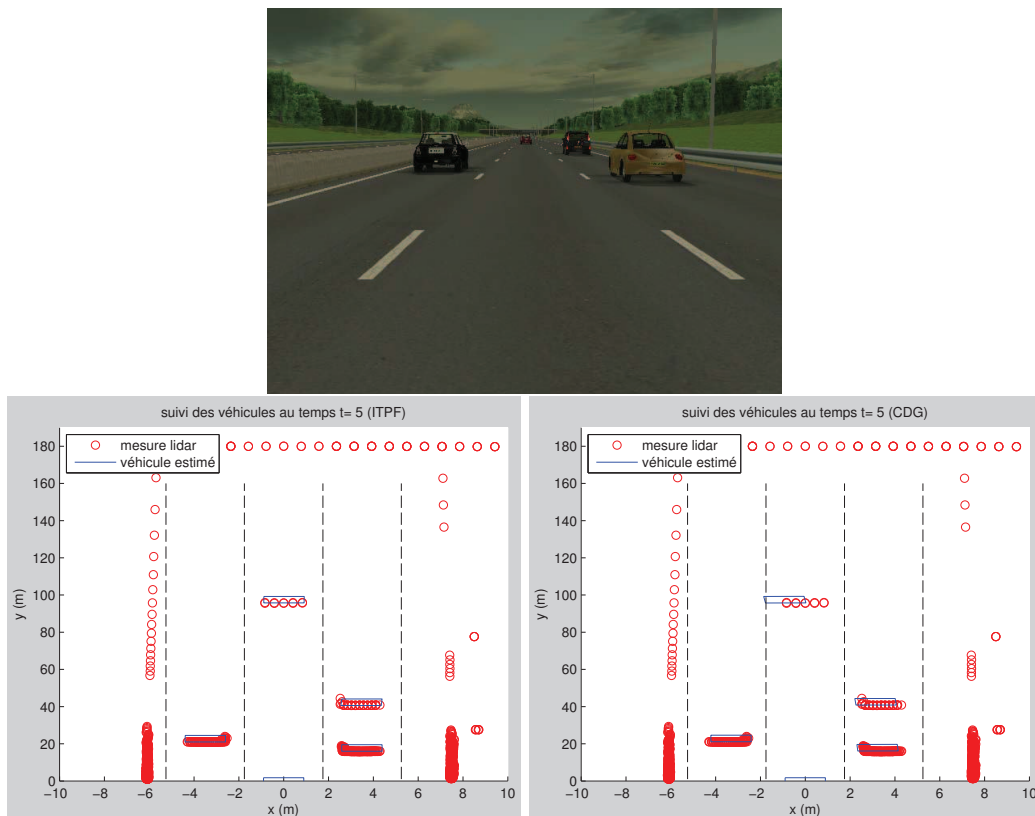


FIGURE IV.21. Image CCD de la scène, au temps $t = 5$ s et la même scène perçue par le laser à balayage (en rouge) superposée aux modèles estimés des véhicules (rectangle en bleu) : approches ML-ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC, pour un cas multiplan

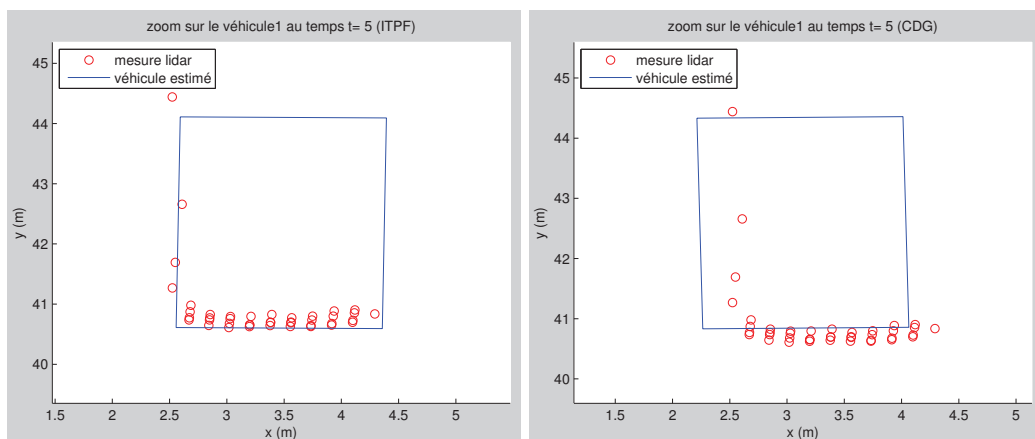


FIGURE IV.22. Agrandissement du véhicule de la voie de droite haut, au temps $t = 5$ s, de la figure IV.21 : approches ML-ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC, pour un cas multiplan

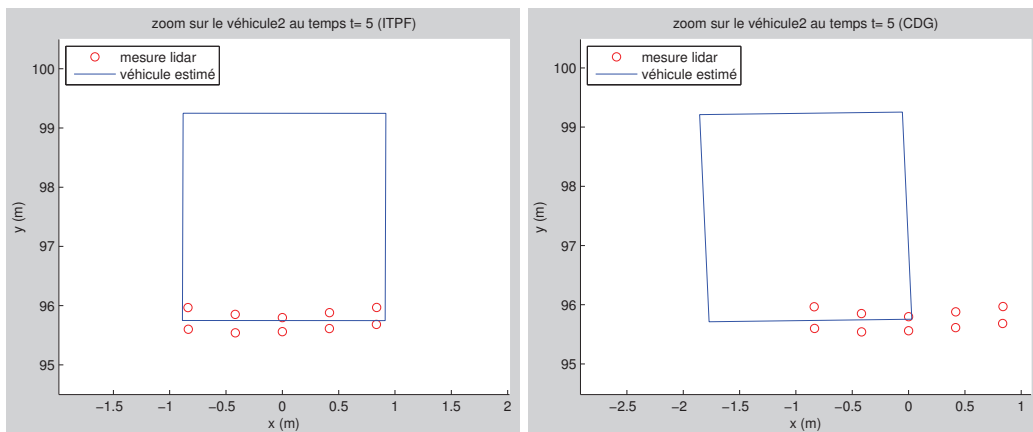


FIGURE IV.23. Agrandissement du véhicule de la voie centrale, au temps $t = 5$ s, de la figure IV.21 : approches ML-ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC, pour un cas multiplan

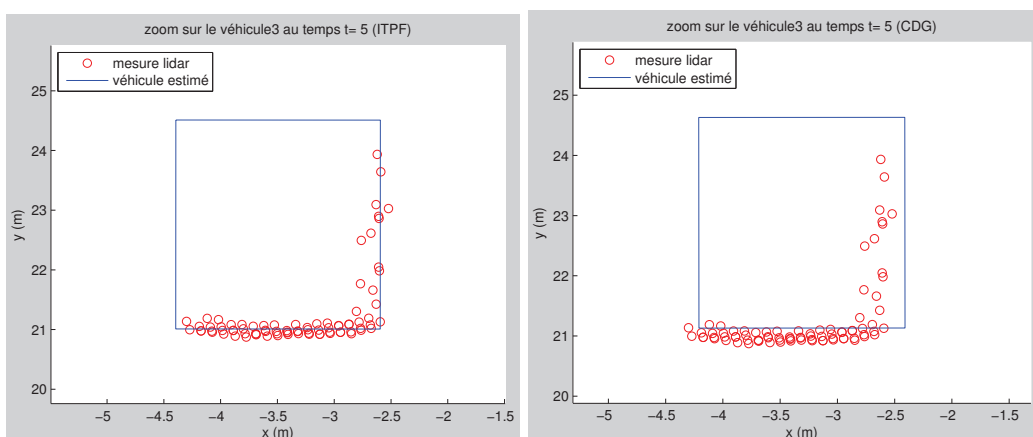


FIGURE IV.24. Agrandissement du véhicule de la voie de gauche, au temps $t = 5$ s, de la figure IV.21 : approches ML-ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC, pour un cas multiplan

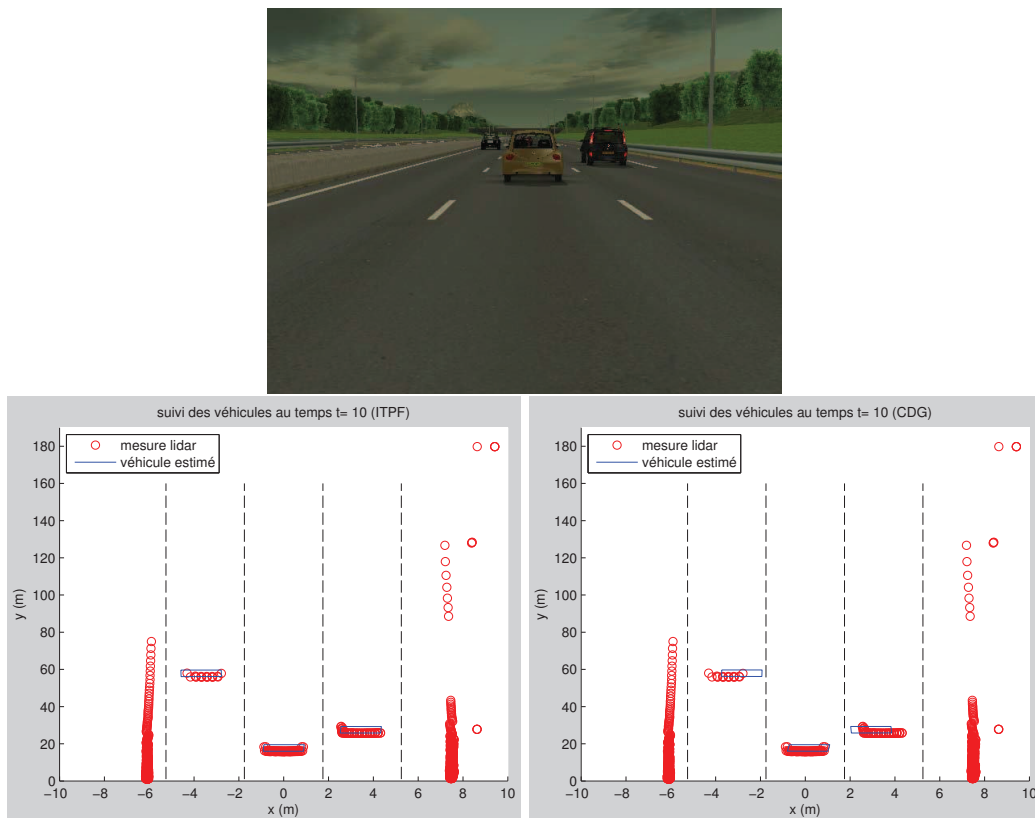


FIGURE IV.25. Image CCD de la scène, au temps $t = 10$ s et la même scène perçue par le laser à balayage (en rouge) superposée aux modèles estimés des véhicules (rectangle en bleu) : approches ML-ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC, pour un cas multiplan

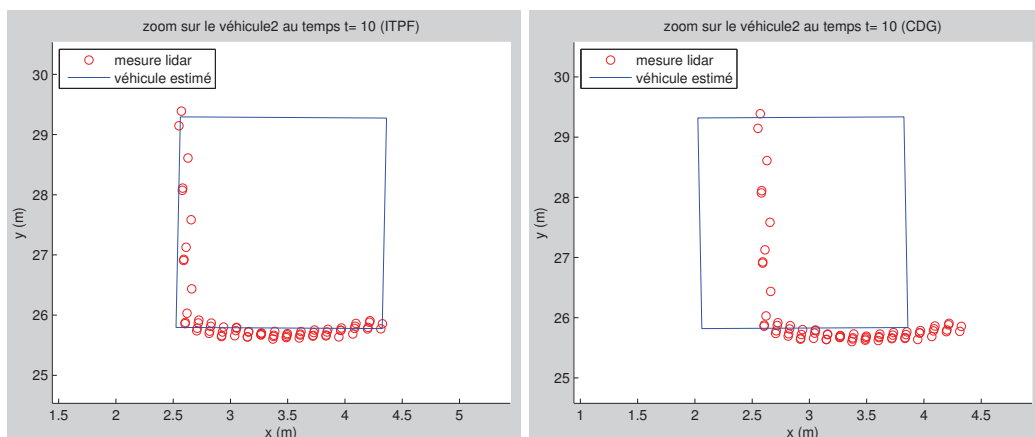


FIGURE IV.26. Agrandissement du véhicule de la voie de droite, au temps $t = 10$ s, de la figure IV.25 : approches ML-ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC, pour un cas multiplan

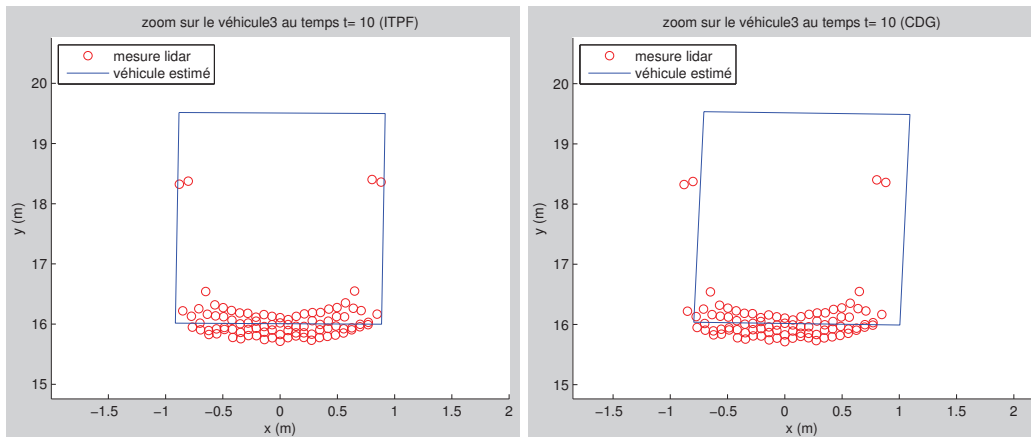


FIGURE IV.27. Agrandissement du véhicule de la voie centrale, au temps $t = 10$ s, de la figure IV.25 : approches ML-ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC, pour un cas multiplan

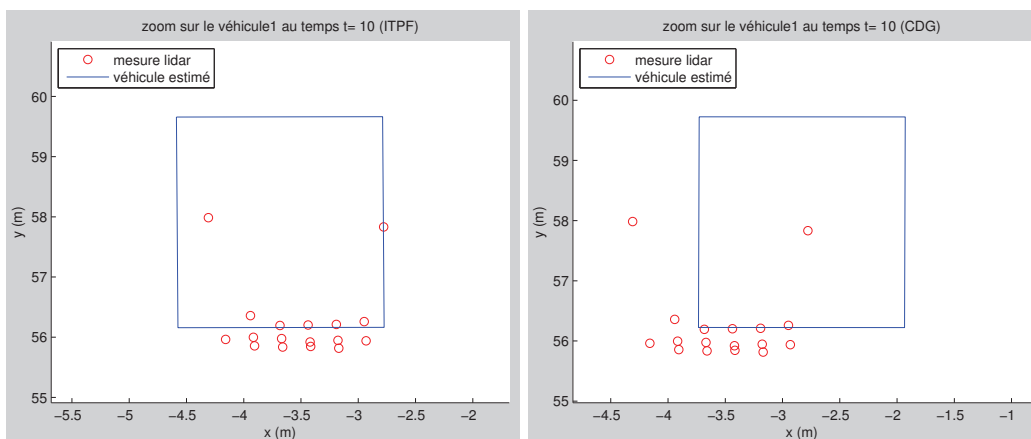


FIGURE IV.28. Agrandissement du véhicule de la voie de gauche, au temps $t = 10$ s, de la figure IV.25 : approches ML-ITPF à gauche et CDG à droite appliquées aux données du simulateur SiVIC, pour un cas multiplan

Les figures IV.29 à IV.32 présentent les vitesses et accélérations de deux véhicules tout au long de la simulation pour les méthodes ML-ITPF à gauche et CDG à droite.

Les figures IV.29 et IV.30 présentent celle d'un véhicule visible du début de la simulation au temps $t = 6$ s, puis occulté par un autre véhicule durant environ 2s avant de réapparaître et de rester présent jusqu'à la fin de la simulation. L'occultation, et donc l'interruption de suivi résultante, se traduit par un changement de piste et donc de couleur sur les figures.

Les figures IV.31 et IV.32 présentent les vitesses et accélérations d'un véhicule qui apparaît à $t = 3$ s puis disparaît à $t = 13$ s.

On remarque que sur les figures de gauche, de l'approche ML-ITPF, les estimations de la vitesse et de l'accélération semblent ne pas varier fortement et ceci sur la majorité de la simulation. Par contre, sur les figures de droite, de l'approche CDG, les estimations de la vitesse et de l'accélération en x varient

plus fortement. L'estimation en x fluctue plus fortement que pour ML-ITPF et semble rester instable. La convergence semble plus difficile. On montre là encore, mais cette fois avec des mesures en provenance d'un capteur multiplan, que le filtre CDG supporte mal les mouvements latéraux des véhicules cibles relativement au véhicule porteur, contrairement à la méthode ML-ITPF.

Les résultats obtenus sur des données d'un capteur multiplan sont donc similaires à ceux obtenus avec les données d'un capteur monoplan. En influant sur la rotation du véhicule, la vitesse et l'accélération contribuent à la précision du véhicule reconstruit par rapport aux mesures (cf. figures IV.21 à IV.28) où l'on a pu observer que les véhicules reconstruits issus de la méthode ML-ITPF sont mieux accolés aux mesures que ceux de la méthode CDG. Ils mettent une fois de plus en exergue la stabilité de la méthode ML-ITPF.

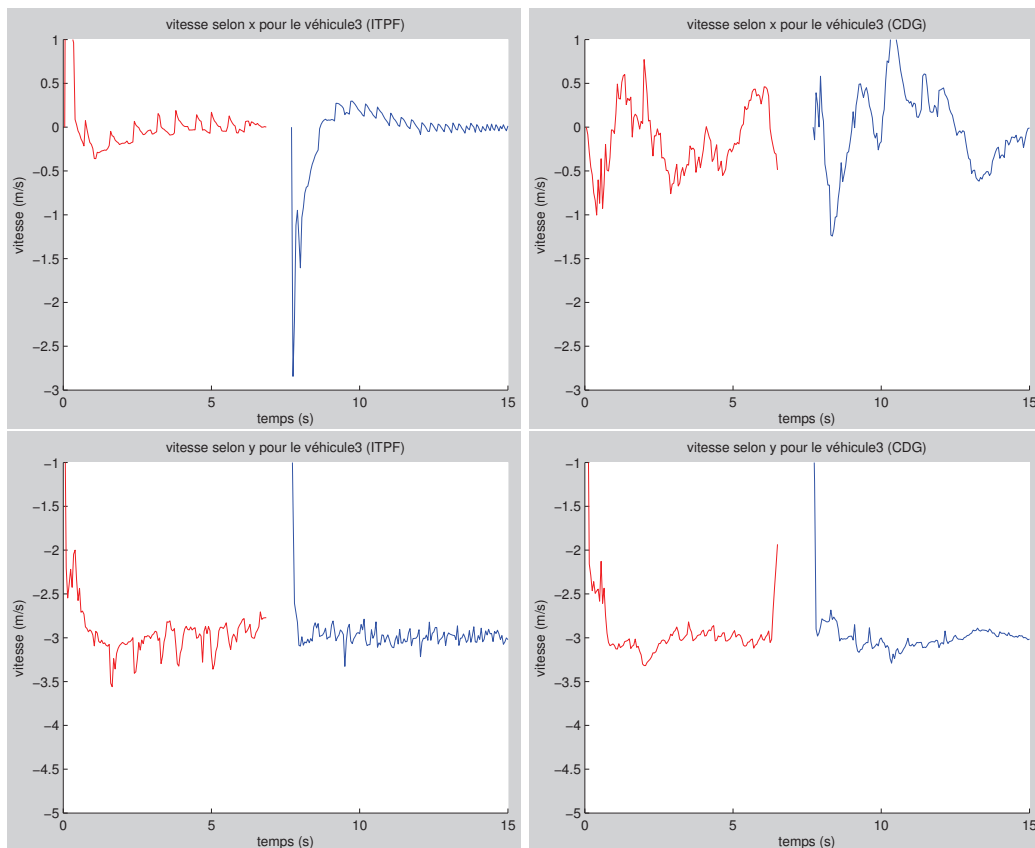


FIGURE IV.29. Courbes de vitesse selon x en haut et y en bas estimées du véhicule de la voie de droite en fonction du temps : approches ML-ITPF à gauche et CDG à droite appliquées aux données SiVIC, pour un cas multiplan

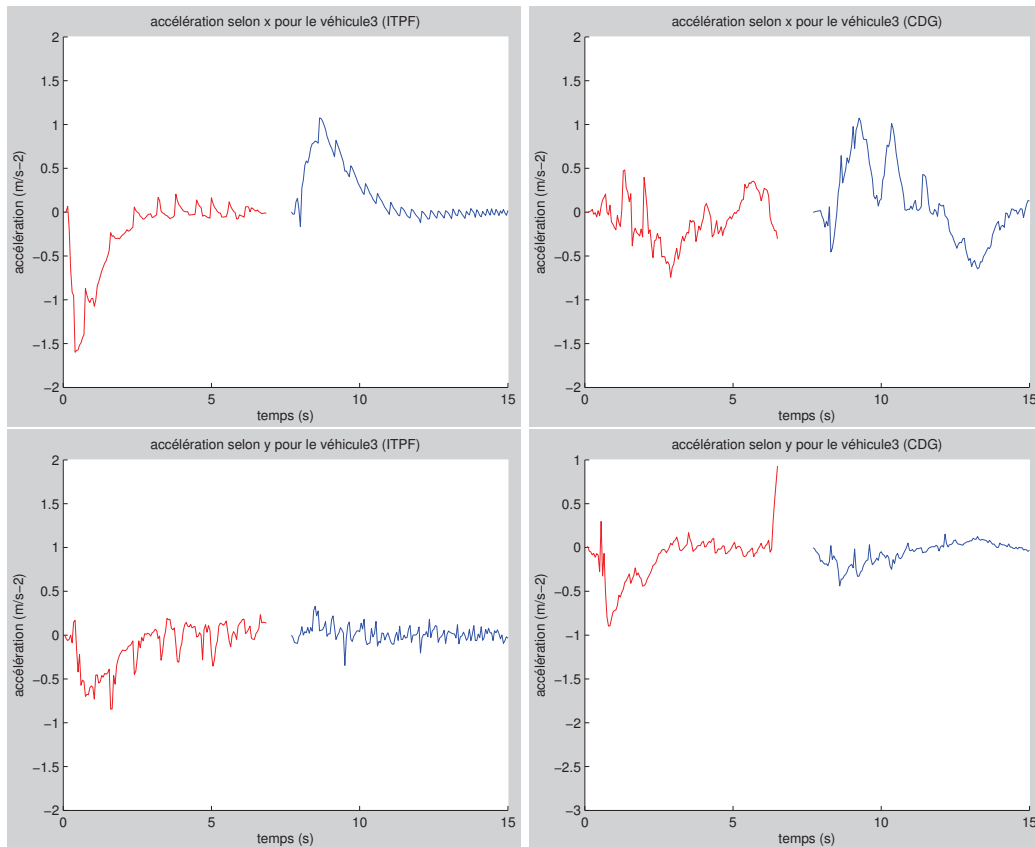


FIGURE IV.30. Courbes d'accélération selon x en haut et y en bas estimées du véhicule de la voie de droite en fonction du temps : approches ML-ITPF à gauche et CDG à droite appliquées aux données SiVIC, pour un cas multiplan

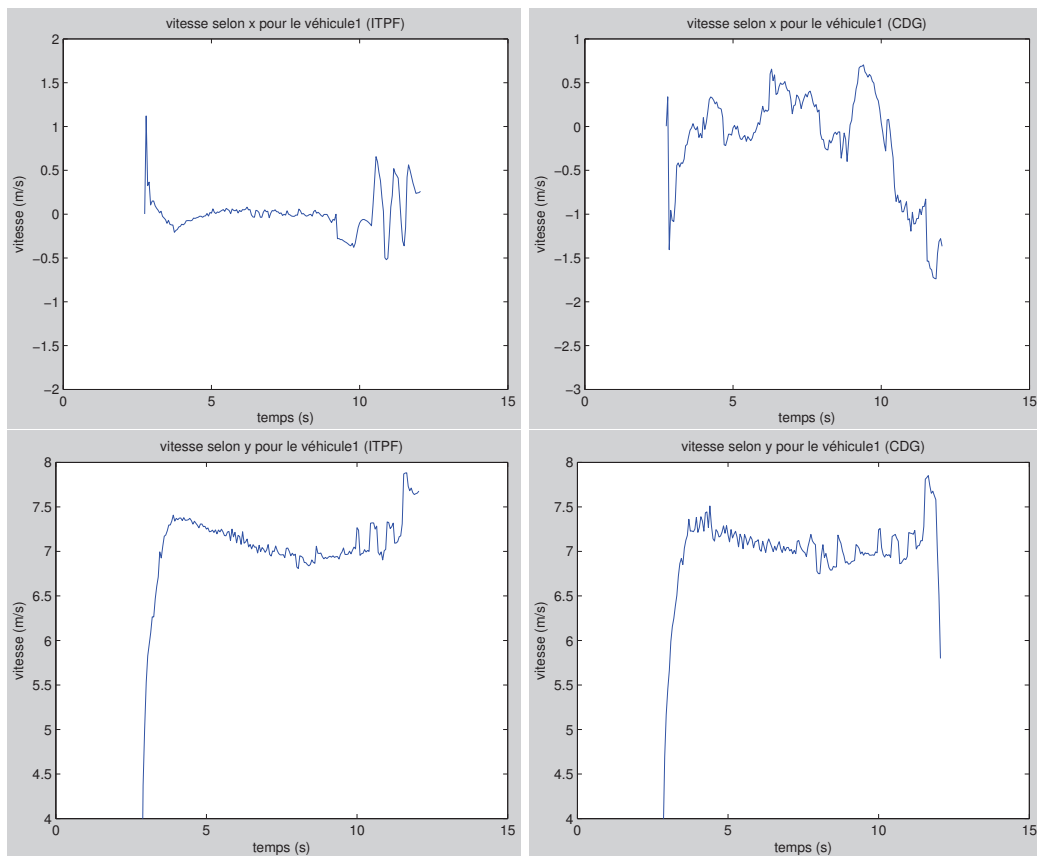


FIGURE IV.31. Courbes de vitesse selon x en haut et y en bas estimées du véhicule de la voie de gauche en fonction du temps : approches ML-ITPF à gauche et CDG à droite appliquées aux données SiVIC, pour un cas multiplan

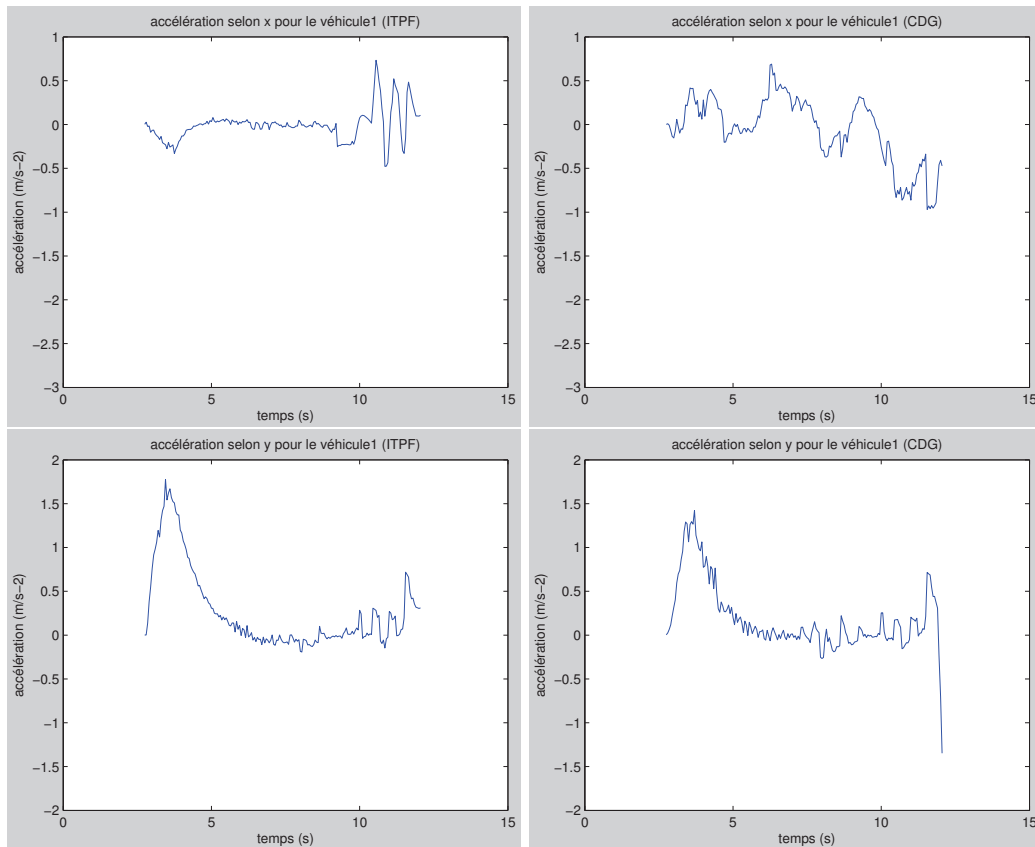


FIGURE IV.32. Courbes d'accélération selon x en haut et y en bas estimées du véhicule de la voie de gauche en fonction du temps : approches ML-ITPF à gauche et CDG à droite appliquées aux données SiVIC, pour un cas multiplan

IV.3.6 Conclusion

Dans cette section, nous avons présenté notre méthode de détection et de suivi sur un cas d'étude multiplan. Nous avons constaté que la présence de multiples plans de balayage implique de revoir la prise en compte des mesures brutes au niveau de la détection et dans le filtre. En effet, afin d'être le plus précis possible, nous cherchons à utiliser les mesures brutes issues du capteur dans le filtre. Pour cela, nous avons fait le choix de faire une moyenne des résultats de détection et de suivi des différents plans afin de prendre en compte toutes les données en provenance du capteur et ainsi estimer au mieux la position de l'objet. D'autres choix auraient pu être faits tels que la prise en compte d'un unique plan de mesures, le calcul d'une moyenne des impacts appartenant à chaque objet avant tout traitement de détection et de suivi afin de ne traiter qu'un unique plan, ou encore une détection et un suivi plan par plan, chaque plan ferait alors l'objet d'un suivi. Cette dernière méthode rendrait les traitements plus longs à mesure que le nombre de plans augmenterait.

Nous avons testé notre méthode de détection et de suivi multiplan sur des données issues du simulateur de trajectoire SiVIC, et l'avons comparé à une méthode classique CDG. On constate rapidement que le résultat obtenu est celui recherché, l'estimation réalisée par le filtre moyenne les différents plans. Les véhicules estimés semblent parfaitement collés aux différents plans de balayage. Comparativement à l'utilisation du centre de gravité les résultats de notre méthode intégrée sont plus performants et plus stables.

Conclusion

Dans cette thèse, nous avons cherché à résoudre les problèmes de détection et de suivi d'objets multiples hétérogènes sur route dans des données issues d'un télémètre laser à balayage. À cette fin, nous avons réalisé un état de l'art des méthodes et techniques de détection et de suivi d'objets (chapitres 1 et 2). Nous avons fait le choix du capteur et mis en place notre méthode que nous avons validée sur des exemples issus de différents simulateurs (chapitres 3 et 4).

La méthodologie de détection et de suivi proposée est construite sur l'utilisation de la mesure brute, en sortie du capteur tout au long des traitements. La méthode réalisée est constituée de deux modules, un module de surveillance et un module de détection et de suivi, appelé DAT (*Detection association and tracking*). Le premier a pour but de surveiller l'environnement autour du véhicule porteur du capteur. À cette fin, à partir de la mesure brute, il détecte les objets présents, chacun d'entre eux est ici représenté par une agrégation de la mesure. Puis, il transmet au second module le nombre d'objets et leur position (la position est symbolisée par le centre de gravité de l'objet suivi, extrait de l'agrégation de la mesure représentant celui-ci). Ces informations représentent « l'information *a priori* » dont le module DAT a besoin pour initier la détection et le suivi des objets détectés. Ainsi, pour chaque nouveau objet détecté, nous créons une piste qui contient les informations de l'objet qu'il suit, à chaque instant. Notre module DAT est ici constitué d'un ensemble de pistes dont un filtre de Rao-Blackwell réalise la détection et le suivi. La particularité de ce filtre, réside dans l'utilisation de plusieurs modélisations successives différentes de l'objet suivi. D'abord, le système est modélisé par les modèles d'états qui suivent la position, la vitesse et l'accélération des centres de gravité des objets. À partir du centre de gravité nous représentons le véhicule sous forme de boîte englobante. Ces dernières sont ensuite utilisées pour reconstruire les échos des véhicules à partir du modèle de capteur lidar. Dans notre filtre de Rao-Blackwell, le calcul du poids de chacune des particules est réalisé à partir de ces échos et de la mesure en sortie du capteur. Le centre de gravité (information estimée à partir de la mesure) n'est pas utilisé dans le calcul du poids ce qui permet de ne pas perdre d'informations concernant la scène.

La mise en œuvre de cette méthodologie a été proposée sur trois types de problèmes de détection et de suivi afin de souligner le caractère générique de l'approche présentée et couvrir un large champ d'étude :

1. détection et suivi d'objets mobiles multiples homogènes (dans ce cas, des véhicules de taille et de forme différentes), sur route, à partir de données lidar monoplan,
2. détection et suivi d'objets mobiles multiples hétérogènes (dans ce cas, des voitures et des camions), sur route, à partir de données lidar monoplan,
3. détection et suivi d'objets mobiles multiples homogènes, sur route, à partir de données lidar multiplan.

Les trois contributions ont permis de mettre en évidence plusieurs aspects de la méthode proposée.

Tout d'abord dans le point 1, nous nous focalisons sur une idéalisation du problème de suivi multicible où les objets, bien que de tailles différentes, sont modélisés de manière identique (boîte englobante de taille fixée). Les résultats obtenus sont intéressants et permettent de montrer la robustesse de notre méthode vis-à-vis d'une erreur de modèle. Les seules erreurs proviennent bien évidemment de la localisation du centre de gravité qui n'est pas nécessairement conforme à sa localisation réelle sur le véhicule considéré, mais les paramètres dynamiques (accélération, vitesse) ainsi que les distances relatives objet-capteur ne sont pas affectées.

Le point 2 nous a permis d'aller plus loin dans l'information délivrée par notre approche. Nous avons proposé une évolution de la méthode ITPF dénommée VS-ITPF (*Variable size ITPF*) qui permet outre une détection et suivi intégrée de décliner à chaque instant une estimation de la taille de l'objet suivi associé à chaque filtre particulière. En outre, par nature, l'approche VS-ITPF permet avant tout de décliner les informations précises de localisation et d'estimation des paramètres dynamiques, même si l'estimation des paramètres géométriques de l'objet (longueur, largeur) n'a pas convergé.

Pour finir, le point 3 aborde la généralisation de notre méthode aux capteurs lidar multiplans. En effet, pour des raisons pratiques (roulis, tangage du véhicule porteur, mouvements parasites, mauvaise réflectivité...) le lidar monoplan se trouve être de moins en moins utilisé. Nous avons donc envisagé l'extension de l'approche ITPF au cas de données laser multiplan approche ML-ITPF (*multilayer ITPF*). Deux difficultés essentielles doivent être résolues dans ce type de données. Tout d'abord, le capteur multiplan fait apparaître la nature tridimensionnelle des objets suivis ainsi que leur variabilité géométrique. En étendant la modélisation de l'objet au cas d'une boîte englobante 3D, nous avons pu profiter de la robustesse de l'approche ITPF à des erreurs de taille de modèle près. En effet, la forme perçue par un laser multiplan ne présente pas nécessairement les mêmes dimensions selon le plan considéré, aspect auquel l'approche ITPF permet de répondre de manière robuste. Le deuxième aspect du capteur multiplan vient de la variabilité (en dimension) des mesures associées à un objet. En effet, chaque objet n'est pas nécessairement observé sur tous les plans et les plans de visibilité peuvent en outre ne pas être les mêmes à chaque instant. L'approche ML-ITPF développée permet de tenir compte également de cet aspect sans avoir recours à une étape de détection préalable. Le filtre est en effet, en mesure de décider à chaque instant de la présence de l'objet sur chaque plan et de réaliser l'estimation de l'état (position, vitesse, accélération, orientation) en fonction de ce nombre variable de mesures.

Nous avons pu également détailler l'approche complète résultant de la combinaison des points 2 et 3 pour proposer une approche ML-VS-ITPF. Pour éviter toute présentation redondante, nous avons préféré ne pas la faire figurer dans ce manuscrit. Sa structure peut en effet aisément être déduite des approches ML-ITPF et VS-ITPF.

L'ensemble de ces approches (ITPF, VS-ITPF, ML-ITPF) a été validé sur des données synthétiques : les premières issues du simulateur de la thèse, données simplifiées d'un scénario routier, et les secondes issues du simulateur SiVIC, dont le comportement est proche de la réalité. Sur chaque type de données, nous avons comparé nos méthodes à une méthode dite « classique », appelée CDG, utilisant les centres de gravité des objets à suivre. Nous avons pu montrer que l'apport de la méthode se situe à plusieurs niveaux. Tout d'abord, l'amélioration des précisions en estimation est sensible, notamment en raison de l'absence d'étape de détection. Les mesures brutes directement exploitées garantissent un traitement optimal de l'information. Par ailleurs, l'approche ITPF se montre plus réactive à des changements brutaux d'accélération, ce qui permet de minimiser les risques de perte de suivi conduisant à une élimination de la piste active et à la création d'une nouvelle piste.

Par ailleurs, nous avons pu montrer que l'approche VS-ITPF permet de calculer avec précision la taille apparente du véhicule, taille remise à jour à chaque instant en fonction des points de mesures associés. Cette approche étend les premiers résultats obtenus avec ITPF qui avait montré une robustesse vis-à-vis d'une erreur de modélisation.

Pour finir, les résultats obtenus avec ML-ITPF permettent d'aller plus loin dans la précision en estimation tout en accroissant la robustesse quant à la détection et au suivi des objets d'intérêt.

Perspectives

Malgré des résultats encourageants, de nombreuses pistes restent à explorer. Nous allons présenter certaines des perspectives d'évolution qui s'inscrivent dans la suite du travail réalisé dans cette thèse.

Une première perspective de ces travaux concerne leur évaluation expérimentale. En effet, même si le simulateur SiVIC propose des données très proches de la réalité, il sera indispensable de valider ces travaux sur des données réelles d'un scénario routier.

Dans le cadre de cette thèse, nous cherchions à étudier un système coopératif de perception active distribué sur véhicule. Nous avons considéré le système composé du véhicule, du conducteur et des véhicules situés dans son voisinage. Chaque véhicule est équipé, de manière identique, d'un système de perception local composé en particulier d'un télémètre laser à balayage. Chaque système de perception local a une connaissance partielle de la scène qui pourrait être fusionnée avec celle délivrée par les autres systèmes locaux. Ceci permettrait d'accroître de façon significative le champ de vision de chaque véhicule de la scène. Comme on l'a vu précédemment, dans cette thèse, nous nous sommes focalisés sur la mise en place des systèmes de perception locaux (c'est-à-dire la phase de détection et suivi des objets). Il serait intéressant, maintenant, d'étudier le problème d'intégration des données provenant des autres systèmes de perception locaux. Deux aspects sont à étudier, la prise en compte de l'information reçue dans le système de perception local, et l'aspect transfert de données.

Dans notre architecture de traitement monocapteur, nous avons inclus un module contenant les informations provenant de capteur externe (information du système de perception local d'un ou plusieurs autres véhicules) et de capteur interne (informations concernant le véhicule porteur du capteur). Durant la thèse, les informations qui pouvaient être contenues dans ce module n'ont pas été prises en compte. On peut envisager que les informations provenant de ses capteurs puissent être utilisées dans le module de surveillance. Dans ce dernier, les informations concernant le véhicule pourraient servir à transmettre sa position aux véhicules voisins, et les informations provenant des autres véhicules pourraient être utilisées lors de la phase de détection. Dans le module DAT, ces informations serviraient, par exemple, au suivi des véhicules et notamment à la reconstruction des véhicules pour le calcul du poids. Cependant, certaines problématiques sont à considérer :

- Quelles sont les informations à transmettre véhicule à véhicule (information concernant la position de l'objet, ses dimensions, sa vitesse, son accélération, connaissance des traitements locaux de l'objet...) ?
- Quel repère global choisir au vu des différentes informations à traiter ?
- Quel est l'impact des contraintes liées à la communication sur l'architecture et la structure de traitement ?

L'aspect transfert de données est également un problème complexe. Il nécessite une étude des systèmes de communication sans fil existants pour les applications ITS (*Intelligent Transport System*). Pour optimiser les traitements, il faudrait également étudier les problèmes de synchronisation des différents capteurs. Ainsi, une première phase de modélisation des communications V2V entre systèmes de perception embarqués pourrait être réalisée mettant en avant les contraintes à prendre en compte.

Annexe A

Simulateur de données

Il est toujours difficile de concevoir les algorithmes d'une chaîne de traitement de détection et de suivi d'obstacles à partir des données de capteurs de perception bruitées et complexes. C'est en particulier le cas pour les systèmes de mesure montés sur des véhicules routiers. Ils sont sensibles aux évolutions environnementales et on ne maîtrise pas l'entièreté de la chaîne d'acquisition et des scénarios de mesures.

Il existe aujourd'hui des simulateurs qui reproduisent fidèlement des environnements routiers. On peut citer la plateforme de prototypage SiVIC (« Simulateur véhicule-infrastructure-capteur »), développée par l'unité de recherche LIVIC de l'Ifsttar, dont l'architecture logicielle a pour but « de reproduire, de la manière la plus fidèle possible, la réalité d'une situation, le comportement d'un véhicule et le fonctionnement des capteurs qui peuvent y être embarqués » Livic (2011).

Cependant, il peut aussi être intéressant de reproduire des données capteurs afin de mieux les comprendre et d'élaborer plus rapidement, de premiers algorithmes de traitement. C'est pour cette raison que nous avons réalisé un « générateur de données télémétriques ». L'objet de ce générateur n'est pas de reproduire avec exactitude le comportement d'un capteur télémétrique monté sur un type de véhicule déterminé et évoluant sur une route aux propriétés particulières, mais d'avoir un ensemble de données permettant de mettre en place les algorithmes de détection et de suivi afin de : bien appréhender le fonctionnement d'un télémètre (type de données, de détection...), appréhender ce qu'est une route simple et l'interaction route, télémètres, monocible ou multicible, faire la part des dysfonctionnements, lors de l'élaboration des algorithmes de détection, entre ceux qui pourraient provenir d'une détection incomplète de la route par le(s) capteur(s) distribués) ou bien d'une mauvaise conception de l'algorithme mais aussi d'évaluer les méthodes en termes de précision d'estimation. En résumé, il s'agit, dans un premier temps, d'avoir une bonne maîtrise de l'ensemble de la chaîne de développement avant d'envisager d'utiliser les données en provenance d'autres simulateurs de capteurs plus évolués ou de données réelles.

C'est pourquoi, nous présentons en section A.1, les différentes étapes de conception du générateur de données capteur que nous avons mis en place. Il est réalisé à partir d'hypothèses simplificatrices fortes que nous expliquons en paragraphe A.1.1. Dans un second temps (paragraphe A.2), nous présentons le simulateur de données SiVIC que nous avons utilisé pour tester notre algorithme de détection et de suivi.

A.1 Le capteur lidar à balayage

A.1.1 Principe du générateur de données télémétriques

Notre générateur de données doit générer un ensemble d'informations similaire à ce que détecterait le capteur physique équivalent s'il était monté sur un véhicule routier. Dans un premier temps, nous considérons des scénarios routiers simples pour la mise au point et la calibration de nos méthodes. Ainsi, le véhicule porteur du capteur évolue en ligne droite et ne pourra pas changer de voie en cours de simulation d'un scénario routier. La scène observée a été fortement simplifiée : elle est uniquement constituée de voiture de gabarits identiques (absence de rambardes, d'arbres, de piétons...). Ces véhicules cibles peuvent évoluer en ligne droite ou changer de voies.

Notre générateur doit reconstruire pour la scène routière un ensemble d'échos (ρ, θ) . Nous appelons « image » le résultat d'un balayage complet de la scène. L'intervalle de temps entre deux images correspond à la durée d'une rotation de l'axe du laser du capteur télémétrique. C'est la fréquence de balayage de la scène. L'intervalle angulaire entre deux échos est fixé par le capteur.

Le générateur de données télémétriques met en œuvre plusieurs étapes détaillées dans les paragraphes A.1.2 à A.1.4. Son fonctionnement est le suivant :

1. Définition d'un scénario dans un fichier script où l'on spécifie les données de simulation du scénario, notamment la position du véhicule capteur et celles des véhicules cibles et leurs trajectoires (paragraphe A.1.2) ;
2. sur la durée du scénario,
 - a. pour chaque véhicule visible au balayage courant,
 - i. la position du véhicule cible est calculée relativement à celle du véhicule capteur en tenant compte des paramètres de sa trajectoire (paragraphe A.1.3),
 - ii. une image est construite à partir d'une modélisation du véhicule cible selon un gabarit rectangulaire et en considérant ce véhicule comme le seul obstacle visible dans le référentiel du capteur (paragraphe A.1.4, en particulier A.1.4.1 à A.1.4.3) ;
 - b. une image de la scène est enfin construite en fusionnant les images obtenues pour chaque véhicule cible (paragraphe A.1.4.4).

A.1.2 Modélisation d'un scénario routier

Le générateur de données capteur repose sur une modélisation du capteur, de la route et des véhicules (véhicule porteur du capteur et véhicules cibles) :

- le capteur simulé est monté sur un véhicule en mouvement (figure A.1). Il détecte l'ensemble des véhicules qui se présentent dans son champ de vision. Ce dernier est caractérisé, d'une part, par une distance de détection minimale et maximale ($dist_{min}$, $dist_{max}$) et, d'autre part, par une ouverture angulaire définie par un angle minimal et maximal de vision (θ_{min} , θ_{max}). Enfin, il balaye la scène avec pas angulaire $\Delta\theta$ et une période d'échantillonnage Δt ;
- la route est caractérisée par le nombre de voies, un numéro de voie et une largeur (Nb_voie , Num_voie , $largeur_voie$) ;
- le véhicule capteur a une vitesse constante et est positionner sur une voie donnée (\vec{v} , Num_voie_{capt}) ;
- chaque véhicule cible est initialement caractérisé par une position relative au « véhicule capteur » et à une voie ($distance_vehicule_capt$, $Num_voie_{vehicule}$), ses vecteurs vitesse et accélération (\vec{v}, \vec{a}), puis un bruit sur l'accélération $\xi_{bruit_sur_a}$.

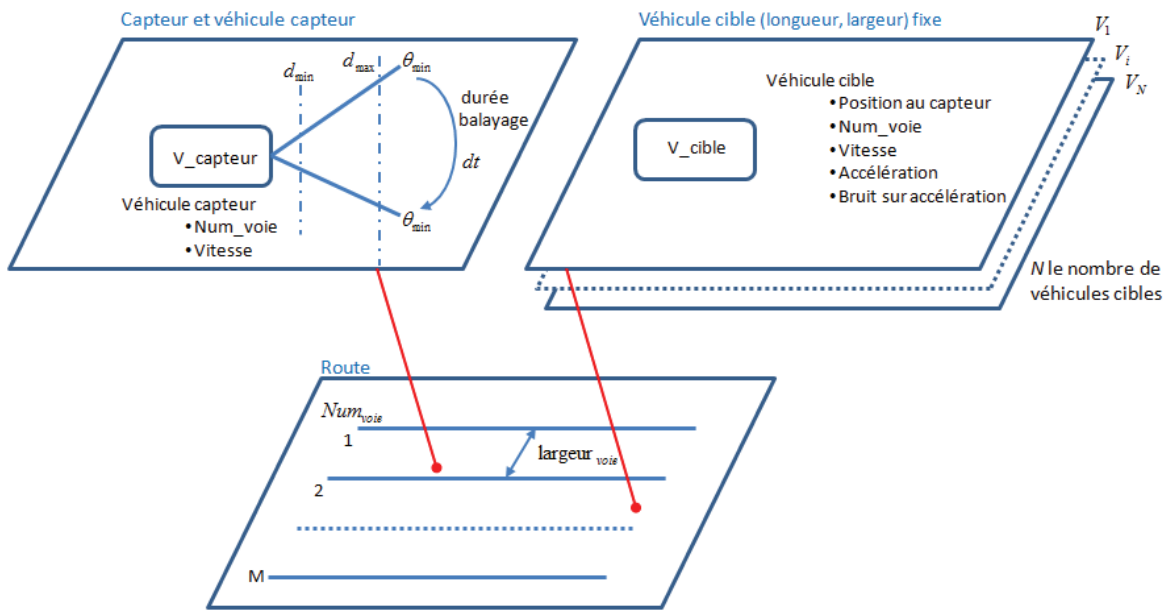


FIGURE A.1. Modélisation de la route, du capteur et des véhicules cibles

L'utilisateur définit donc un scénario dans un fichier script où il spécifie les données de simulation de chaque sous-ensemble du scénario (durée du sous-ensemble, nombre de véhicules cibles et l'ensemble des paramètres de modélisation du capteur de la route et des véhicules). Il définit ainsi un scénario global de circulation routière à partir duquel des données sont générées et sauvegardées dans un fichier texte.

Les tableaux A.1 et A.2 récapitulent les différents paramètres nécessaires.

	Notation	Valeur
Véhicule capteur	(v_x, v_y)	
	Num_voie	
Capteur	$(\theta_{min}, \theta_{max})$	
	$(dist_{min}, dist_{max})$	
	$\Delta\theta$	
	Δt	
Route	Nb_voie	
	Num_voie	
	$largeur_voie$	

Tableau A.1. Données paramétrables pour un scénario donné

	Notation	V1	Vi	Vn
Véhicules cibles (V1,...,Vn)	date d'apparition			
	<i>Num_voie</i>			
	(v_x, v_y)			
	(a_x, a_y)			
	<i>Bruit_sur_acc</i>			
	<i>distance_vehicule_capteur</i>			
	durée de simulation d'un sous-ensemble			

Tableau A.2. Données paramétrables pour chaque sous-ensemble du scénario

A.1.3 Modélisation de la trajectoire des véhicules cibles

Chaque portion de trajectoire d'un véhicule cible est modélisée par le système d'équations A.1 :

$$\forall t \in [t_{\text{apparition}}, t_{\text{apparition}} + \text{durée}] \begin{cases} X_t = \frac{1}{2}a_x t^2 + v_x t + X_{t-1} \\ Z_t = \frac{1}{2}a_y t^2 + v_y t + Z_{t-1} \end{cases} \quad (\text{A.1})$$

En modifiant portion par portion les vitesses et accélérations de ce système d'équations A.1, l'utilisateur peut spécifier, pour chaque véhicule cible, des scénarios de conduite comportant des sections parcourues à différentes allures et ainsi décrire des situations de suivi de véhicules, de changement de voie, de dépassement, de masquage...

A.1.4 Construction d'une image pour un balayage donné

À chaque balayage (soit tous les Δt), le capteur laser effectue un tir laser tous les $\Delta\theta$ entre les angles θ_{\min} et θ_{\max} . En retour, il reçoit soit un signal qu'il transforme en information de distance ρ (par calcul du temps de vol) lorsqu'un obstacle a été rencontré, soit une absence de signal en l'absence d'obstacle réflecteur qu'il transforme en une information « tir à l'infini » (ρ_{inf}). Ainsi, à chaque balayage, le capteur retournera une image finale composée d'un ensemble (ρ, θ) d'échos ou d'absence d'échos ($\rho_{\text{inf}}, \theta$), selon qu'il a ou non détecté un voire plusieurs obstacles.

Plusieurs étapes sont nécessaires à la construction de cette image pour un balayage donné. Dans un premier temps, chacun des véhicules va être considéré comme étant le seul obstacle présent. Une représentation de ces véhicules est alors réalisée selon un modèle de véhicule rectangulaire qui permet de calculer, puis de ne garder, pour chacun d'eux que les premiers échos (ρ, θ) détectés par le capteur. Dans un second temps, les échos des différents véhicules, obtenus pour ce balayage, sont comparés afin de ne garder que le sous-ensemble des échos les plus proches du capteur. C'est ce sous-ensemble qui définit l'image du scénario routier.

A.1.4.1 Représentation d'un véhicule dans le repère cartésien du capteur

Un véhicule cible est représenté longitudinalement et latéralement dans le champ de vision du capteur selon un gabarit rectangulaire de largeur L et de longueur l . Ce gabarit est spécifié par quatre segments ($S_{12}, S_{23}, S_{34}, S_{41}$) et identifié par ses points d'extrémités numérotés de $E1$ à $E4$ (figure A.2). La position relative au capteur de ce véhicule, pour un balayage particulier, est donnée par l'équation A.1 qui calcule les coordonnées de son centre de gravité dans le repère cartésien du capteur (x, y) . De même, son orientation relative par rapport à la route (et donc au repère cartésien du capteur) est connue via l'angle φ calculé à partir des vitesses (v_x, v_y) du véhicule à ce même instant.

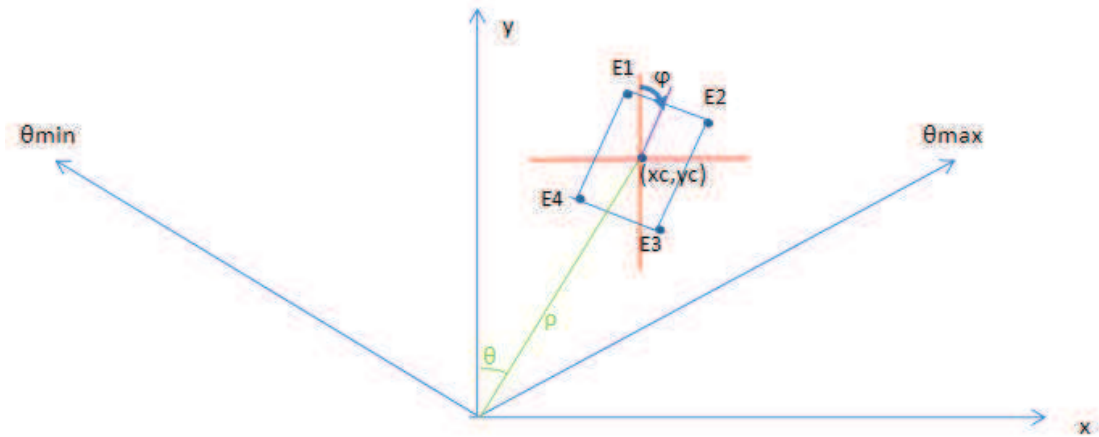


FIGURE A.2. Modèle de la voiture dans les repères cartésien et polaire du capteur

Les coordonnées des extrémités des segments sont ainsi calculées par la relation A.2.

$$Ei = \begin{pmatrix} \cos(\varphi) & \sin(\varphi) \\ -\sin(\varphi) & \cos(\varphi) \end{pmatrix} Mi + \begin{pmatrix} x_c \\ y_c \end{pmatrix} \quad (\text{A.2})$$

où $i = 1$ à 4 et

$$M1 = \begin{pmatrix} -L/2 \\ l/2 \end{pmatrix}, M2 = \begin{pmatrix} L/2 \\ l/2 \end{pmatrix}, M3 = \begin{pmatrix} L/2 \\ -l/2 \end{pmatrix} \text{ et } M4 = \begin{pmatrix} -L/2 \\ -l/2 \end{pmatrix}$$

Les équations des segments de droite du gabarit sont calculées aux moyens des équations A.3 des droites porteuses :

$$y = a_{ij}x + b_{ij} \text{ où } (i,j) \in \{(1,2); (2,3); (3,4); (4,1)\} \quad (\text{A.3})$$

Sachant que les coordonnées des barycentres $(x_{S_{ij}}, y_{S_{ij}})$ des quatre segments de droite S_{ij} sont connues et données par :

$$(x_{S_{ij}}, y_{S_{ij}}) = \begin{pmatrix} \cos(\varphi) & \sin(\varphi) \\ -\sin(\varphi) & \cos(\varphi) \end{pmatrix} Mij + \begin{pmatrix} x_c \\ y_c \end{pmatrix}$$

avec

$$M12 = \begin{pmatrix} 0 \\ l/2 \end{pmatrix}, M23 = \begin{pmatrix} L/2 \\ 0 \end{pmatrix}, M34 = \begin{pmatrix} 0 \\ -l/2 \end{pmatrix} \text{ et } M41 = \begin{pmatrix} -L/2 \\ 0 \end{pmatrix}$$

on en déduit a_{ij} le coefficient directeur de la droite S_{ij} , ainsi que b_{ij} son ordonnée à l'origine. Les formules présentées (équations A.4) ne sont pas simplifiées pour éviter les risques de division par zero.

$$\begin{cases} a_{ij} = \begin{cases} 1/\tan(\varphi + \pi/2) & S_{ij} \in \{S_{12}; S_{34}\} \\ \tan(3\pi/2 - \varphi) & S_{ij} \in \{S_{23}; S_{41}\} \end{cases} \\ b_{ij} = y_{S_{ij}} - a_{ij}x_{S_{ij}} \end{cases} \quad (\text{A.4})$$

Les droites porteuses des segments S_{ij} et les extrémités Ei de ces segments sont maintenant connues dans le repère cartésien du capteur. Il convient de transposer ces segments dans le référentiel en coordonnées polaires du capteur pour, ensuite, les utiliser afin de générer des échos.

A.1.4.2 Représentation d'un véhicule dans le repère polaire du capteur

Une fois les paramètres de droites connus, les équations de droite de ces segments S_{ij} sont transposées dans le repère polaire du capteur :

$$\left\{ \begin{array}{l} \rho = \frac{b_{ij}}{\cos \theta - a_{ij} \sin \theta} \\ \theta \in [\theta_{Ei}, \theta_{Ej}] \end{array} \right\} \text{ pour un segment } S_{ij} \quad (\text{A.5})$$

A.1.4.3 Calcul d'une image d'un véhicule dans le repère polaire du capteur

L'image du véhicule est réalisée en deux phases. La première phase consiste à calculer les échos que détecteraient le capteur sur son ouverture angulaire (θ_{min} et θ_{max}) avec un intervalle angulaire entre deux tirs de $\Delta\theta$, si chacun des segments du gabarit était considéré comme un obstacle unique. On obtient alors les valeurs de (ρ, θ) suivantes pour chacun des segments :

$$\left\{ \begin{array}{l} \rho = \rho_{inf}, \forall \theta \in [\theta_{min}, \theta_{Ei} \cup \theta_{Ej}, \theta_{max}] \\ \rho = \frac{b_{ij}}{\cos \theta - a_{ij} \sin \theta}, \forall \theta \in [\theta_{Ei}, \theta_{Ej}] \\ \text{avec } \theta \text{ variant de } \Delta\theta \end{array} \right\} \text{ échos d'un balayage en présence de } S_{ij} \quad (\text{A.6})$$

La seconde phase met ensemble les échos obtenus pour les quatre segments du gabarit. Elle sélectionne alors à chaque tir, entre θ_{min} et θ_{max} , l'écho le plus proche du capteur parmi les quatre valeurs obtenues pour définir au final l'image du véhicule.

A.1.4.4 Calcul de l'image capteur d'une scène multivéhicule pour un balayage donné

En présence de plusieurs véhicules, on génère leur image indépendamment selon le schéma détaillé précédemment. La scène finale est alors construite en considérant uniquement pour chaque tir entre θ_{min} et θ_{max} , l'écho le plus proche du capteur.

A.1.5 Exemple de données obtenues à partir du simulateur

La figure A.3 illustre les données construites par le générateur aux instants de balayages 10, 50, 90, 110 et 150 dans le cas d'un scénario de conduite sur une route à trois voies sur laquelle circulent deux véhicules V1 et V2, en plus du véhicule capteur (non représenté). Les paramètres de simulation choisis pour le véhicule capteur, le capteur et la route sont résumés dans le tableau A.3, les paramètres du capteur ont été choisis conformément à ceux du capteur LD d'IBEO.

	Notation	Valeur
Véhicule capteur	(v_x, v_y) en $m.s^{-1}$	(0, 15)
	<i>Num_voie</i>	2
Capteur	$(\theta_{min}, \theta_{max})$ en $^{\circ}C$	(-80, 80)
	$(dist_{min}, dist_{max})$ en m	(5, 250)
	$\Delta\theta$ en $^{\circ}C$	0,25
	Δt en s	0,05
Route	<i>Nb_voie</i>	3
	<i>Num_voie</i>	{1,2,3}
	<i>largeur_voie</i> en m	3,5

Tableau A.3. Paramètres de simulation

Les paramètres de simulation concernant les véhicules cibles sont résumés dans les tableaux A.4 et A.5. Ces véhicules s'éloignent du véhicule capteur. Le véhicule V1 est initialement sur la voie de gauche, puis rejoint la voie centrale où se trouve le véhicule porteur au moyen de trois changements de son accélération en x , son accélération en y restant constante tout au long de la simulation. Le véhicule V2 reste sur la voie de droite et s'éloigne du véhicule capteur avec une accélération constante en y .

	Notation	Phase 1	Phase 2	Phase 3	Phase 4
Sous-ensemble V1	date d'apparition en s	$t = 0$	$t = 2$	$t = 4,5$	$t = 7$
	Num_voie	1			
	(v_x, v_y) en $m.s^{-1}$	(0, 6)			
	(a_x, a_y) en $m.s^{-2}$	(0, 0,2)	(0,5, 0,2)	(-0,5, 0,2)	(0, 0,2)
	$Bruit_sur_acc$ en $m.s^{-2}$	0,03	0,03	0,03	0,03
	$distance_vehicule_capteur$ en m	20			
	durée de simulation d'un sous-ensemble en s	2	2,5	2,5	3

Tableau A.4. Paramètres du véhicule V1

	Notation	V2
Sous-ensemble V2	date d'apparition en s	$t = 0$
	Num_voie	3
	(v_x, v_y) en $m.s^{-1}$	(0, 7)
	(a_x, a_y) en $m.s^{-2}$	(0, 0,8)
	$Bruit_sur_acc$ en $m.s^{-2}$	0,03
	$distance_vehicule_capteur$ en m	25
	durée de simulation d'un sous-ensemble en s	10

Tableau A.5. Paramètres du véhicule V2

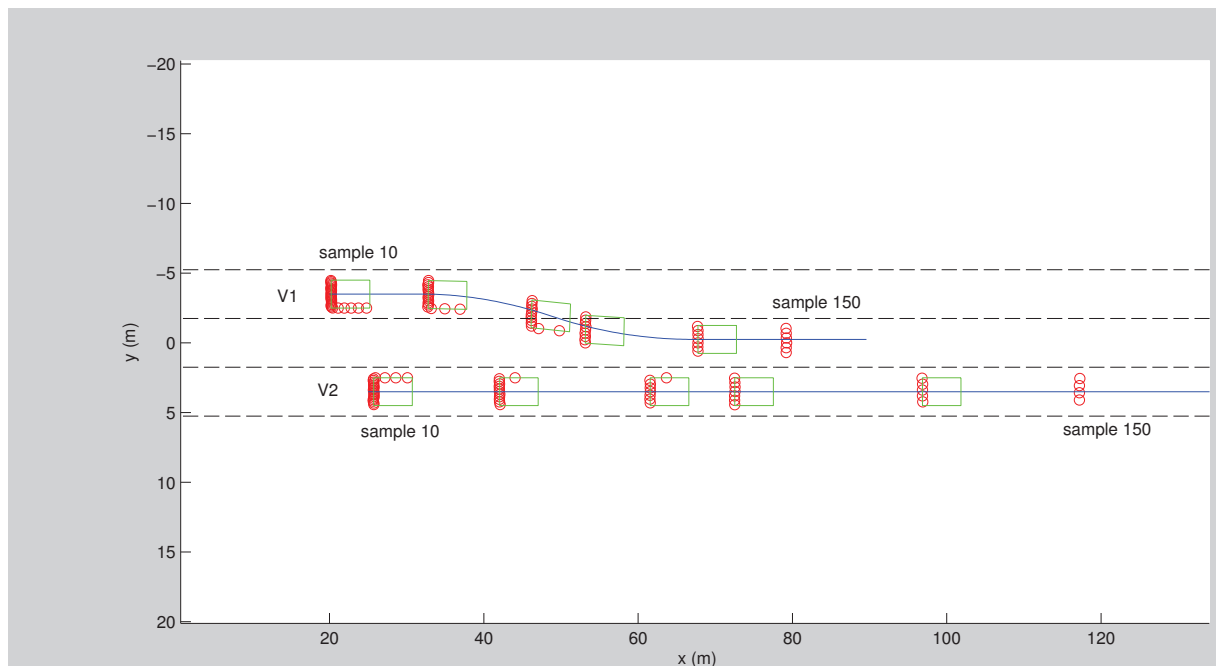


FIGURE A.3. Données générées pour le scénario considéré (deux véhicules V1, V2 dont l'un change de voie)

A.2 Le simulateur SiVIC

A.2.1 Présentation

Le simulateur SiVIC est une plateforme de prototypage de capteurs virtuels, dont le but est de reproduire, fidèlement un scénario routier. Il offre un environnement virtuel incluant des véhicules, des obstacles, des infrastructures et des capteurs que l'on intègre de manière modulaire. Cette plate-forme propose de générer à terme des données homogènes avec celles qui peuvent être enregistrées par le système d'acquisition de données des véhicules d'essai, à savoir un flux continu de données numériques datées et synchronisées, issues des caméras, télémètres, GPS, centrale inertielle, etc (Gruyer *et al.*, 2009; Livic, 2011). D'après (Gruyer *et al.*, 2009), ce logiciel procure des graphismes réalistes ainsi qu'une modélisation précise du comportement dynamique d'un véhicule qui tient compte de l'angle de braquage, de l'accélération, du freinage, des suspensions...

Ces données peuvent être enregistrées par des systèmes d'acquisition de données comme RTMaps, Matlab (Gruyer *et al.*, 2009; Livic, 2011).

Objectif de SiVIC

Le développement du logiciel SiVICTM a été lancé par le laboratoire Livic¹ du département COSYS² de l'Ifsttar³ (Bossu *et al.*, 2010; Gruyer *et al.*, 2009). Cette plateforme de simulation a été utilisée dans de nombreux projets : eMotive, Isi-PADAS, SafeSpot, Have-It, LoVE, D030, ARCOS. Le projet eMotive a pour objectif de fournir un produit industriel basé sur SiVIC et commercialisé par la société CIVITEC (Livic, 2011).

1. Laboratoire sur les Interactions Véhicules Infrastructure Conducteur

2. COMposant et SYStème

3. Institut Français des Sciences et Technologie des Transports, de l'Aménagement et des Réseaux

Dans (Bossu *et al.*, 2010), les auteurs partent du constat que l'évaluation des systèmes d'aide à la conduite (de type ADAS : *Advanced Driver Assistance Systems*) dans des environnement réel n'est pas aisée. Ils constatent que le contrôle et la répétitivité des expérimentations est difficile à mettre en œuvre dans la réalité, notamment pour des raisons de sécurité. Généralement, pour une évaluation correcte et exhaustive du système mis en place, un manque de capteurs, de données ou la complexité de l'expérimentation sont pénalisants. La difficulté de mise en œuvre du systèmes mis en place revient souvent. C'est pour ces raisons que le Livic a lancé le développement du logiciel SiVICTM.

Ce type de plateforme permet la modélisation d'un environnement routier composé de différents capteurs embarqués ou sur infrastructure. Les données qu'il fournit peuvent être enregistrés ce qui permet d'éviter l'investissement lié aux essais sur le terrain à un stade précoce du processus de conception de recherche et de développement. Ce simulateur donne la possibilité de tester des situations dangereuses et permet d'éviter dans un premier temps les difficultés liées à la mise en œuvre de véhicules d'essais réels et au stockage des données avant de passer au stade de la validation expérimentale.

A.2.2 Fonctionnalités du simulateur SiVIC

Les fonctionnalités de SiVIC sont multiples et nous n'en détaillons ici que quelques-unes. La plateforme SiVIC permet de simuler un environnement routier dans lequel évoluent des véhicules instrumentés. Créé et utilisé pour tester des systèmes d'assistance à la conduite de type ADAS (*Advanced Driver Assistance Systems*), ce logiciel doit être capable de synthétiser des données de différentes situations, à partir de différents types de capteurs. Les scénarios routiers sont ajustables. L'utilisateur peut ainsi choisir le type d'environnement routier, évoluer sur une route classique ou en ville avec des carrefours. En plus du choix de l'environnement routier, l'utilisateur peut choisir les conditions climatiques, soleil, pluie... Les auteurs de (Gruyer *et al.*, 2011), utilisent cette fonctionnalité. Pour leurs tests, ils mettent notamment en place un scénario aux conditions dégradées par la pluie. La modélisation de la route est complète, avec la possibilité d'ajouter plusieurs types d'infrastructures, de mobiliers urbains, d'environnement paysages (arbres, glissières de sécurité, signalisation horizontale, ponts...). Dans l'article (Revilloud *et al.*, 2012), les auteurs présentent notamment un outil générique pour générer les signalisations horizontales de la route. Cet outil est fondé sur la texture du bitume, il prend en compte le trafic et l'usure de la route pour modéliser le plus précisément possible les signalisations horizontales. Le logiciel SiVIC ne permet pas uniquement de modéliser des objets de type véhicule, mais également, des objets de type piéton comme le montre l'article (Bossu *et al.*, 2010).

A.2.3 Exemple de données obtenues avec le simulateur SiVIC pour un lidar multiplan

Le cas du lidar multiplan est un peu particulier. En effet, il n'existe pas encore de fonction dans le logiciel SiVIC permettant de générer des données capteur lidar multiplan. Cependant, comme on peut le voir sur la figure A.5 du bas et au milieu à droite, il est possible de générer des données lidar multiplan. Ces données sont obtenues à partir de plusieurs simulations monoplans. À chacune d'entre elle, le capteur lidar subit une rotation verticale qui permet l'élévation du tir laser (comme le montre le schéma figure A.4). Par la suite, les mesures obtenues par chacune des simulations sont concaténées en un unique vecteur constitué de plusieurs plans de balayage.

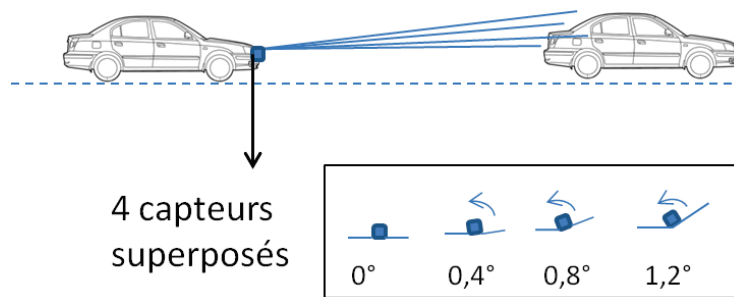


FIGURE A.4. Modélisation de la rotation du capteur lidar pour générer des données multiplans

De même que pour le simulateur de la thèse, les paramètres de simulation choisis pour le véhicule capteur, le capteur et la route sont résumés dans le tableau A.6, les paramètres du capteur étant proches de ceux des capteurs laser scanner LD d'IBEO.

	Notation	Valeur
Véhicule capteur	(v_x, v_y) en $m.s^{-1}$	(0, 20)
	Num_voie	2
Capteur	$(\theta_{min}, \theta_{max})$ en $^{\circ}C$	(-80, 80)
	$(dist_{min}, dist_{max})$ en m	(10, 180)
	$\Delta\theta$ en $^{\circ}C$	0,25
	Δt en s	0,05
Route	Nb_voie	3
	Num_voie	{1,2,3}
	$largeur_voie$ en m	3,5

Tableau A.6. Paramètres de simulation du véhicule capteur, du capteur et de la route pour un scénario donné associé au simulateur SiVIC

La figure A.5 présente ainsi, une scène vue par le capteur vidéo (en haut). Le scénario présenté est composé de quatre véhicules situés à différentes distances du véhicule porteur et dans différentes voies de circulation. Les deux figures du milieu présentent quand a elle cette même scène vue par le capteur lidar monoplan (à gauche) et le capteur lidar multiplan (à droite). Enfin, les deux figures du bas représentent deux agrandissements des véhicules de la voie centrale (à droite) et en bas de la voie de droite (à gauche) Le capteur monoplan n'a pas subi d'élévation du tir laser. Pour le capteur lidar multiplan le tir laser a une élévation de (0° , $0,4^{\circ}$, $0,8^{\circ}$, $1,2^{\circ}$) représenté sur la figure par les couleurs respectives bleu, rouge, vert et cyan. Sur la figure du bas à droite, le véhicule est loin du véhicule porteur du capteur, à cette distance, seul deux tirs d'élévation 0° et $0,4^{\circ}$ touchent le véhicule cible. Les tirs à élévation de $0,8^{\circ}$ et $1,2^{\circ}$ passent ainsi au dessus du véhicule cible.

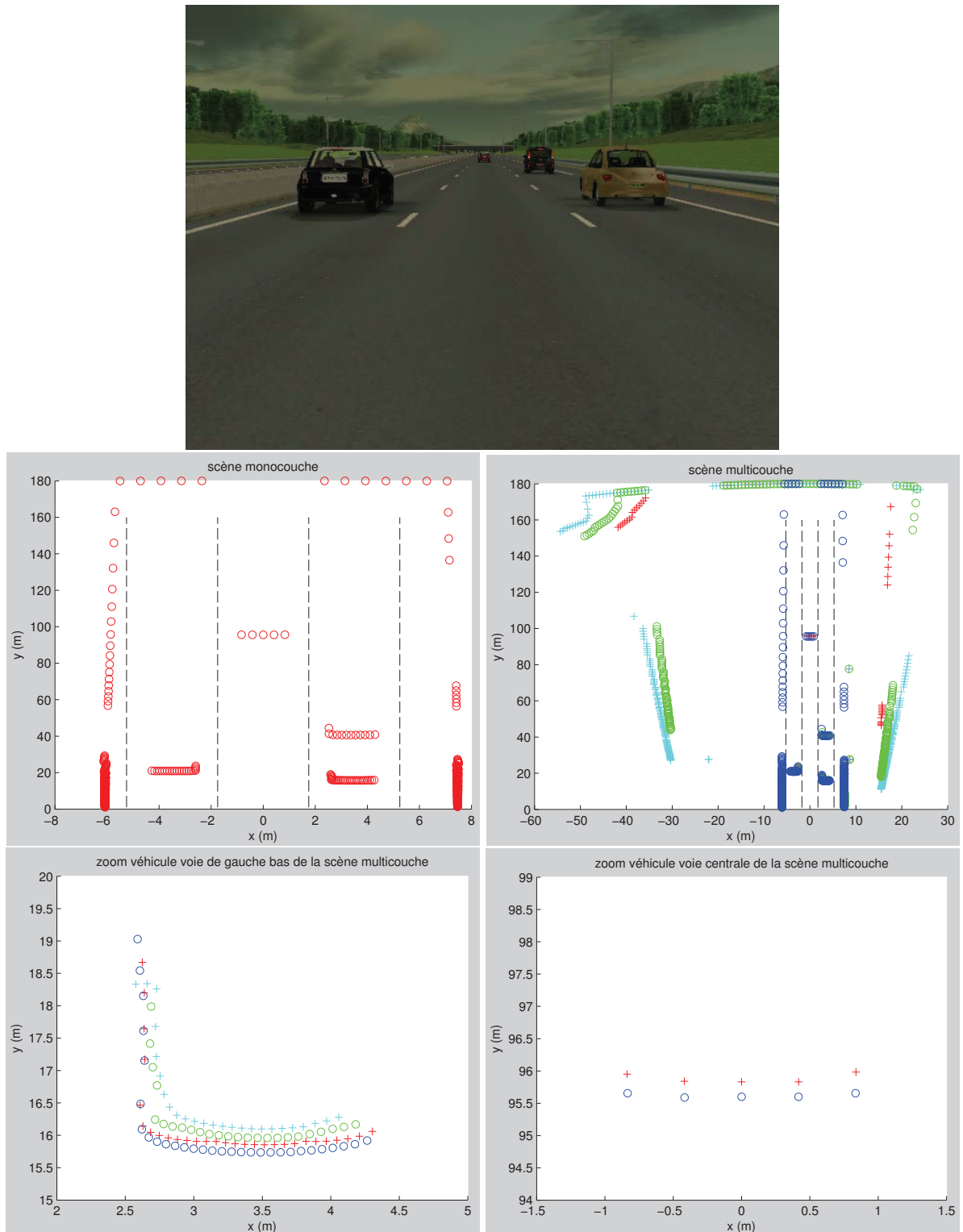


FIGURE A.5. Exemple de données générées par le simulateur SiVIC

Références bibliographiques

- Active - safety chalmers, 2014. URL <http://activesafety.wikispot.org/Lidar>. disponible le 06/03/2014.
- R. Arndt, R. Schweiger, W. Ritter, D. Paulus et O. Löhlein. Detection and tracking of multiple pedestrians in automotive applications. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 13–18, Istanbul, Turkey, June 13-15, 2007.
- M. S. Arulampalam, S. Maskell, N. Gordon et T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Trans. on Signal Processing*, 50(2) :174–188, 2002.
- R. J. Aulay et E. Denlinger. A decision directed adaptive tracker. *IEEE Trans. on Aerospace and Electronic Systems*, 9(2) :229–236, March, 1973.
- Q. Baig, O. Ayacard, T. D. Vu et T. Fraichard. Fusion between laser and stereo vision data for moving objects tracking in intersection like scenario. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 362–367, Baden-Baden, Germany, June 5-9, 2011.
- D. Bank. Short range radar signal processing for lateral collision warning in road traffic scenarios. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 111–116, Istanbul, Turkey, June 13-15, 2007.
- Y. Bar-Shalom et E. T. Fortmann. *Tracking and data association*, volume 179. Academic Press, 1988.
- Y. Bar-Shalom et X. R. Li. *Multitarget-multisensor tracking : principles and techniques*. YBS publishing, 1995.
- M. Bertozzi, A. Broggi, A. Fascioli, A. Tibaldi, R. Chapuis et F. Chausse. Pedestrian localization and tracking system with Kalman filtering. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 584–589, Parma, Italy, June 14-17, 2004.
- P. Besesty. Télémètre à laser puce. Disponible dans la base : Archives Analyse / mesures, mars, 1999. URL https://www.techniques-ingenieur.fr/dossier/telemetre_a_laser_puce/R1811_03_1999. Consulté le 13/05/2013.
- G. A. Borges et M. J. Aldon. A split-and-merge segmentation algorithm for line extraction in 2D range images. Dans *International Conference on Pattern Recognition (ICPR)*, volume 1, pages 1441–1444, September 3-8, 2000.
- G. A. Borges et M. J. Aldon. Line extraction in 2D range images for mobile robotics. *Journal of Intelligent and Robotic Systems*, 40(3) :267–297, July, 2004.
- J. Bossu, D. Gruyer, J. C. Smal et J. M. Blosseville. Validation and benchmarking for pedestrian video detection based on a sensors simulation platform. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 115–122, San Diego, CA, USA, June 21-24, 2010.

- S. Bourassa et S. Kamoun. A parallel filtering technique for a surveillance radar. Dans *IEEE Conference on Electrical and Computer Engineering*, volume 1, pages 55–58, Vancouver, Canada, September 14-17, 1993.
- Chameleon. Transport research and innovation portal, 2013. URL http://www.transport-research.info/web/projects/project_details.cfm?ID=2653. consulté le 10 septembre 2013.
- Z. Chen. Bayesian filtering : from Kalman filters to particle filters, and beyond. Rapport technique, Communication Research Laboratory, Mc Master University, Hamilton, ON, Canada, 2003.
- H. Cheng et N. Zheng. Interactive road situation analysis for driver assistance and safety warning systems : framework and algorithms. *IEEE Trans. on Intelligent Transportation Systems*, 8(1) :157–167, March, 2007.
- S. M. Chow, E. Ferrer et J. R. Nesselrode. An unscented Kalman filter approach to the estimation of nonlinear dynamical systems models. *Multivariate Behavioral Research*, 42(2) :283–321, 2007.
- L. Chuanjin, Q. Xiaohu, H. Xiyue, C. Yi et Z. Xin. A monocular vision based driver assistance system for collision avoidance. Dans *Proceedings of the IEEE 6th International Conference on Intelligent Transportation Systems*, volume 1, pages 463–468, Shanghai, China, October 12-15, 2003.
- V. Constantin, A. Waegli et J. Skaloud. Le filtre de Kalman unscented outil performant en géodésie cinématique. Dans *Géomatique Suisse*, volume 1, pages 12–17, 2008.
- D. Crisan et A. Doucet. A survey of convergence results on particle filtering methods for practitioners. *IEEE Trans. on Signal Processing*, 50(3) :736–746, March, 2002.
- K. Dahia. *Nouvelles méthodes en filtrage particulaire. Application au recalage de navigation inertielle par mesures altimétriques*. PhD thesis, Université Joseph Fourier, France, 2005.
- R. Danescu et S. Nedevschi. Probabilistic lane tracking in difficult road scenarios using stereovision. *IEEE Intelligent Transportation System*, 10(2) :272–282, June, 2009.
- M. Darms, P. Rybski et C. Urmson. Classification and tracking of dynamic objects with multiple sensor for autonomous driving in urban environments. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 1197–1202, Eindhoven, Netherland, June 4-6, 2008.
- A. P. Dempster, N. M. Laird et D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B*, 39(1) :1–38, 1977.
- K. Dietmayer, J. Sparbert et D. Streller. Model based object classification and object tracking in traffic scenes from range images. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 25–30, Tokyo, Japan, May 13-17, 2001.
- Diginext. *Estimation non linéaire optimale Filtrage Particulaire*. Le tertia II, Aix en provence, France, octobre 1996. URL srilas.univ-bpclermont.fr/ftp/pub/trasson/SLAM/biblioslam/pegale.pdf. Consulté le 05/01/11.
- A. Doucet, N. de Freitas, K. Murphy et S. Russel. Rao-Blackwellised particle filtering for dynamic Bayesian networks. Dans *Proceeding of the Annual Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 176–183, San Francisco, CA, USA, June 30- July 3, 2000a.
- A. Doucet, S. J. Godsill et C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10(3) :197–208, 2000b.

- R. O. Duda et P. E. Hart. Use of the Hough transformation to detect lines and curves in pictures. Rapport technique, AI Center, SRI International, 333 Ravenswood Ave, Menlo Park, CA 94025, April 1971. URL http://www.ai.sri.com/pub_list/.
- R. O. Duda et P. E. Hart. Use of the Hough transformation to detect lines and curves in pictures. *Communication of the ACM*, 15(1) :11–15, 1972.
- E. Duquenoy et A. Taleb-Ahmed. Applying the Hough transform pseudo-linearity property to improve computing speed. *Pattern Recognition Letters*, 27(16) :1893–1904, July, 2006.
- A. Eidehall et F. Gustafsson. Combined road prediction and target tracking in collision avoidance. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 619–624, Parma, Italy, June 14-17, 2004.
- A Eidehall, T. B. Schön et F. Gustafsson. The marginalized particle filter for automotive tracking application. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 370–375, Las Vegas, NV, USA, June 6-8, 2005.
- T. Einsele. Real time self localization in unknown indoor environments using a panorama laser range finder. Dans *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 2, pages 697–702, Grenoble, France, September 7-11, 1997.
- B. Fardi, U. Scheunert, H. Cramer et G. Wanielik. Multi modal detection and parameter based tracking of road borders with a laser scanner. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 95–99, Columbus, OH, USA, June 9-11, 2003.
- M. Fischler et R. Bolles. Random sample consensus : A paradigm for model fitting with application to image analysis and automated cartography. *Communications of the ACM*, 24(6) :381–385, 1981.
- D. A. Forsyth et J. Ponce, éditeurs. *Computer vision : A modern approach*. Prentice hall, 2003.
- B. Fortin, R. Lherbier et J.C. Noyer. Feature in scanning laser range data using invariant parameters : application to vehicle detection. *IEEE Transactions*, 61(9) :3838–3850, November, 2012.
- J. E. G. Freitas, M. Niranjan, A. H. Gee et A. Doucet. Sequential Monte Carlo to train network models. *Neural computation*, 12(4) :955–993, April, 2000.
- K. Fuerstenberg. Intersection safety - The EC project INTERSAFE. Dans *Intelligent Transport System (ITS)*, pages 432–436, San Francisco, CA, USA, November 6-10, 2005a.
- K. Fuerstenberg. New European approach for intersection safety - The EC project INTERSAFE. Dans *Intelligent Transport System (ITS)*, pages 532–536, Hanover, Germany, June, 2005b.
- K. Fuerstenberg, P. Baraud, G. Caporaletti, S. Citelli, Z. Eitan, U. Lages et C. Lavergne. Development of a pre-crash sensorial system - The CHAMELEON project. *VDI BERICHTE*, 1653 :289–310, 2001d.
- K. Fuerstenberg, K. C. J. Dietmayer, S. Eisenlauer et V. Willhoeft. Multilayer laserscanner for robust object tracking and classification in urban traffic scenes. Dans *Intelligent Transport System (ITS)*, pages 7–8, Chicago, IL, USA, November, 2002a.
- K. Fuerstenberg, K. C. J. Dietmayer et U. Lages. Object tracking and classification for multiple active safety and comfort applications using a multilayer laserscanner. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 802–807, Parma, Italy, June 14-17, 2004.
- K. Fuerstenberg, K. C. J. Dietmayer et V. Willhoeft. Pedestrian recognition in urban traffic using a vehicle based multilayer laserscanner. Dans *IEEE Intelligent Vehicles Symposium (IV)*, volume 1, pages 31–35, Versailles, France, June 17-21, 2002b.

- K. Fuerstenberg, T. Linzmeier et K. C. J. Dietmayer. Pedestrian recognition and tracking of vehicles using a vehicle based multilayer laserscanner. Dans *Intelligent Transport Systems (ITS)*, November, 2003.
- K. Fuerstenberg et V. Willhoeft. Object tracking and classification using laserscanner - pedestrian recognition in urban environment. Dans *IEEE International Conference on Intelligent Transport Systems (ITSC)*, pages 451–453, Oakland, CA, USA, August, 2001a.
- K. Fuerstenberg et V. Willhoeft. Pedestrian recognition in urban traffic using laserscanners. Dans *Intelligent Transport Systems (ITS)*, pages 31–35, Sydney, Australia, October, 2001b.
- K. Fuerstenberg, V. Willhoeft et K. C. J. Dietmayer. New sensor for 360° vehicle surveillance - innovative approach to stop & go, lane assistance and pedestrian recognition. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 3–7, Tokyo, Japan, May 13-17, 2001c.
- J. Ge, Y. Luo et G. Tei. Real time pedestrian detection and tracking at nighttime for driver assistance systems. *IEEE trans. on Intelligent Transportation Systems (ITSC)*, 10(2) :283–298, June, 2009.
- S. Gidel, C. Blanc, T. Chateau, P. Checchin et L. Trassoudaine. A methode based on multilayer laserscanner to detect and track pedestrians in urban environment. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 157–162, Xi'an, China, June 3-5, 2009.
- S. Gidel, P. Checchin, C. Blanc, T. Chateau et L. Trassoudaine. Pedestrian detection using a multilayer laserscanner. Dans *16th Congrès francophone AFRIF-RFIA*, Amien, France, January, 2008.
- N. J. Gordon, D. J. Salmond et A. F. M. Smith. Novel approach to non-linear/non-Gaussian Bayesian state estimation. *IEE Proceedings*, 140(2) :107–113, April, 1993.
- D. Gruyer, S. Glaser, S. Pechberti, R. Gallen et N. Hautiere. Distributed simulation architecture for the design of cooperative ADAS. Dans *Future Active Safety Technology (FAST-ZERO)*, pages 1–8, Tokyo, Japan, September 5-9, 2011.
- D. Gruyer, S. Glaser, B. Vanholme et B. Monnier. Simulation of automatic vehicle speed control by transponder equipped infrastructure. Dans *IEEE Intelligent Transport Systems Telecommunications (ITST)*, pages 628–633, Lille, France, October 20-22, 2009.
- D. Hall. *Mathematical Techniques in Multisensor data Fusion*. Artech House, 1992.
- D. Hall et J. Llinas. An introduction to multisensor data fusion. *Proceedings of the IEEE*, 85(1) :6–23, January, 1997.
- J. Heikkilä et O. Silven. A real time system for monitoring of cyclists and pedestrians. Dans *IEEE Workshop on Visual Surveillance (VS)*, pages 74–81, Fort Collins, CO, USA, July, 1999.
- T. Herpel, C. Lauer, R. German et J. Salzberg. Multisensor data fusion in automotive applications. Dans *International Conference on Sensing Technology*, pages 206–211, Taiwan, China, November 30 - December 3, 2008.
- F. Himm, N. Kaempchen, J. Ota et D. Burschka. Efficient occupancy grid computation on the GPU with lidar and radar for road boundary detection. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 1006–1013, San Diego, CA, USA, June 21-24, 2010.
- T. Huck, A. Westenberger, M. Fritzsche, T. Schwarz et K. Dietmayer. Precise timestamping and temporal synchronization in multi-sensor fusion. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 242–247, Baden-Baden, Germany, June 5-9, 2011.

- C. Idler, R. Schweiger, D. Paulus, M. Mahlich et W. Ritter. Realtime vision based multi-target-tracking with particle filters in automotive applications. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 188–193, Tokyo, Japan, June 13-15, 2006.
- B. Jida. *Intégration du contexte par réseaux bayésiens pour la détection et le suivi multi-cibles*. PhD thesis, Mémoire de thèse de l'Université du Littoral Côte d'Opale, Ifsttar, Leost, France, 2008.
- S. J. Julier et J. K. Uhlmann. A new extension of the Kalman filter to nonlinear systems. Dans *SPIE Aerosense Symposium*, Orlando, FL, USA, April, 1997.
- S. J. Julier et J. K. Uhlmann. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3) : 401–422, March, 2004.
- S. J. Julier et J. K. Uhlmann. Fusion of time delayed measurements with uncertain time delays. Dans *American Control Conference*, pages 4028–4033, Portland, OR, USA, June 8-10, 2005.
- R. E. Kalman. A new approach to linear filtering and prediction problems. Dans *Transaction of the ASME - Journal of Basic Engineering*, pages 35–45, March, 1960.
- R. Kandepu, B. Foss et L. Imsland. Applying the unscented Kalman filter for nonlinear state estimation. *Journal of Process Control*, 18(7/8) :753–768, August-September, 2008. URL dx.doi.org/10.1016/j.jprocont.2007.11.004.
- Z. Khan, T. Balch et F. Dellaert. MCMC based particle filtering for tracking a variable number of interacting targets. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(11) :1805–1819, November, 2005.
- A. Khitwongwattana et T. Maneewarn. Extended Kalman filter with adaptive measurement noise characteristics for position estimation of an autonomous vehicle. Dans *IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications*, pages 505–509, Beijing, China, October 12-15, 2008.
- M. Kiehn, V. Willhieft et U. Lages. Final laser sensor sample : Telematics applications programme sector transport, 3th September, 2002.
- J. Klein. Technique de filtrage particulaire. Rapport technique, LITIS, juillet 2006.
- P. Kmiotek et Y. Ruichek. Representing and tracking of dynamics objects using oriented bounding box and extended Kalman filter. Dans *IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 322–328, Beijing, China, October 12-15, 2008.
- D. Koller, K. Daniilidis, T. Thorhallsson et H. Nagel. Model-based object tracking in traffic scenes. Dans *ECCV*, pages 437–452, 1992.
- F. LeGland. Filtrage particulaire. URL www.math.u-bordeaux1.fr/~delmoral/gretsi03.pdf. Consulté le 05/01/11, 2003.
- M. E. II Liggins, C. Chee-Yee, I. Kadar, M. G. Alford, V. Vannicola et S. Thomopoulos. Distributed fusion architectures and algorithms for target tracking. *Proceedings of the IEEE*, 85(1) :95–107, 1997.
- P. Lindner et G. Wanielik. 3D lidar processing for vehicle safety and environment recognition. Dans *IEEE Workshop on Computational Intelligence in Vehicles and Vehicular Systems*, pages 66–71, March 30 - April 2, 2009.
- G. Liu, F. Wörgöttee et I. Markelic. Combining statistical Hough transform and particle filter for robust lane detection and tracking. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 993–997, San Diego, CA, USA, June 21-24 2010.

- W. Liu, H. Zhang, B. Duan, H. Yuan et H. Zhao. Vision based real time lane marking detection and tracking. Dans *IEEE Intelligent Transportation System (ITSC)*, pages 49–54, Beijing, China, October 12-15, 2008.
- Y. Liu, R. Emery, D. Chakrabarti, W. Burgard et S. Thrun. Using EM to learn 3D models of indoor environments with mobile robots. Dans *International Conference on Machine Learning*, pages 329–336, San Francisco, MA, USA, 28 June - 1 July 2001. Morgan Kaufmann.
- Livic. Une plateforme pour capteurs, véhicules et environnements virtuels, 2011. URL www.inrets.fr/linstitut/unites-de-recherche-unites-de-service/livic/themes-de-recherche/experimentation-et-evaluation/outils-de-simulation.html. Consulté le 2 août 2011.
- O. Masoud et N. P. Papanikolopoulos. A novel method for tracking and counting pedestrians in real time using a single camera. *IEEE Trans. on Vehicular Technology*, 50(5) :1267–1278, September, 2001.
- M. Meuter, U. Lurgel, S. B. Park et A. Kummert. The unscented Kalman filter for pedestrian tracking from a moving host. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 37–42, Eindhoven, Netherland, June 4-6, 2008.
- M. Meuter, S. Muller-Schneiders, A. Mika, S. Hold, C. Nunn et A. Kummert. A novel approach to lane detection and tracking. Dans *IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6, St. Louis, MO, USA, October 3-7, 2009.
- S. Moqqaddem, Y. Ruichek, R. Touahni et A. Sbihi. A spectral clustering and Kalman filtering based objects detection and tracking using stereo vision with linear cameras. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 902–907, Baden-Baden, Germany, June 5-9, 2011.
- P. Del Moral, G. Rigal et G. Salut. Estimation et commande optimale non-linéaire : un cadre unifié pour la résolution particulière. contrat dret-digilog, LAAS/CNRS, 1992.
- M. Morelande et B. Moran. Multiple target detection and tracking with a sensor network. Dans *International Conference on Information Fusion*, pages 1–8, Québec, Canada, 9-12 July 2007.
- F. Nashashibi et A. Bargetin. Laser-based vehicles tracking and classification using occlusion reasoning and confidence estimation. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 847–852, Eindhoven, Netherland, June 4-6, 2008.
- V. Nguyen, A. Martinelli, N. Tomatis et R. Siegwart. A comparison of line extraction algorithms using 2D laser rangefinder for indoor mobile robotics. Dans *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1929–1934, Edmond, Canada, August 2-6, 2005.
- V. Nguyen, A. Martinelli, N. Tomatis et R. Siegwart. A comparison of line extraction algorithms using 2D range data for indoor mobile robotics. *Autonomous Robots*, 23(2) :97–111, August, 2007.
- P. J. Nordlund. Recursive state estimation of nonlinear systems with applications to integrated navigation. Rapport technique, Linköping University, Departement of Electrical Engineering, Sweden, November 29, 2000.
- P. J. Nordlund. *Sequential Monte Carlo filters and integrated navigation*. PhD thesis, Linköping, Department of Electrical Engineering, Sweden, 2002.
- T. Ogawa, H. Sakai et Y. Suzuki. Pedestrian detection and tracking using in vehicle lidar for automotive application. Dans *IEEE Intelligent Vehicles Symposium (IV)*, Baden-Baden, Germany, June 5-9, 2011.
- T. Ogawa et K. Takagi. Lane recognition using on-vehicle lidar. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 540–545, Tokyo, Japan, June 13-15, 2006.

- D. Olmeda, A. De La Escalera et J. M. Armingol. Detection and tracking of pedestrians in infrared images. Dans *IEEE International Conference on Signals, Circuits and Systems*, pages 1–6, Medenine, Tunisia, November 6-8, 2009.
- R. O’Malley, M. Glavin et E. Jones. Vision based detection and tracking of vehicles to the rear with perspective correction in low light conditions. *Intelligent Transport System (IET)*, 5(1) :1–10, 2011.
- C. Otto, W. Gerber, F. P. Leon et J. Wirtzner. A joint integrated probabilistic data association filter for pedestrian tracking across blind regions using monocular camera and radar. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 636–641, Alcala de Henares, Spain, June 3-7, 2012.
- T. Pavlidis et S. L. Horowitz. Segmentation of plane curves. *IEEE Trans. on Computers*, 23(8) :860–870, 1974.
- M. Perrollaz. *Détection d’obstacles multi-capteurs supervisée par stéréovision*. PhD thesis, Pierre et Marie Curie, 13 juin, 2008.
- A. Petrovskaia, M. Perrolaz, L. Oliviera, L. Spinello, R. Triebel, A. Makris, J.D. Yoder, C. Laugier, U. Nunes et P. Bessiere. Awareness of road scene participants for autonomous driving. Dans *Handbook of Intelligent Vehicles*, pages 1383–1432, 2012.
- A. Petrovskaia et S. S. Thrun. Model based vehicle detection and tracking for autonomous urban driving. Dans *Autonomous Robots Journal*, volume 26, pages 123–139, April 2009.
- A. Petrovskaia et S. Thrun. Model based vehicle tracking for autonomous driving in urban environment. Dans *RSS*, 2008.
- S. T. Pfister, S. I. Roumeliotis et J. W. Burdick. Weighted line fitting algorithms for mobile robot map building and efficient data representation. Dans *IEEE International Conference on Robotics and Automation*, Taipei, Taiwan, September 14-19, 2003.
- M. Pitt et N. Shephard. Filtering via simulation : auxiliary particle filter. *American Statistical Association*, 94(446) :590–599, June, 1999.
- B. K. Ray et K. S. Ray. A new split-and-merge technique for polygonal approximation of chain coded curves. *Pattern Recognition Letters*, 16(2) :161–169, February, 1995.
- D. B. Reid. An algorithm for tracking multiple targets. *IEEE Trans. on Automatic Control*, 24(6) :843–854, December, 1979.
- M. Revilloud, D. Gruyer et E. Pollard. Generator of road marking textures and associated ground truth. Dans *IEEE Intelligent Transportation System (ITSC)*, pages 933 – 938, Anchorage, AK, USA, September 16-19, 2012.
- E. Richter, R. Schubert et G. Wanielik. Radar and vision based data fusion - Advanced filtering techniques for a multi object vehicle tracking system. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 120–125, Netherland, Eindhoven, June 4-6, 2008.
- A. Robert. Estimating optimal tracking filter performance for manned maneuvering targets. *IEEE Trans. on Aerospace and Electronic Systems*, 6(4) :473–483, July 1970.
- B. Rössler et K. Fürstenberg. Intersection safety - the EC project INTERSAFE, 2013. URL http://link.springer.com/chapter/10.1007%2F3-540-33410-6_31. consulté le 10 septembre 2013.

- S. Santos, J. E. Faria, F. Soares, R. Araujo et U. Nunes. Tracking of multi-obstacles with laser range data for autonomous vehicles. Dans *Proceeding of the 3rd Nat. Festival of Robotics Scientific Meeting (ROBOTICA 03)*, pages 59–65, Lisbonne, Portugal, May 9, 2003.
- S. Sato, M. Hashimoto, M. Takita, K. Takagi et T. Ogawa. Multilayer lidar-based pedestrian tracking in urban environments. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 849–854, San Diego, CA, USA, 2010.
- Sécurité-routière. Bilans annuels de la sécurité routière en france, 2012.
URL <http://www.securite-routiere.gouv.fr/la-securite-routiere/1-observatoire-national-interministeriel-de-la-securite-routiere/bilans-annuels/bilans-annuels-de-la-securite-routiere-en-france>. consulté le 16 avril 2013.
- X. Shao, H. Zhao, K. Nakamura, K. Katabira, R. Shibasaki et Y. Nakagawa. Detection and tracking of multiple pedestrians by using laser range scanners. Dans *IEEE Intelligent Robots and Systems (RSJ)*, pages 2174–2179, San Diego, CA, USA, October 29 - November 2, 2007.
- A. Siadat, A. Kaske, S. Klausmann, M. Dufaut et R. Husson. An optimized segmentation method for a 2D laser scanner applied to mobile robot navigation. *Proceeding of the 3rd IFAC Symposium on Intelligent Components and Instruments for Control Application*, 3 :153–158, June 9-11, 1997.
- M. Skutek, M. Mekhaïel et G. Wanielik. A precrash system based on radar for automotive applications. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 37–41, Columbus, OH, USA, June 9-11, 2003.
- M. Spengler. *On the applicability of sequential Monte Carlo Methods to multiple target tracking*. PhD thesis, Swiss Federal Institute of Technology Zurich, 2005.
- K. Takagi, K. Morikawa, T. Ogawa et M. Saburi. Road environment recognition using on vehicle lidar. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 120–125, Tokyo, Japan, June 13-15, 2006.
- A. Takeuchi, S. Mita et D. McAllester. On road vehicle tracking using deformable object model and particle filter with integrated likelihood. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 1014–1021, San Diego, CA, USA, June 21-24, 2010.
- Y. Tan, F. Han et F. Ibrahim. A radar guided vision system for vehicle validation and vehicle motion characterization. Dans *IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 1059–1066, Seattle, WA, USA, September 30 - October 3, 2007.
- F. Tango, E. Richter, U. Scheunert et G. Wanielik. Advanced multiple objects tracking by fusing radar and image sensor data - application on a case study. Dans *International Conference on Information Fusion*, pages 1–7, Cologne, Germany, June 30 - July 3, 2008.
- Z. Teng, J. H. Kim et D. J. Kang. Real time lane detection by using multiple cues. Dans *International Conference on Control, Automation and Systems (ICROS)*, pages 2334–2337, Gyeonggi-do, Korea, October 27-30, 2010.
- D. V. Thombre et J. H. Nirmal. Human detection and tracking using image segmentation and Kalman filter. Dans *International Conference on Intelligent Agent Multi-Agent Systems (IAMA)*, pages 1–5, Chennai, India, July 22-24, 2009.
- M. Thuy et F. P. Leon. Non-linear, shape independent object tracking based on 2D lidar data. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 532–537, Xi'an, China, June 3-5, 2009.

- M. Tian, F. Liu et Z. Hu. Single camera 3D lane detection and tracking based on EKF for urban intelligent vehicle. Dans *IEEE International Conference on Vehicular Electronics and Safety (ICVES)*, pages 413–418, Shanghai, China, December 13-15, 2006.
- E. Vanpoeringhe, M. Wahl et J. C. Noyer. Model-based detection and tracking of vehicle using a scanning laser rangefinder : a particle filtering approach. Dans *IEEE Intelligent Vehicles Symposium (IV)*, pages 1144–1149, Alcalá de Henares, Spain, June 3-7, 2012a.
- E. Vanpoeringhe, M. Wahl et J. C. Noyer. Detection and tracking of moving objects for automotive driver assistance system. Dans *Transport Research Arena (TRA)*, Athens, Greece, April 23-26, 2012b.
- Velodyne. High definition lidar HDL-32E, 2013. URL http://velodynelidar.com/lidar/hdlproducts/97-0038d%20HDL-32E_datasheet.pdf. Consulté le 9 septembre 2013.
- M. Wahl. CARSENSE project - Telematics applications programme - Sector transport (5ème PCRD), 2002. Deliverable D4.b : Architecture, livrable sur le système de fusion de données, CARSENSE Project IST-1999-12224.
- E. Waltz et J. Llinas. *Multisensor Data fusion*. Artech House, 1990.
- E. A. Wan et R. Van Der Merwe. The unscented Kalman filter for nonlinear estimation. Dans *IEEE Symposium on Adaptive Systems for Signal Processing Communications and Control (AS-SPCC)*, pages 153–158, Canada, October, 2000.
- E. A. Wan et R. Van Der Merwe. Chapter 7 : The unscented Kalman filter. Dans S. Haykins, éditeur, *Kalman filtering and Neural Networks*, pages 221–282. Wiley Publishing, September, 2001.
- C. Wang, Z. Hu et K. Uchimura. A novel lane detection approach fusion by vehicle localization. Dans *Proceedings of the 8th World Congress on Intelligent Control and Automation*, pages 1218–1223, Taipei, Taiwan, June 21-25, 2011.
- H. Wang, R. Lu, X. Wu, L. Zhang et J. Shen. Pedestrian detection and tracking algorithm design in transportation video monitoring system. Dans *IEEE Information Technology and Computer Science (ITCS)*, pages 53–56, Kiev, Ukraine, July 25-26 2009a.
- L. Wang, S. Pan et Y. Zheng. Moving vehicle tracking based on unscented Kalman filter algorithm. Dans *World congress on Computer Science and Information Engineering*, volume 2, pages 33–38, Los Angeles, CA, USA, 31 March - 2 April, 2009.
- W. S. Wijesoma, K. R. S. Kodagoda et A. P. Balasuriya. Road boundary detection and tracking using lidar sensing. *IEEE Trans. on Robotics and Automation (ICRA)*, 20(3) :456–464, June, 2004.
- Wikipedia. Système anti-blocage des roues, 2013. URL http://fr.wikipedia.org/wiki/Syst%C3%A8me_anti-blocage_des_roues. Consulté le 9 septembre 2013.
- F. Xu et K. Fujimura. Pedestrian detection and tracking with night vision. Dans *IEEE Intelligent Vehicles Symposium (IV)*, volume 1, pages 21–30, Versailles, France, June 17-21, 2002.
- Y. W. Xu, X. B. Cao et T. Li. Extend Kalman filter based pedestrian localization for collision avoidance. Dans *Proceeding of the 2009 IEEE International Conference on Mechatronics and Automation (ICMA)*, pages 4366–4370, Changchun, China, August 9-12, 2009.
- L. Zhang et B. K. Ghosh. Line segment based map building and localization using 2D laser rangefinder. *Proceeding of the International Conference on Robotics and Automation (ICRA)*, 3 :2538–2543, April 24-28, 2000.

M. Zuliani. Ransac for dummies (draft under gnu free documentation license). Rapport technique, Vision research lab (URL : vision.ece.ucsb.edu/~zuliani), 13th August, 2012.

Résumé :

Dans cette thèse, nous nous intéressons à la résolution de problèmes de détection et de suivi d'objets mobiles multiples sur route, à partir de données télémétrique de type lidar à balayage.

Les travaux dans le domaine de la détection et de suivi d'obstacles à partir de données lidar mettent généralement en œuvre trois principales étapes : la détection, l'association de mesures et le filtrage. Cependant, il est connu que cette chaîne de traitement peut engendrer des pertes d'informations pouvant être à l'origine de cas de non détection ou de fausse alarme. Par ailleurs, les non-linéarités liées à la transformation polaire-cartésien des mesures lidar au cours de l'étape de détection ne permettent plus de préserver la statistique des bruits de mesure après traitement. Une autre difficulté, compte-tenu de la nature spatialement distribuée des mesures lidar liées à un objet, est de pouvoir associer chaque impact à un unique véhicule tout en prenant en compte la variabilité temporelle du nombre d'impacts à lui associer. Seule une approche exploitant les données brutes permet de garantir l'optimalité de la chaîne de traitement.

Cette thèse explore une nouvelle approche conjointe de détection et de suivi exploitant les données brutes du lidar, éliminant toute étape de pré-détection. L'approche proposée repose, d'une part, sur l'usage des méthodes de Monte Carlo séquentielles en raison de leur aptitude à traiter des modèles fortement non linéaire, et, d'autre part, sur une modélisation des objets compatible avec la perception lidar. La méthode est validée avec des données du simulateur SIVIC dans différentes situation expérimentales pour la détection et le suivi d'objets hétérogènes dans un cas lidar monoplan puis multiplan.

Mots-clés : détection, suivi d'objets, filtre particulaire, mesures lidar

Abstract :

This thesis focus on the problem of multiobject detection and tracking multiple moving objects on the road, using a scanning laser rangefinder.

The works in the field of obstacle detection and tracking from lidar data generally use three main stages : detection, measurement association and filtering. However, it is known that this processing chain can lead to a loss of information that may be responsible for non-detection or false alarm problems. Furthermore, the non-linearities associated to the polar-to-Cartesian transformation of lidar measurements during the detection step cannot preserve the statistical properties of the measurement noise. Another difficulty, related to the spatially distributed nature of a lidar measurements of an object, is to associate each impact with a single vehicle while taking into account the temporal variability of the number of impacts. An approach that only exploits the raw data ensures the optimality of the processing chain. This thesis explores a new joint approach for detection and tracking that uses raw lidar data, while eliminating any step of predetection. The proposed approach is based , first , on the use of sequential Monte Carlo methods due to their ability to deal with highly non-linear models , and secondly , on an object modeling related to lidar measure. The method is validated with data from the simulator SIVIC under different experimental conditions for the detection and tracking of heterogeneous objects with monolayer and multilayer lidar.

Keywords : detection, objects tracking, particle filters, lidar measurements