



**HAL**  
open science

# Évaluation de systèmes d'aide à la conduite. Génération automatique de vérité terrain augmentée à partir d'un capteur haute résolution et d'une cartographie sémantique et 3D ; Evaluation de fonctions de perception tierces

Rémi Defraiteur

## ► To cite this version:

Rémi Defraiteur. Évaluation de systèmes d'aide à la conduite. Génération automatique de vérité terrain augmentée à partir d'un capteur haute résolution et d'une cartographie sémantique et 3D ; Evaluation de fonctions de perception tierces. Automatique / Robotique. Université Paris-Saclay, 2021. Français. NNT : 2021UPASG040 . tel-03359644

**HAL Id: tel-03359644**

**<https://theses.hal.science/tel-03359644>**

Submitted on 30 Sep 2021

**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.

Evaluation de systèmes d'aide à la conduite.  
Génération automatique de vérité terrain augmentée à partir d'un capteur haute résolution et d'une cartographie sémantique et 3D ;  
Evaluation de fonctions de perception tierces.

## Thèse de doctorat de l'université Paris-Saclay

Ecole doctorale n°580  
Sciences et technologies de l'information et de la communication (STIC)  
Spécialité de doctorat : Robotique  
Unité de recherche : Université Paris-Saclay, ENS Paris-Saclay, CNRS, SATIE,  
91190, Gif-sur-Yvette, France  
Réfèrent : Faculté des sciences d'Orsay

Thèse présentée et soutenue à Gif-sur-Yvette,  
le 15/06/2021, par

**Rémi DEFRAITEUR**

### Composition du Jury

|   |                         |
|---|-------------------------|
| <b>Samia BOUCHAFA-BRUNEAU</b><br>Professeure des universités, Université Evry | Présidente              |
| <b>Pierre CHARBONNIER</b><br>Directeur de recherche, CEREMA Est               | Rapporteur et Examineur |
| <b>François GOULETTE</b><br>Professeur des universités, Mines ParisTech       | Rapporteur et Examineur |
| <b>Guillaume BRESSON</b><br>Ingénieur de recherche, VEDECOM                   | Examineur               |
| <b>Anne VERROUST-BLONDET</b><br>Chargée de recherche, INRIA Paris             | Examinatrice            |

### Direction de la thèse

|   |                       |
|---|-----------------------|
| <b>Roger REYNAUD</b><br>Professeur des universités, Université Paris-Saclay | Directeur de thèse    |
| <b>Sergio RODRIGUEZ</b><br>Maître de conférences, Université Paris-Saclay   | Co-Encadrant          |
| <b>Nicole EL ZOGHBY</b><br>Ingénieure, Groupe Renault                       | Co-Encadrante         |
| <b>Marie-Anne MITTET</b><br>Ingénieure, Groupe Renault                      | Tutrice en entreprise |

# Remerciements

A l'issue de ces travaux, je tiens à adresser des remerciements aux personnes sans qui cette thèse n'aurait pu aboutir et qui font, de façon directe ou indirecte, parties intégrantes de ces travaux.

Tout d'abord, je souhaite remercier l'ensemble des membres du jury pour l'intérêt porté à mes travaux, ainsi que pour les discussions qui ont pu se tenir lors de la soutenance. Je remercie particulièrement Samia Bouchafa-Bruneau, pour avoir endossé le rôle de présidente. Je remercie également Pierre Charbonnier et François Goulette, pour m'avoir fait l'honneur d'être rapporteurs de mes travaux. Les différents retours que vous m'avez fait, ainsi que les échanges que nous avons pu avoir, m'ont permis d'améliorer davantage la qualité de ce document.

J'adresse mes plus vifs remerciements Roger Reynaud et Sergio Rodriguez pour le temps que vous m'avez consacré, pour les divers échanges, pour vos nombreuses relectures ainsi que pour m'avoir guidé au cours de cette période qui n'a pas toujours été facile à gérer dans ces conditions particulières. Votre aide et vos nombreux conseils ont permis de donner à ces travaux la qualité qu'on leur trouve aujourd'hui.

Je remercie tout autant Marie-Anne Mittet et Nicole El Zoghby. Vous m'avez accordé votre confiance à la suite de mon stage et vous avez cru en mes capacités à réaliser cette thèse. Vous êtes à l'origine de ce projet et je n'aurai pu mener ces travaux sans vous. Je vous remercie d'avoir su trouver du temps à m'accorder malgré un contexte compliqué et des plannings chargés. Vous m'avez permis de vivre une expérience très enrichissante qui me servira dans le futur.

Mes remerciements vont aussi à l'ensemble des collègues que j'ai pu côtoyer au cours de ces travaux. Au sein du Groupe Renault, je remercie l'ensemble des équipes dont j'ai pu faire partie et particulièrement Franck, Romain, David, Eric, Johann et Ghayath pour les nombreuses discussions intéressantes que nous avons pu avoir. Pour les mêmes raisons, je remercie toute l'équipe MOSS du laboratoire SATIE. J'ai une pensée particulière à Flavien pour tous les cafés à discuter de groupes de rock et de concerts, ainsi qu'à Emi pour toutes les soirées pizzas partagées. Je pense également aux co-doctorants du labo avec qui j'ai partagé cette expérience de mener une thèse : Alireza, Huichin, Hernan, Jennifer, Nicolas, Christophe et Pauline, pour les différentes discussions partagées et sorties organisées.

Je tiens également à remercier mes amis et mes proches qui m'ont toujours épaulé au cours de cette période et ont toujours répondu présent pour partager de bons moments. Tout le monde ne peut pas être nommé ici, mais je tiens tout de même à citer ma sœur Solène, mon frère Baptiste, Thomas qui est comme un frère, ainsi que certains de mes plus proches amis : Victor, Julie, Pierre, Claire, Laurent, Julien et Benjamin. Enfin, mes remerciements les plus chaleureux s'adressent à mes parents et à Anne-Louise. Ils ont toujours été derrière moi et ont fait preuve d'une grande patience au long de ces trois années pour me changer les idées, m'encourager et me redonner confiance pour mener ce travail à son terme.



# Résumé

Le véhicule autonome représente un des défis technologiques actuels majeurs dans le secteur automobile. Les véhicules se complexifient et intègrent de nouvelles fonctionnalités telles que la perception, qui joue un rôle clé dans ces systèmes. Permettant au véhicule d'appréhender l'environnement dans lequel il évolue, elle est exploitée sous différents aspects pour garantir une mobilité plus sûre. Pour rendre cela possible, les véhicules sont équipés de différentes solutions de perception. Celles-ci rassemblent différents types de capteurs (LiDARs, caméras, radars, ultrasons...), et une couche logicielle chargée de traiter les données des capteurs et d'interpréter l'environnement dans lequel le véhicule évolue. Étant donné le rôle essentiel de la perception dans le bon comportement d'un véhicule autonome, il est nécessaire de s'assurer que les solutions de perception exploitées soient suffisamment performantes pour garantir une circulation sécurisée.

Ces travaux concernent donc l'évaluation des systèmes implémentant des techniques d'intelligence artificielle. De telles techniques existent aujourd'hui et passent par l'annotation manuelle de jeux de données, ce qui limite alors l'évaluation à quelques jeux de données reconnus.

L'objectif de cette thèse est de mettre au point un nouvel outil de validation permettant d'évaluer, en utilisant le minimum d'annotations manuelles, les performances et niveaux d'erreurs de différentes solutions de perception. Via cet outil, il sera possible de mettre en concurrence différentes solutions en se basant sur des critères communs. La mise au point de cet outil se décompose en deux parties principales qui sont : la génération automatisée de données de référence et la méthode d'évaluation des solutions de perceptions testées.

Les travaux réalisés au cours de cette thèse ont permis de mettre au point un prototype d'outil de validation. La collaboration entre un laboratoire universitaire et un industriel de l'automobile, ainsi que les critères propres à leur domaine de recherche, ont permis de prendre en compte les différentes contraintes auxquelles cet outil de validation doit répondre. L'originalité de la méthodologie développée se situe dans l'utilisation conjointe de deux sources de données : des données issues de la simulation de la scène ; et des données issues du capteur de référence embarqué. Le capteur de référence correspond à un LiDAR de haute définition permettant de numériser l'environnement du véhicule fidèlement lors des acquisitions. Les données simulées sont créées en exploitant un modèle 3D correspondant à l'environnement de test, couplé à un modèle de capteur reprenant les caractéristiques physiques du capteur de référence. Les données simulées sont alors très proches des données réelles et contiennent de plus des informations sémantiques provenant du modèle 3D de la scène. L'association entre les données générées par simulation et celles acquises via le capteur de référence permet d'annoter automatiquement ces dernières qui deviennent alors exploitables en tant que référence annotée lors des évaluations. Afin de développer et d'expérimenter l'approche, des acquisitions ont permis d'enregistrer des jeux de données avec différentes solutions de perceptions (considérées comme des composants sur étagère, fournis par des équipementiers). Ces enregistrements ont également permis de tester la robustesse de la méthodologie, de mettre en avant les points forts, mais également de mesurer ses limites d'exploitation.

**Mots clés :** robotique, perception, véhicules autonome, ADAS, évaluation, caractérisation, performances.

# Abstract

Autonomous driving is one of the current major technological challenges in the automotive sector. Vehicles are becoming more complex and are integrating new functionalities such as perception, which plays a key role in such systems. It is used in various ways to ensure safer mobility, allowing the main inboard system to understand the environment in which the vehicle evolves. To achieve this, vehicles are equipped with different perception solutions. They include different types of sensors (LiDARs, cameras, radar, ultrasound, etc.), and a software layer used to process the sensor data and to infer the environment in which the vehicle is moving. Perception plays a critical role in the proper behavior of an autonomous vehicle. It is necessary to ensure that the embedded perception solutions are effective enough to meet safe driving requirements. This work deals with the evaluation of systems implementing artificial intelligence techniques. Such techniques are used today and are based on manually annotated datasets, limiting the evaluation to a few popular datasets.

The purpose of this thesis is to develop a new validation tool to evaluate the performances and error levels of different perception solutions, while minimizing the manual annotations. With this tool, it will be possible to lead benchmark studies on different solutions based on common criteria. The development of this tool is split into two main parts : the automated production of reference data and the evaluation method of the tested perception solutions

The works done during this thesis have made it possible to develop a validation tool prototype. The collaboration between a university laboratory and an automotive manufacturer, as well as the criteria specific to their field of research, made it possible to consider the different constraints that this validation tool must satisfy. The originality of the developed methodology lies in the joint use of two data sources: data from the scene simulation and data from the onboard reference sensor. The reference sensor corresponds to a high-definition LiDAR, scanning accurately the vehicle environment during the acquisitions. Synthetic data are created by exploiting a 3D model representing the test environment and the sensor model based on the physical characteristics of the reference sensor. The synthetic data are close to the real one, and also include the semantic information from the 3D model of the scene. The association between data acquired from the reference sensor and produced synthetic data grants access to automatic labeling of the real data measured by the LiDAR, that then become usable as a reference during the evaluation process. To develop and experiment with the approach, acquisitions made it possible to record datasets with different perception solutions (considered as COTS, given by industrials). These datasets were also used to test the reliability of the methodology, to highlight the strengths, but also to measure its limitations.

**Key words :** robotics, perception, autonomous driving, ADAS, evaluation, characterization, performances.

# Symboles et notations

## Modèles et données de capteurs

$P$  : plan image ;

$f$  : distance focale ;

$pp$  : point principal ;

$u_0, v_0$  : coordonnées du point principal ;

$C$  : repère caméra ;

$M$  : point en 3 dimensions dans une scène perçue par une caméra ;

$m$  : projection de  $M$  dans le plan image ;

$f_x, f_y$  : distances focales selon les axes  $x$  et  $y$  du plan image ;

$u, v$  : coordonnées d'un pixel dans le plan image ;

$X_i, Y_i, Z_i$  : coordonnées d'un point 3D exprimées dans le référentiel défini par le repère  $i$  ;

$entraxe$  : entraxe séparant les deux caméra dans le cadre de la vision stéréoscopique ;

$disp$  : disparité ;

$D$  : distance commune mesurée ;

$c$  : célérité ;

$\Delta t$  : interval de temps mesuré entre l'émission et la réception d'une onde ;

$F$  : fréquence de mesure d'un capteur ;

$\Delta\varphi$  : déphasage mesuré entre l'onde émise et l'onde reçue ;

$L$  : LiDAR commun ;

$L_{ref}$  : LiDAR utilisé comme capteur de référence dans le cadre de ce travail ;

$L_{synth}$  : capteur modélisé dans le cadre de la simulation ;

$M_L$  : modèle de capteur LiDAR commun ;

$M_{L_{estim}}$  : modèle de capteur estimé à partir de mesures acquises en conditions réelles ;

$M_{L_{const}}$  : modèle de capteur dont les paramètres sont transmis par le constructeur ;

$M_{L_{synth}}$  : modèle composé des caractéristiques physiques d'un capteur modélisé ;

$M_{L_{ref}}$  : modèle de capteur dont les caractéristiques physiques correspondent à celles de  $L_{ref}$  ;

$p$  : point 3D composé, au minimum, des coordonnées  $X, Y$  et  $Z$  ;

$p_{synth}$  : point LiDAR généré par simulation ;

$I$  : attribut d'un point LiDAR et correspondant à l'intensité du signal renvoyé lors de la mesure ;

$ring$  : indice de la nappe LiDAR à laquelle un point  $p$  appartient ;

$time$  : horodatage d'un point  $p$  ;

*label* : valeur numérique associée à une classe sémantique spécifique ;  
*Az* : angle azimutal ;  
*r<sub>Az</sub>* : résolution azimutale ;  
 $\mathcal{V}$  : angle d'élévation ;  
 $\mathcal{V}_{ring}$  : angle d'élévation en fonction de l'indice de nappe laser ;  
*scan* : ensemble de points obtenu à l'issue d'un balayage azimutal complet d'un LiDAR ;  
*scan<sub>réel</sub>* : scans issus des mesures de  $L_{ref}$ , acquises en condition réelles ;  
*scan<sub>synth</sub>* : scans de synthèse générés par simulation grâce au modèle  $L_{model}$  ;  
*D<sub>max</sub>* : portée de mesure maximale d'un capteur donné ;  
 $\epsilon_{biais,i}$  : bruit de mesure utilisé par BlenSor ;  
 $\epsilon_{mesure}$  : bruit de mesure utilisé par BlenSor ;  
 $\mathcal{N}(0, \sigma)$  : loi normale centré sur 0 et d'écart-type  $\sigma$  ;  
 $\sigma_{Az}$  : écart-type de la résolution azimutale pour un capteur LiDAR donné ;  
*e* : erreur de position d'un point pour un angle et une portée de mesure donnés ;  
 $\sigma_{synth}$  : écart-type utilisé pour paramétrer BlenSor et bruitez des données de synthèse ;  
 $\epsilon_{loc}$  : bruit de localisation utilisé pour bruitez les mesures du système de localisation ;  
 $\sigma_{loc}$  : écart-type utilisé pour bruitez les données de localisation ;  
 $\mathcal{D}_i$  : jeux de données acquis au cours de ce travail ;  
 $\nu$  : vitesse de roulage lors des acquisitions ;  
*R* : solution de perception de référence ;  
*T* : solution de perception testée ;  
*P<sub>max</sub>* : portée maximale pour laquelle les polynômes associés aux lignes détectées sont appliquées ;  
*P<sub>test</sub>* : portées maximales utilisées au cours des expériences pour limiter  $P_{max}$ .

## Structures octrees

- $N$  : nombre de sous-espaces dans lesquels un espace est divisé pour une structure octree ou quadtree ;
- $d$  : niveau de profondeur (depth) dans l'arborescence octree ;
- $d_{max}$  : profondeur maximale associée à une structure octree ;
- $r_d$  : résolution de l'octree associé au niveau de profondeur  $d$  ;
- $r_{max}$  : résolution maximale d'un octree ;
- $V$  : voxel commun / voxel parent ;
- $V_d$  : voxel situé au niveau de profondeur  $d$  de l'arborescence octree ;
- $v$  : voxel enfant ;
- $t_{V_d}$  : taille d'un voxel en fonction de sa profondeur  $d$  dans l'octree ;
- $key$  : identifiant permettant de localiser un voxel dans une structure octree ;
- $value$  : valeur numérique stockée dans le noeud d'un voxel et correspondant généralement à la probabilité d'occupation de ce voxel ;
- $V_o$  : voxel appartenant à l'octree ou la carte octree  $o$  ;
- $o$  : octree commun ;
- $o_{réel}$  : octree créé à partir d'un scan  $scan_{réel}$ , issu de mesures acquises en conditions réelles ;
- $o_{synth}$  : octree créé à partir d'un des scans de synthèse  $scan_{synth}$  générés par simulation ;
- $O_{synth}$  : carte octree obtenue par intégration temporelle et regroupant plusieurs octrees  $o_{synth}$  basés sur des données de synthèse ;
- $O_{ref}$  : carte octree de référence, obtenue par intégration temporelle des données acquises en conditions réelles, puis référencées par la méthode introduite au cours de ce travail ;
- $O_\epsilon$  : octree basé sur les données bruitées ;
- $o_{test}$  : octree basé sur les données produites par les solutions de perception testées ;
- $\mathbb{P}$  : critère de pertinence s'appliquant sur la classe sémantique associée ( $label$ ) à un voxel ;
- $\mathcal{P}_{max}$  : valeur maximale de probabilité d'occupation trouvée parmi les 8 voxels enfants  $v_{i=\{1 \text{ à } 8\}}$  d'un voxel parent  $V$ .

## Mesures de localisation et référentiels

*Pose* : couple de position et attitude permettant ;

*roll* : angle de roulis ;

*pitch* : angle de tangage ;

*yaw* : angle de lacet ;

*Ego* : référentiel local dont l'origine et l'orientation sont définis par rapport à la plateforme ;

$\mathcal{L}$  : référentiel local propre à un capteur LiDAR donné ;

$\mathcal{C}$  : référentiel local propre à une caméra donnée ;

$\mathcal{G}$  : système de coordonnées de référence connu ;

$\lambda$  : coordonnée de longitude ;

$\varphi$  : coordonnées de latitude.

## Méthode de référencement et critères d'association

*EE* : environnement d'essai ;

*CRA* : pour Conditions Réelles d'Acquisition, correspond au premier domaine utilisé dans le cadre de l'adaptation de domaine et correspondant aux mesures acquises en conditions réelles ;

*MAP* : correspond au second domaine, représenté par une carte sémantique 3D utilisée comme source d'information sémantique lors de la génération de données de synthèse ;

$N_{i=\{0 \grave{a} 4\}}$  : type de voisinage pris en compte et servant de critère lors de l'association des données basée sur la connexité ;

$D_s$  : distance seuil dans l'application d'un critère d'association basé sur la distance euclidienne ;

$D_M$  : distance de Mahalanobis.

## Métriques d'évaluation

*CA* : métrique de précision de classe ;

$N_{pred}$  : nombre de voxels prédits correctement par une solution testée ;

$N_{ref}$  : nombre de voxels de référence ;

*IoU* : indice de Jaccard ;

*REC*, *PRE* : métriques correspondant à la précision et au rappel ;

$F_{score}$  : moyenne harmonique de la précision et du rappel.

# Table des matières

|  |            |
|--|------------|
| <b>Remerciements</b>   | <b>i</b>   |
| <b>Résumé</b>  | <b>iii</b> |
| <b>Abstract</b>  | <b>iv</b>  |
| <b>Symboles et notations</b>                                       | <b>v</b>   |
| <b>1 Introduction générale</b>                                     | <b>1</b>   |
| 1.1 Contexte général . . . . .                                     | 3          |
| 1.2 Problématique liée à la perception . . . . .                   | 4          |
| 1.3 Objectifs de la thèse . . . . .                                | 5          |
| 1.4 Organisation du manuscrit . . . . .                            | 5          |
| <b>2 Perception des véhicules</b>                                  | <b>7</b>   |
| 2.1 Introduction . . . . .   | 7          |
| 2.2 Capteurs extéroceptifs . . . . .                               | 8          |
| 2.2.1 Capteurs passifs . . . . .                                   | 9          |
| 2.2.2 Capteurs actifs . . . . .                                    | 13         |
| 2.2.3 Synthèse comparative des capteurs . . . . .                  | 21         |
| 2.2.4 Moyens de représentation des données de perception . . . . . | 23         |
| 2.3 Traitement des données de perception . . . . .                 | 26         |
| 2.3.1 Géoréférencement des données de perception . . . . .         | 27         |
| 2.3.2 Algorithmes d'interprétation de l'environnement . . . . .    | 33         |
| 2.3.3 Fusion de données . . . . .                                  | 38         |
| 2.4 Validation de la perception . . . . .                          | 38         |
| 2.5 Conclusion . . . . .   | 40         |
| <b>3 Conception de la méthode de référencement</b>                 | <b>41</b>  |
| 3.1 Introduction . . . . .   | 41         |
| 3.2 Méthodologies de référencement . . . . .                       | 42         |
| 3.2.1 Présentation des approches existantes . . . . .              | 43         |
| 3.2.2 Synthèse comparative . . . . .                               | 47         |
| 3.3 Concept méthodologique proposé . . . . .                       | 48         |

|          |  |            |
|----------|--|------------|
| 3.3.1    | Analyse des forces méthodologiques . . . . .                     | 48         |
| 3.3.2    | Idée générale proposée . . . . .                                 | 50         |
| 3.3.3    | Détails du concept . . . . .                                     | 52         |
| 3.4      | Travaux préparatoires . . . . .                                  | 55         |
| 3.4.1    | Carte sémantique 3D de l'environnement . . . . .                 | 55         |
| 3.4.2    | Spécification des systèmes . . . . .                             | 60         |
| 3.4.3    | Méthode de représentation des données . . . . .                  | 63         |
| 3.5      | Synthèse et discussions . . . . .                                | 64         |
| 3.5.1    | Positionnement de l'approche . . . . .                           | 65         |
| 3.5.2    | Application de la méthodologie . . . . .                         | 66         |
| 3.5.3    | Validation méthodologique . . . . .                              | 67         |
| 3.6      | Conclusion . . . . .   | 69         |
| <b>4</b> | <b>Référencement automatique des données</b>                     | <b>71</b>  |
| 4.1      | Introduction . . . . .   | 71         |
| 4.2      | Prédiction des annotations . . . . .                             | 72         |
| 4.2.1    | Méthode et outil de simulation . . . . .                         | 73         |
| 4.2.2    | Modélisation du capteur de référence . . . . .                   | 74         |
| 4.2.3    | Utilisation des sources de données . . . . .                     | 75         |
| 4.3      | Transfert d'informations sémantiques . . . . .                   | 78         |
| 4.3.1    | Méthodes d'association de données . . . . .                      | 78         |
| 4.3.2    | Caractéristiques de la méthode choisie . . . . .                 | 85         |
| 4.4      | Validation de la méthodologie de référencement . . . . .         | 94         |
| 4.4.1    | Analyse de la méthode proposée . . . . .                         | 94         |
| 4.4.2    | Analyse du modèle paramétrique du capteur de référence . . . . . | 95         |
| 4.4.3    | Analyse du référencement . . . . .                               | 100        |
| 4.5      | Perspectives . . . . .   | 109        |
| 4.5.1    | Critères d'association . . . . .                                 | 109        |
| 4.5.2    | Utilisation du géoréférencement . . . . .                        | 110        |
| 4.6      | Conclusion . . . . .   | 111        |
| <b>5</b> | <b>Évaluation de solutions de perception</b>                     | <b>113</b> |
| 5.1      | Introduction . . . . .   | 113        |
| 5.2      | Processus d'évaluation . . . . .                                 | 114        |
| 5.2.1    | Application de la solution testée . . . . .                      | 114        |
| 5.2.2    | Projection des données . . . . .                                 | 114        |
| 5.2.3    | Évaluation de la solution testée . . . . .                       | 115        |
| 5.2.4    | Métriques d'évaluation quantitative . . . . .                    | 116        |
| 5.2.5    | Évaluation qualitative . . . . .                                 | 117        |
| 5.3      | Évaluation de solutions de perception . . . . .                  | 118        |
| 5.3.1    | Solutions de perception évaluées . . . . .                       | 118        |

|          |  |            |
|----------|--|------------|
| 5.3.2    | Évaluation des solutions de perception basées LiDAR . . . . .    | 121        |
| 5.3.3    | Évaluation de la solution de perception basée Caméra . . . . .   | 127        |
| 5.4      | Perspectives . . . . .   | 136        |
| 5.4.1    | Évaluation de solutions connues . . . . .                        | 136        |
| 5.4.2    | Prise en compte des variations temporelles . . . . .             | 136        |
| 5.4.3    | Diversification des systèmes évalués . . . . .                   | 136        |
| 5.5      | Conclusion . . . . .   | 137        |
| <b>6</b> | <b>Conclusion générale</b>                                       | <b>139</b> |
| 6.1      | Synthèse des travaux réalisés à l'issue de cette thèse . . . . . | 139        |
| 6.1.1    | Conception d'une méthode de référencement . . . . .              | 140        |
| 6.1.2    | Évaluation de solutions de perception . . . . .                  | 142        |
| 6.2      | Perspectives . . . . .   | 143        |
| 6.2.1    | Évolution du référencement . . . . .                             | 143        |
| 6.2.2    | Déploiement de l'approche . . . . .                              | 144        |
| 6.3      | Publications scientifiques . . . . .                             | 144        |
| <b>A</b> | <b>Documentation de la centrale Ixblue Atlans-C</b>              | <b>145</b> |
| <b>B</b> | <b>Transformation rigide</b>                                     | <b>148</b> |
| <b>C</b> | <b>Documentation Riegl</b>                                       | <b>150</b> |
| <b>D</b> | <b>Description du « wall test »</b>                              | <b>152</b> |



# Table des figures

|      |   |    |
|------|---|----|
| 1.1  | Démonstrateurs historiques de la conduite autonome . . . . .                        | 2  |
| 1.2  | Nouvelles fonctions intégrées aux véhicules modernes . . . . .                      | 2  |
| 1.3  | Fonctionnalités majeures des ADAS . . . . .   | 3  |
| 1.4  | Capteurs de perception . . . . .  | 4  |
| 2.1  | Exemple de données de perception traitées . . . . .                                 | 8  |
| 2.2  | Organisation de la section dédiée aux capteurs extéroceptifs . . . . .              | 8  |
| 2.3  | Exemples de caméras embarquées sur véhicule . . . . .                               | 9  |
| 2.4  | Modélisation d'une caméra par le modèle sténopé . . . . .                           | 10 |
| 2.5  | Types de caméras . . . . .  | 11 |
| 2.6  | Limites de la perception basée vision . . . . .                                     | 11 |
| 2.7  | Modèles de caméras stéréoscopiques . . . . .  | 12 |
| 2.8  | Principe de la stéréo-vision . . . . .  | 13 |
| 2.9  | Principe de mesure basé sur le temps de vol . . . . .                               | 14 |
| 2.10 | Principe de mesure basé sur la différence de phase . . . . .                        | 14 |
| 2.11 | Modèles de LiDARs . . . . .   | 15 |
| 2.12 | Types de LiDARs à balayage . . . . .  | 16 |
| 2.13 | Modélisation du capteur LiDAR . . . . .   | 17 |
| 2.14 | Scan LiDAR . . . . .  | 17 |
| 2.15 | Exemple de données type mesurées par un capteur LiDAR . . . . .                     | 18 |
| 2.16 | Représentation de l'impact de la pluie sur des données LiDAR . . . . .              | 18 |
| 2.17 | Exemple de radars exploités pour la perception . . . . .                            | 19 |
| 2.18 | champs de vue des différents types de radar . . . . .                               | 20 |
| 2.19 | Exemple de capteurs à ultrasons . . . . .   | 20 |
| 2.20 | Exemples de LiDARs flashs et de caméras 3D . . . . .                                | 21 |
| 2.21 | Champs de vue et portée des différents types de capteurs . . . . .                  | 22 |
| 2.22 | Analyse comparative des différents types de capteurs . . . . .                      | 23 |
| 2.23 | Structure d'un octree . . . . .   | 25 |
| 2.24 | Représentation de différents niveaux de profondeur d'une structure octree . . . . . | 26 |
| 2.25 | Organisation de la section dédiée aux traitements des données . . . . .             | 27 |
| 2.26 | Systèmes types pour la localisation . . . . .                                       | 27 |
| 2.27 | Pose d'objet ou de capteur . . . . .  | 29 |

|      |  |    |
|------|--|----|
| 2.28 | Représentation des différents référentiels généralement exploités dans le cadre de la perception.                        | 32 |
| 2.29 | Représentation d'une transformation rigide . . . . .   | 32 |
| 2.30 | Représentation des différentes transformations utilisées . . . . .   | 33 |
| 2.31 | Exemples de résultats d'algorithmes de perception . . . . .  | 33 |
| 2.32 | Organisation de la section dédiée aux algorithmes de traitement . . . . .  | 34 |
| 2.33 | Représentation de la distinction entre le sol et les objets . . . . .  | 34 |
| 2.34 | Illustration des résultats d'algorithmes de détection du sol . . . . .   | 35 |
| 2.35 | Exemples d'algorithmes de détection d'objets . . . . .   | 36 |
| 2.36 | Représentation des étapes de traitements jusqu'à la classification des données. . . . .                                  | 36 |
| 2.37 | Exemples de résultats obtenus à partir d'algorithmes de segmentation sémantique . . . . .                                | 37 |
|      |  |    |
| 3.1  | Approches de validation étudiées . . . . .   | 42 |
| 3.2  | Processus d'évaluation avec méthodologie de référencement des données supervisée . . . . .                               | 43 |
| 3.3  | Processus d'évaluation avec méthodologie de référencement des données semi-supervisée . . . . .                          | 44 |
| 3.4  | Processus d'évaluation avec méthodologie de référencement des données non-supervisée . . . . .                           | 45 |
| 3.5  | Processus d'évaluation avec données référencées par la simulation . . . . .  | 46 |
| 3.6  | Points forts associés aux différents référencements étudiés . . . . .  | 49 |
| 3.7  | Approche de validation des solutions de perception proposée . . . . .  | 50 |
| 3.8  | Exemple d'application de l'adaptation de domaine . . . . .   | 52 |
| 3.9  | Positionnement analogue entre capteur dans le monde réel et capteur modélisé . . . . .                                   | 53 |
| 3.10 | Représentation des domaines et types de données utilisés par la méthode de référencement . . . . .                       | 54 |
| 3.11 | Facettes composant un modèle 3D . . . . .  | 56 |
| 3.12 | Cartographie mobile . . . . .  | 56 |
| 3.13 | Représentation de différents éléments maillés . . . . .  | 58 |
| 3.14 | Hiérarchie des classes sémantiques . . . . .   | 59 |
| 3.15 | Transformation rigide entre le référentiel local d'un capteur LiDAR $\mathcal{L}$ et le référentiel <i>Ego</i> . . . . . | 61 |
| 3.16 | Représentation du relevé . . . . .   | 62 |
| 3.17 | Trimble SX10 . . . . .   | 62 |
| 3.18 | Distinction entre référencement et évaluation . . . . .  | 65 |
| 3.19 | MELO . . . . .   | 67 |
|      |  |    |
| 4.1  | Sous-étapes du référencement automatique . . . . .   | 72 |
| 4.2  | Sous-étape de la prédiction des annotations . . . . .  | 73 |
| 4.3  | Logo de l'outil de simulation BlenSor . . . . .  | 73 |
| 4.4  | Scan de synthèse . . . . .   | 74 |
| 4.5  | Positionnement et orientation du capteur synthétisé dans l'environnement simulé . . . . .                                | 76 |
| 4.6  | Comparaison entre un scan réel et un scan de synthèse . . . . .  | 77 |
| 4.7  | Représentation d'un scan de synthèse généré via BlenSor . . . . .  | 77 |
| 4.8  | Sous-étape « Transfert d'informations sémantiques » . . . . .  | 78 |
| 4.9  | Composition d'une méthode d'association . . . . .  | 79 |
| 4.10 | Catégories de critères d'association . . . . .   | 80 |

|      |  |     |
|------|--|-----|
| 4.11 | Association par correspondance directe entre les données . . . . .                                 | 81  |
| 4.12 | Association de données basée sur la distance euclidienne . . . . .                                 | 81  |
| 4.13 | Association de données par les $k$ plus proches voisins . . . . .                                  | 82  |
| 4.14 | Impact de la variation de densité pour l'association basée sur les plus proches voisins . . . . .  | 83  |
| 4.15 | Connexité des données . . . . .  | 84  |
| 4.16 | Association de données en se basant sur la connexité . . . . .                                     | 84  |
| 4.17 | Représentation de la structuration des données . . . . .   | 86  |
| 4.18 | Représentation de la structure des voxels modifiée . . . . .                                       | 87  |
| 4.19 | Répartition des points en fonction des voxels . . . . .  | 87  |
| 4.20 | Représentation des octrees annotés sémantiquement . . . . .  | 89  |
| 4.21 | Représentation des données à associer . . . . .  | 91  |
| 4.22 | Application de la méthode d'association définie . . . . .  | 92  |
| 4.23 | Représentation du transfert d'information sémantique . . . . .                                     | 94  |
| 4.24 | Données réelles référencées automatiquement . . . . .  | 94  |
| 4.25 | Analogie entre un cône et la forme produite par un faisceau laser lors d'une rotation . . . . .    | 96  |
| 4.26 | Estimation de la résolution azimutale . . . . .  | 97  |
| 4.27 | Représentation du couloir d'étude . . . . .  | 101 |
| 4.28 | Représentation du filtrage appliqué . . . . .  | 101 |
| 4.29 | Représentation des erreurs de localisation . . . . .   | 105 |
| 4.30 | Approche modifiée pour les expériences . . . . .   | 106 |
| 4.31 | Représentation graphique de l'impact du bruit de localisation sur le référencement . . . . .       | 107 |
| 5.1  | Approche de validation des solutions de perception proposée . . . . .                              | 114 |
| 5.2  | Sous-étapes de la « Projection des données » . . . . .   | 115 |
| 5.3  | Représentation des quatre cas de matrice de confusion lors des comparaisons entre voxels . . . . . | 116 |
| 5.4  | Évaluation qualitative d'une solution $T$ . . . . .  | 117 |
| 5.5  | Configuration sensorielle de la plateforme MELO . . . . .  | 118 |
| 5.6  | Trajectoire suivie par la plateforme . . . . .   | 119 |
| 5.7  | Données obtenues dans le cadre de l'extraction du sol . . . . .                                    | 119 |
| 5.8  | Calcul des points associés aux lignes détectées . . . . .  | 120 |
| 5.9  | Valeurs de $P_{max}$ au cours du jeu de données . . . . .  | 121 |
| 5.10 | Évaluation des performances des solutions d'extraction du sol . . . . .                            | 122 |
| 5.11 | Courbes du $F_{score}$ au cours de la séquence étudiée . . . . .                                   | 123 |
| 5.12 | $F_{score}$ en fonction de la trajectoire d'acquisition . . . . .                                  | 124 |
| 5.13 | Évolution du $F_{score}$ et du roulis au cours de la séquence étudiée . . . . .                    | 125 |
| 5.14 | Évolution du $F_{score}$ et du tangage au cours de la séquence étudiée . . . . .                   | 125 |
| 5.15 | Impact du roulis en fonction de la position du capteur . . . . .                                   | 126 |
| 5.16 | Impact du tangage en fonction de la position du capteur . . . . .                                  | 126 |
| 5.17 | Évaluation des performances de la solution de perception $T_3$ . . . . .                           | 128 |
| 5.18 | Représentation du paramètre $P_{max}$ en fonction de la trajectoire suivie par la caméra . . . . . | 129 |
| 5.20 | Représentation de la métrique CA en fonction de la trajectoire suivie par la plateforme . . . . .  | 130 |

|      |   |     |
|------|---|-----|
| 5.19 | Courbes de $CA$ en fonction des paramétrages de portée maximale . . . . .                     | 131 |
| 5.21 | Superposition des courbes de $CA$ pour deux résolutions d'octree . . . . .                    | 133 |
| 5.22 | Représentation de $CA$ par histogrammes de fréquence pour les différents paramétrages étudiés | 134 |
| B.1  | Représentation d'une transformation rigide . . . . .  | 148 |
| D.1  | Représentation du « wall test » . . . . .   | 152 |

# Chapitre 1

## Introduction générale

### Sommaire

---

|     |  |   |
|-----|--|---|
| 1.1 | Contexte général . . . . .                   | 3 |
| 1.2 | Problématique liée à la perception . . . . . | 4 |
| 1.3 | Objectifs de la thèse . . . . .              | 5 |
| 1.4 | Organisation du manuscrit . . . . .          | 5 |

---

Devenu l'un des défis technologiques majeurs de la dernière décennie, le véhicule autonome est pourtant un sujet évoqué depuis les années 1920. Les progrès technologiques réalisés depuis ont permis de présenter divers prototypes de voitures autonomes au cours du temps (Fig 1.1). On peut citer par exemple, les premiers véhicules sans conducteurs « Navlab » [1], développés depuis les années 1980 par des universitaires, dans un but initialement militaire. En 1994, c'est le constructeur automobile Mercedes-Benz qui réalise une démonstration avec deux prototypes « VaMP » et « VITA-2 », parcourant 1 000 kilomètres en conduite autonome sur une autoroute multivoie parisienne, en conditions de trafic réelles, et à une vitesse atteignant 130 km/h [2].

Ces différentes démonstrations ont permis de faire croître l'intérêt scientifique, comme médiatique, autour du véhicule autonome. Actuellement, les différents sujets de recherches gravitant autour du véhicule autonome occupent de nombreux laboratoires universitaires. On peut le voir avec des challenges tels que le DARPA de l'université de Stanford [3], avec des travaux tels que ceux de l'institut de technologie de Karlsruhe avec le jeu de données populaire KITTI [4] ou encore, avec les différents intervenants auprès du Groupe de Recherche (GdR) en robotique du CNRS [5]. Une grande partie du secteur industriel automobile est aussi engagé dans ce défi technologique tels que les constructeurs ou les équipementiers (Valeo, ZF, Aptiv,...). On y retrouve également des acteurs extérieurs au monde automobile tels des entreprises intervenant dans divers domaines technologiques (Google, Apple, Uber...), mais aussi de nombreuses start-ups créées uniquement autour de ce domaine de recherche (Navya, EasyMile, Zoox...). Bien que des aspects techniques et législatifs actuels limitent les constructeurs dans la commercialisation de véhicules totalement autonomes, de nombreuses démonstrations sont présentées chaque année. Les développements réalisés permettent néanmoins de produire des systèmes de conduite semi-autonome, ou d'aides à la conduite, regroupés sous le terme ADAS<sup>1</sup> et qui équipent de plus en plus de véhicules accessibles au grand public. Au cours du temps, les véhicules n'ont cessé d'évoluer et d'intégrer de nouvelles fonctionnalités. Les véhicules sont ainsi devenus des systèmes de plus en plus complexes (Fig 1.2). Avec l'arrivée des ADAS, une

---

1. ADAS : Advanced Driver Assistance Systems

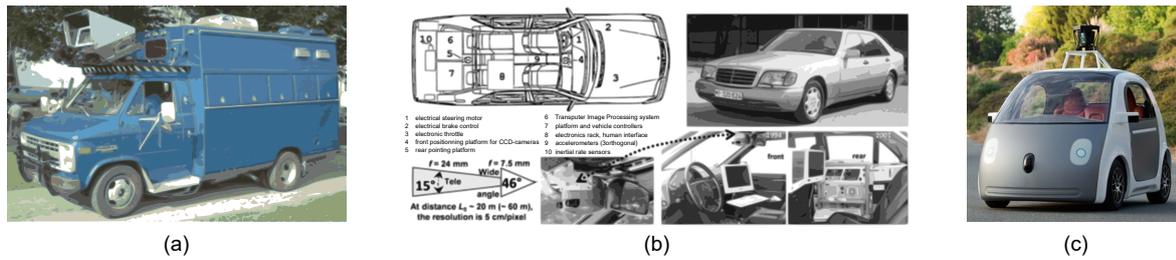


FIGURE 1.1 – Démonstrateurs historiques de la conduite autonome : Exemples de véhicules autonomes de démonstration présentés au cours du temps : (a) Navlab1 présenté en 1986 ; (b) VaMP présenté en 1994 ; (c) Firefly de Google utilisé entre 2015 et 2017.

nouvelle couche s'intègre à cet écosystème technologique et renforce la complexification des véhicules et des systèmes qu'ils embarquent.

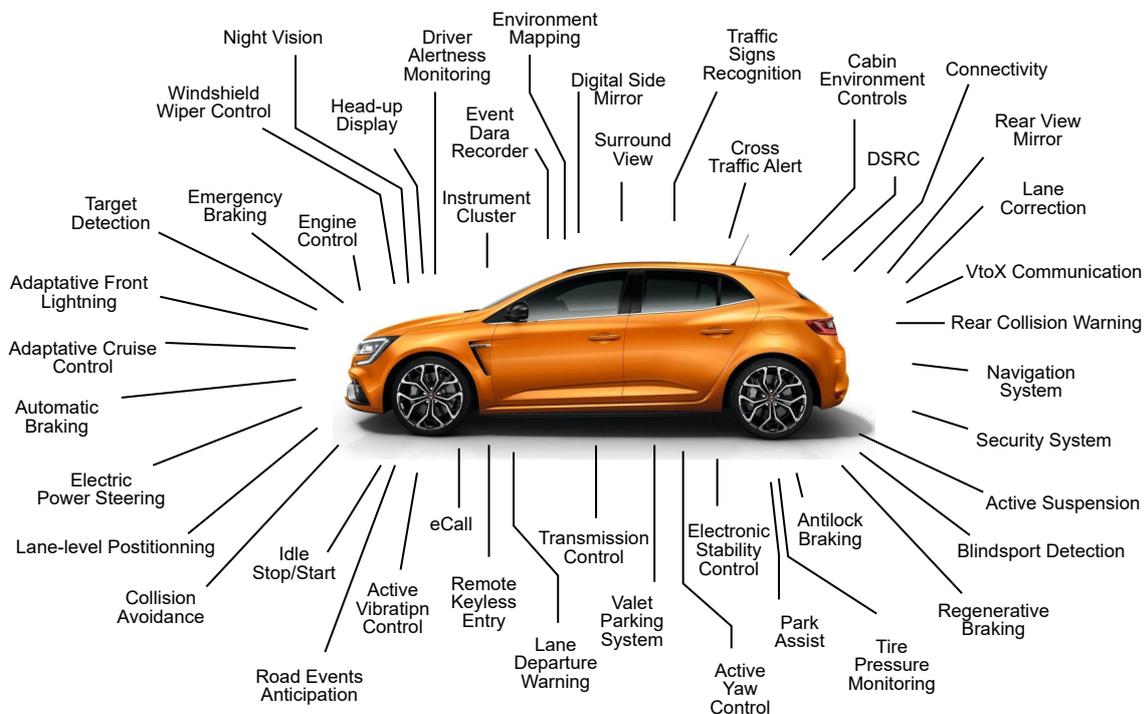


FIGURE 1.2 – Nouvelles fonctions intégrées aux véhicules modernes : Exemples de fonctions actuellement disponibles et rencontrées sur des véhicules de série.

Pour fonctionner, les ADAS s'appuient sur trois fonctionnalités majeures que sont : la planification, le contrôle et la perception, cette dernière étant au cœur de ce sujet de thèse (Fig 1.3). En exploitant différents composants qui équipent le véhicule (capteurs et/ou logiciels), ces fonctionnalités majeures interviennent conjointement pour permettre au véhicule de réaliser des actions de façon autonome. Ces composants mesurent et traitent les données pour transmettre les informations nécessaires aux fonctionnalités majeures. Ces informations permettent de répondre aux différentes tâches qui leur sont assignées. Cela permet alors de contrôler le véhicule et d'automatiser sa circulation. Cette évolution des véhicules, auparavant purement mécaniques, amène à de nouvelles problématiques rencontrées par le secteur automobile et touchant des domaines technologiques jusqu'alors inexplorés dans ce milieu industriel.

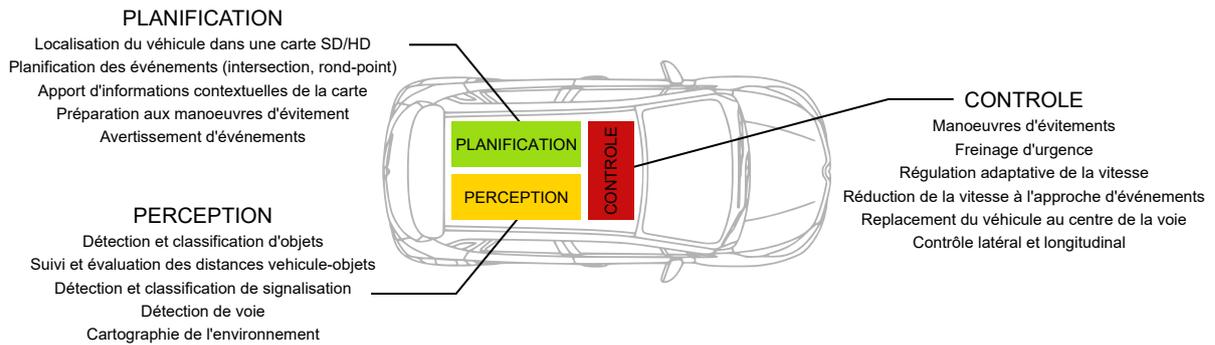


FIGURE 1.3 – Fonctionnalités majeures des ADAS : Composition et répartition des tâches en fonction des différentes fonctionnalités majeures des ADAS.

## 1.1 Contexte général

En permettant au véhicule de percevoir l'environnement dans lequel il évolue, la perception détient un rôle clé parmi ces fonctionnalités majeures des systèmes ADAS (Fig 1.3). Elle est exploitée pour différentes applications fondamentales à ces systèmes.

On y retrouve la détection des cibles fixes (panneaux de signalisation, glissières de sécurité, trottoirs...), ou mobiles (véhicules, piétons, cyclistes...). Les cibles mobiles peuvent faire l'objet de suivi (tracking) afin de prédire l'évolution de leurs déplacements vis-à-vis du véhicule. Ces détections permettent de définir si les cibles situées dans l'environnement du véhicule peuvent représenter un obstacle, et si une manœuvre de freinage, ou d'évitement, est nécessaire. La reconnaissance de ces cibles peut aussi être utilisée d'un point de vue contextuel. A titre d'exemple, le système ISA<sup>2</sup> [6] et qui devrait devenir réglementaire au sein de l'union européenne pour les nouveaux véhicules, en 2022. Il permet à un véhicule de réguler sa vitesse par la reconnaissance des panneaux de signalisation, ou en fonction de sa géolocalisation. Ce type d'application permet de démontrer le lien existant entre la perception et le contrôle.

Du point de vue de la planification, le lien avec la perception peut être fait via d'autres applications telles que la construction d'une carte locale basée sur les données de perception. Celle-ci peut être exploitée pour permettre au véhicule de se localiser vis-à-vis de l'environnement dans lequel il évolue. Elle peut également intervenir dans un positionnement global du véhicule en utilisant conjointement les données de perception et les données utilisées par le système de localisation. En faisant correspondre l'environnement perçu à celui d'une carte haute définition par le biais d'éléments communs, la perception peut permettre d'estimer le positionnement du véhicule dans cette carte. En connaissant précisément la position du véhicule au sein d'une carte, il est alors possible d'anticiper des événements futurs (intersections, rond-points, ...) de façon plus précise. Il est ensuite possible de mieux planifier les actions à réaliser sur le comportement du véhicule.

La perception joue donc un rôle primordial et inhérent aux autres fonctionnalités majeures, puisqu'elle va leur transmettre les informations nécessaires pour la bonne exécution de leurs tâches.

Pour être capable de percevoir son environnement, un véhicule doit être équipé de capteurs de perception pouvant être de plusieurs types (LiDARs, caméras, radars, capteurs à ultrasons...) et pouvant produire différents types de données illustrées dans (Fig 1.4). Un système de perception peut être décomposé en deux composantes. La première correspond aux capteurs de perception. Ils réalisent des mesures à partir desquelles le véhicule récupère les informations brutes, nécessaires pour interpréter son environnement.

2. Intelligent Speed Adaptation

La seconde est la partie algorithmique qui correspond aux algorithmes de traitement de données. Les données brutes acquises par un capteur sont analysées par ces algorithmes pour être interprétées. Cela permet d'extraire des informations compréhensibles et de les transmettre aux autres fonctionnalités. (Exemple : lorsque le système de perception détecte un obstacle sur la trajectoire du véhicule, la fonctionnalité de perception transmettra alors les informations nécessaires au contrôle pour permettre l'évitement de cet obstacle).

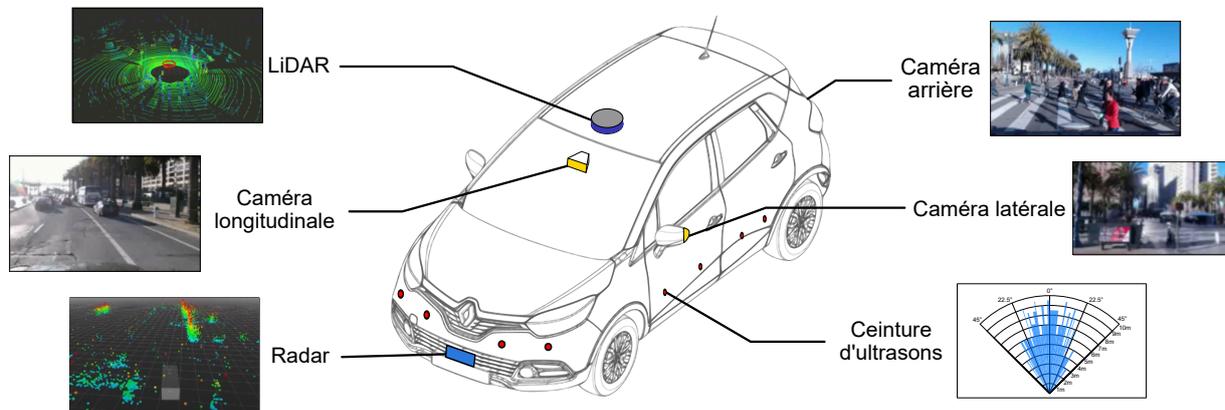


FIGURE 1.4 – Capteurs de perception : Exemples de capteurs de perception couramment exploités dans le cadre des ADAS ainsi que la représentation des données qu'ils produisent. [7, 8]

## 1.2 Problématique liée à la perception

La conception d'un système de perception, performant et suffisamment robuste pour permettre aux systèmes de prendre des décisions sécuritaires, est une tâche complexe. La diversité de solutions exploitables pour ce rôle, en termes de capteurs ou d'algorithmes de traitement, contribue à complexifier davantage cette tâche. De nos jours, la perception reste un domaine de recherche actif [9, 10, 11]. Capteurs et algorithmes de traitement continuent de connaître des évolutions rapides et conséquentes. Ces évolutions, en plus d'accentuer la difficulté pour une solution de rester compétitive à long terme, peuvent rendre l'évaluation des performances complexe. Ainsi, il devient de plus en plus compliqué de comparer les solutions existantes.

Compte tenu du rôle central de la perception dans le bon fonctionnement des systèmes autonomes et ADAS, il reste néanmoins primordial d'être en mesure d'évaluer et de caractériser les éléments dont elle dépend. Ces évaluations permettent de garantir qu'une solution est suffisamment précise et robuste pour percevoir l'ensemble des éléments primordiaux de son environnement. Cela permet aussi de la positionner vis-à-vis de ce qui est couramment exploité dans ce domaine.

Pour évaluer un système de perception, il est nécessaire de disposer de données de référence auxquelles il est possible de comparer les données produites par le système évalué. Bien que de nombreux efforts aient été faits autour de l'évaluation des systèmes de perception [12, 13, 14, 4], il n'existe pas à ce jour de solution permettant de les évaluer dans divers scénarios, avec une configuration de capteurs variable. L'approche actuellement privilégiée est d'utiliser des outils de benchmark populaires tels que « The KITTI Vision Benchmark Suite » [4] et reposant sur la création de jeux de données référencés manuellement. Cependant, ces jeux de données référencés sont complexes à mettre en œuvre et ne peuvent évoluer dans le temps. Ils restent figés avec une configuration de capteurs définie, ce qui limite le cadre d'évaluation et rend difficile l'évaluation de solutions dans différents scénarios ou différentes configurations.

En prenant en compte ces éléments, on comprend que pour concevoir un système de perception, il est nécessaire de disposer d'un moyen de validation suffisamment flexible pour s'adapter aux différents types de capteurs. Il faut que cet outil soit également le plus autonome possible pour automatiser le référencement des jeux de données, limiter l'intervention humaine qui est consommatrice de temps et faciliter le processus d'évaluation. Cette automatisation du référencement et la flexibilité qui en résulterait permettraient de faciliter les évaluations pour différents scénarios et pour différentes configurations de capteurs. Un moyen de validation tel que décrit, permettrait d'évaluer rapidement les nouvelles solutions et de réduire la complexité de la conception d'un système de perception.

### 1.3 Objectifs de la thèse

L'objectif de ce travail de thèse est donc de répondre à cette problématique en mettant au point une méthodologie de validation innovante, permettant de caractériser différentes solutions de perception exploitables dans le cadre des systèmes ADAS et du véhicule autonome. L'élaboration de cette nouvelle méthodologie est guidée par certaines contraintes.

Dans le milieu industriel, deux alternatives sont possibles : développer ses propres solutions de perception en interne, ou exploiter des solutions COTS<sup>3</sup> proposées par des équipementiers ou des fournisseurs. Le panel de solutions exploitables peut alors être varié et basé sur des principes de fonctionnements différents (solutions basées images avec les caméras, solutions basées laser avec les LiDARs, ...). L'une des premières contraintes à prendre en compte est l'adaptabilité de la méthodologie. Celle-ci doit être capable de prendre en charge les données issues de ces solutions variées et de les évaluer.

Une seconde contrainte est également liée à l'adaptabilité, l'outil doit être applicable sur n'importe quel jeu de données enregistré avec les solutions testées. Cela permettra au constructeur de mettre en place ses propres scénarios de validation et d'y appliquer l'outil développé.

Enfin, la dernière contrainte est de rendre l'outil le plus autonome possible. L'évaluation manuelle réalisée par des opérateurs sur des jeux de données enregistrés représente une tâche complexe, consommatrice en termes de coûts et de temps. Automatiser au maximum le processus de validation de cet outil permettra alors de limiter ces consommations de ressources tout en renforçant l'équité des évaluations, dans la continuité des premières contraintes.

### 1.4 Organisation du manuscrit

Dans le cadre de ces travaux de thèse, nous proposons une méthodologie de traitement destinée à l'évaluation des solutions de perception. Pour présenter ces travaux, le manuscrit est organisé de la façon suivante : au sein du chapitre 2, les connaissances générales sur le domaine des systèmes de perception seront présentées. Ces bases permettront d'étudier les divers capteurs et algorithmes de traitement qui composent un système de perception. Davantage de détails seront apportés sur les éléments qui ont été exploités au cours de cette thèse afin de faciliter la bonne compréhension des travaux réalisés.

---

3. Commercial Off-The-Shelf

Le chapitre 3 portera sur la conception de notre approche. Dans un premier temps, nous présenterons les méthodes employées pour référencer les données. Cette étape est cruciale afin de disposer d'une référence par rapport à laquelle une solution de perception peut être confrontée et évaluée. Cette étude nous permettra d'analyser les besoins toujours existants parmi les méthodes de référencement étudiées et qui nous guideront lors de la conception de notre approche. Dans la suite de ce chapitre, le concept méthodologique final est présenté et détaillé. Les éléments clés à mettre en place en amont de son application seront également discutés.

Après avoir introduit le concept méthodologique permettant de produire des données de référence de façon automatisée au cours du chapitre 3, le chapitre 4 se concentre sur la mise en application de cette méthodologie de référencement. Les éléments méthodologiques et techniques, exploités pour rendre ce référencement automatique applicable, seront étudiés. Dans l'objectif de mesurer l'intégrité des données de référence produites par notre méthodologie, des expériences ont été mises en œuvre concernant les points clés intervenant dans le processus de référencement des données.

Dans le chapitre 5, les critères d'évaluation utilisés ainsi que la façon d'exploiter les données référencées automatiquement seront présentées. Afin de valider notre concept, celui-ci a été déployé sur des solutions exploitant différents types de données ou différents principes de fonctionnement. Dans l'objectif de démontrer la flexibilité de notre approche, ces solutions correspondent à différents niveaux de maîtrise avec : une solution dont l'algorithme est totalement maîtrisé et qui a été implantée au cours de ces travaux, une solution dont l'algorithme est utilisé tel que proposé (déjà implémenté) et enfin, une solution fermée intervenant en boîte noire.

Enfin, le chapitre 6 permettra de conclure de manière synthétique sur les travaux menés, de discuter des résultats obtenus durant les expérimentations et de présenter les perspectives qui permettront de compléter et d'améliorer le concept proposé.

# Chapitre 2

## Perception des véhicules

### Sommaire

---

|       |  |    |
|-------|--|----|
| 2.1   | Introduction . . . . .                                       | 7  |
| 2.2   | Capteurs extéroceptifs . . . . .                             | 8  |
| 2.2.1 | Capteurs passifs . . . . .                                   | 9  |
| 2.2.2 | Capteurs actifs . . . . .                                    | 13 |
| 2.2.3 | Synthèse comparative des capteurs . . . . .                  | 21 |
| 2.2.4 | Moyens de représentation des données de perception . . . . . | 23 |
| 2.3   | Traitement des données de perception . . . . .               | 26 |
| 2.3.1 | Géoréférencement des données de perception . . . . .         | 27 |
| 2.3.2 | Algorithmes d'interprétation de l'environnement . . . . .    | 33 |
| 2.3.3 | Fusion de données . . . . .                                  | 38 |
| 2.4   | Validation de la perception . . . . .                        | 38 |
| 2.5   | Conclusion . . . . .   | 40 |

---

### 2.1 Introduction

Afin de percevoir son environnement, un véhicule peut embarquer des capteurs de différents types. De façon à assurer la redondance entre ceux-ci, ils peuvent être basés sur des principes physiques de mesure différents. A partir de ces capteurs, il est possible de collecter des données issues des mesures réalisées et d'obtenir une représentation 2D, ou 3D, des environs du véhicule. En fonction des capteurs embarqués et de leur configuration, la représentation produite à partir des données sera plus ou moins détaillée et fidèle à la scène observée. Cependant, produire une représentation de l'environnement ne suffit pas à disposer des informations nécessaires pour interpréter la scène et donc opérer des décisions sur la trajectoire ou la vitesse du véhicule. C'est pourquoi, les algorithmes de perception sont cruciaux. Les traitements de données effectués par ces algorithmes permettent d'extraire des informations concrètes des données brutes mesurées par les capteurs. Le but est d'en tirer les informations pertinentes et nécessaires à l'interprétation de l'environnement<sup>1</sup> du véhicule (Fig 2.1). Ces informations transmises aux autres fonctionnalités majeures composant les ADAS leurs permettent d'agir sur le véhicule et de prendre des décisions de manière sécuritaire.

---

1. L'interprétation de l'environnement correspond à l'action d'identifier les éléments et cibles présents dans la scène perçue. Cette étape intègre également l'association de chacun de ces objets à une classe sémantique ou à un état (par ex : obstacle / non-obstacle). On pourra également établir le caractère dynamique ou statique de l'objet.

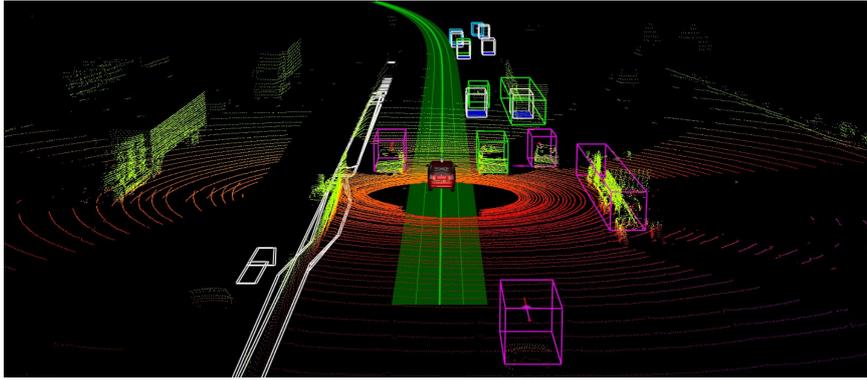


FIGURE 2.1 – Exemple de données de perception traitées : Données LiDAR traitées avec les objets détectés représentés par des boîtes englobantes 3D et détection de la voie circulaire représentée en vert [15].

Au cours de ce chapitre, nous proposons un état de l'art ciblé sur la perception, avec les moyens techniques et algorithmiques existant, permettant l'interprétation de l'environnement d'un véhicule. Un aperçu global des capteurs et approches algorithmiques pertinents vis-à-vis de la problématique étudiée sera présenté. Dans un premier temps, les différents capteurs, leurs caractéristiques, ainsi que leurs principes de fonctionnement seront introduits. Une étude comparative de ces différents capteurs sera présentée. Elle permettra de synthétiser les forces et faiblesses de ces sources de données. Dans un second temps, nous nous intéresserons aux algorithmes de perception et présenterons les algorithmes ciblés lors de la réalisation de ces travaux, ainsi que les données qu'ils produisent en sortie et sur lesquelles les évaluations peuvent être menées. Enfin, nous aborderons les raisons qui motivent l'évaluation des solutions de perception.

**Remarque :** dans cet état de l'art de la perception, d'autres domaines d'exploitation que les ADAS seront évoqués. En effet, la perception embarquée sur des véhicules terrestres est également exploitée pour la robotique et la cartographie mobile. Les configurations des véhicules destinés à ces différents domaines sont très proches, notamment concernant la localisation et la perception.

## 2.2 Capteurs extéroceptifs

Comme nous avons pu l'indiquer précédemment, les capteurs peuvent être basés sur différents principes de fonctionnement. Les capteurs exploités dans le cadre de la perception peuvent être regroupés en deux grandes catégories. La première catégorie correspond aux capteurs passifs qui se limitent à capter l'énergie d'une source extérieure pour réaliser leurs mesures. La seconde catégorie représente les capteurs actifs qui utilisent leur propre source émettrice pour produire le signal servant à mesurer.

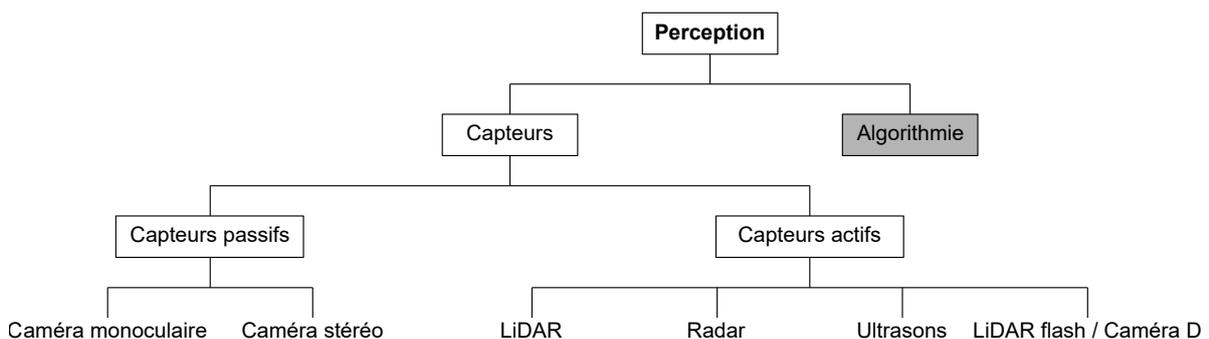


FIGURE 2.2 – Organisation de la section dédiée aux capteurs extéroceptifs.

### 2.2.1 Capteurs passifs

Par définition, les capteurs passifs mesurent un phénomène physique extérieur de leur environnement et sans émettre de signal ou d'onde. Dans le cadre de la perception embarquée sur véhicule, ce type de capteur est majoritairement représenté par les caméras. Le phénomène physique mesuré est la lumière, émise ou réfléchi par les éléments présents dans l'environnement. Elle est captée par les cellules photo-électriques : les photosites, qui composent les capteurs photographiques (CMOS ou CDD). On distingue deux classes de caméras majoritairement exploitées dans la littérature : les caméras monoculaires et les caméras stéréoscopiques.

#### 2.2.1.1 Caméras monoculaires

**Technologie.** Les capteurs photographiques des caméras sont recouverts par un ensemble de photosites, structurés sous la forme d'une matrice (Fig 2.5). La matrice de photosites correspond au format de l'image produite par le capteur. Chaque photosite correspond à un pixel spécifique de l'image et permet de mesurer un rayonnement électromagnétique pour le convertir en signal électrique. Avant d'attribuer une valeur numérique à un pixel, un filtre recouvre le photosite et permet de définir le domaine spectral dans lequel le signal électrique est converti. La valeur obtenue par cette conversion est ensuite stockée au sein du pixel.

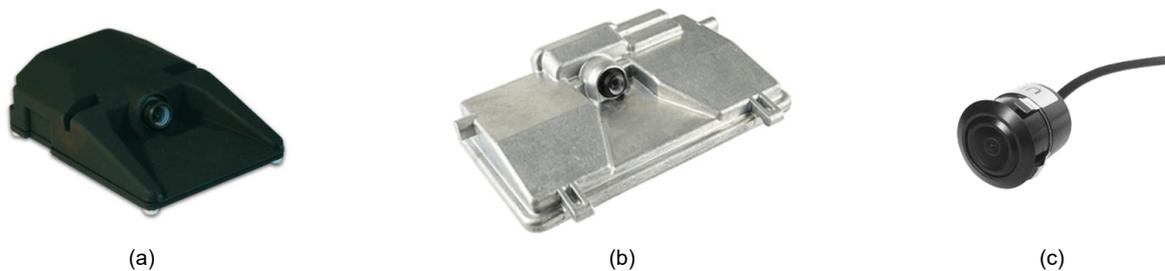


FIGURE 2.3 – Exemples de caméras embarquées : (a) Camera frontale MobilEye ; (b) Camera frontale Bosch ; (c) Exemple de caméra intégrée à l'arrière d'un véhicule.

Deux types de filtres sont majoritairement représentés : le premier est le filtre coloré (exemple : le filtre Bayer ou Foveon) sensible à l'ensemble du spectre de la lumière visible et permettant d'obtenir des images contenant des informations de couleurs sur la scène perçue par la caméra ; le second est le filtre infrarouge, sensible au domaine du même nom. Les images infrarouges permettent de distinguer les différentes natures de surfaces et sources de chaleurs qui composent la scène. Ces images produites sont ensuite interprétées indépendamment les unes des autres, ou en séquence, via des algorithmes de traitements. Cela permet d'en tirer les informations pertinentes sur l'environnement du véhicule.

**Modélisation du capteur.** Les caméras embarquées exploitées peuvent généralement être modélisées géométriquement sur la base du modèle sténopé, correspondant à une projection perspective [16]. Cette projection, représentée (Fig 2.4), décrit la relation entre un point 3D  $M$  de la scène et sa projection  $m$  dans le plan image  $P$ .  $f$  représente la distance focale séparant le centre optique du point principal  $pp$  : son projeté sur  $P$  selon l'axe optique.

$M$  a pour coordonnées  $(X_C, Y_C, Z_C)$  exprimées dans le repère de la caméra  $C$ . Sa projection  $m$  dans le plan image a pour coordonnées  $(x, y)$ .  $Z_C$  représentant la distance entre  $M$  et le centre optique selon l'axe optique et  $f$  la distance séparant le centre optique du plan image selon ce même axe, on obtient les relations suivantes (Eq 2.1)

$$\frac{f}{Z_C} = \frac{x}{X_C} = \frac{y}{Y_C} \quad (2.1)$$

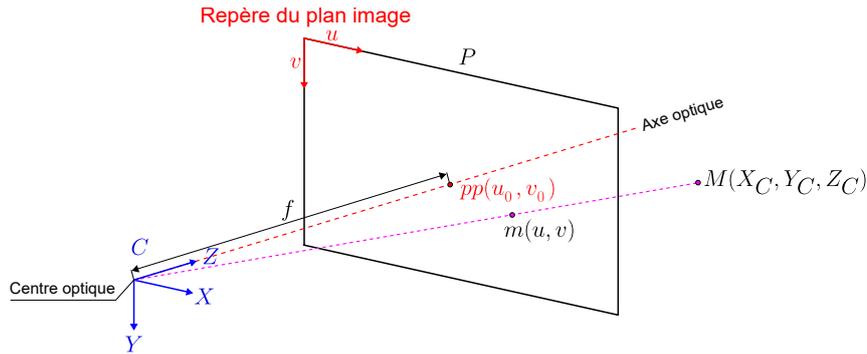


FIGURE 2.4 – Modélisation d'une caméra par le modèle sténopé.

On en déduit les coordonnées de  $m$  par la relation (Eq 2.2).

$$x = f \cdot \frac{X_C}{Z_C}, y = f \cdot \frac{Y_C}{Z_C} \quad (2.2)$$

La définition des paramètres de calibration intrinsèques  $f_x$ ,  $f_y$ ,  $u_0$  et  $v_0$ , représentant respectivement les distances focales selon les directions  $x$  et  $y$ , ainsi que les coordonnées du point principal  $pp(u_0, v_0)$ , permet le passage en coordonnées homogènes. La projection perspective peut alors être réécrite sous forme linéaire par l'équation (Eq 2.3).

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_C \\ Y_C \\ Z_C \end{bmatrix} \quad (2.3)$$

Sur la base de ce modèle, les caméras embarquées sur les plateformes proposent en général une résolution de plusieurs millions de pixels, et une portée pouvant dépasser les 150m en fonction des conditions d'obstruction et d'éclairage. Les caméras embarquées sont associées à une focale  $f$  fixe, dont va directement dépendre le champ de vision de la caméra. Pour les caméras industrielles, on retrouve majoritairement trois types de champ de vision horizontaux : 1) large aux environs de 150° ; 2) moyen autour de 50° ; 3) restreint aux alentours de 25-30°. Verticalement les champs de vue sont généralement compris entre 20° et 60°.

Les données produites par une caméra monoculaire correspondent à des images représentant la scène contenue dans son champ de vision. En fonction du filtre utilisé, deux types d'images peuvent être obtenues (Fig 2.5). Les images obtenues avec un filtre coloré et dont les pixels sont composés de trois canaux : R, G et B<sup>2</sup>. Chaque canal est associé à une valeur comprise entre 0 et 255. L'association des valeurs de chaque canal permet d'obtenir le code de la couleur associée au pixel. L'infrarouge n'étant pas visible par l'œil humain, les images capturées avec ce filtre peuvent être colorisées en interpolant les valeurs  $[min; max]$  mesurées sur l'ensemble des pixels sur une échelle de  $[0; 255]$ . En fonction de cette échelle, les images sont colorisées avec différentes palettes de couleurs, telles que les niveaux de gris. Cela permet de les rendre interprétables par l'homme et d'appliquer des traitements comparables à ceux exploités sur des images RGB, mais sur un seul canal.

**Points forts et limitations.** Pour différentes raisons, les caméras RGB et infrarouges ont été les premiers capteurs exploités dans le cadre de la perception embarquée sur véhicule. Tout d'abord, le traitement d'image et la vision par ordinateur sont des domaines de recherche explorés depuis les années soixante et dans divers domaines scientifiques (robotique, médecine, production industrielle, militaire, etc...) [17].

2. RGB (Red, Green, Blue) : sigle représentant les trois couleurs primaires rouge, vert et bleu.

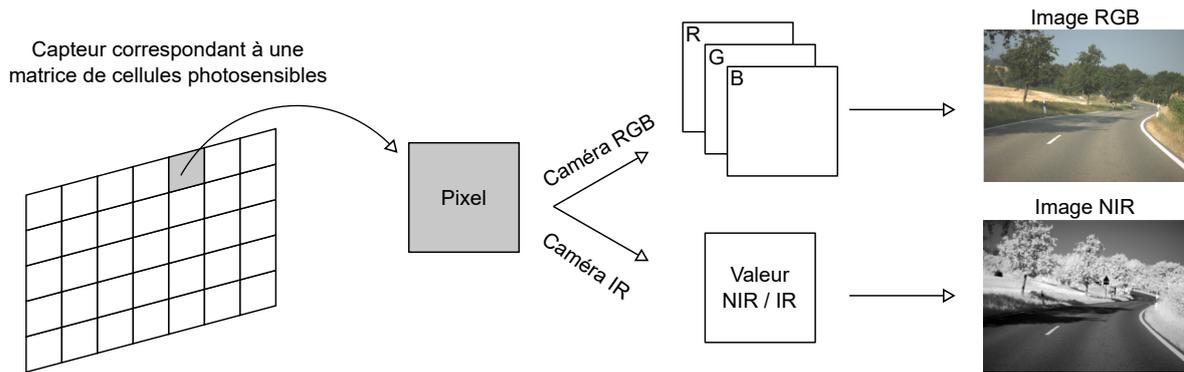


FIGURE 2.5 – Types de caméras : Représentation des deux types de caméras majoritairement exploitées pour la perception embarquée et du type d'image qu'elles produisent.

Cette technologie dispose donc d'un certain niveau de maturité. Les développements technologiques réalisés depuis ont également permis de diminuer drastiquement le coût de ces capteurs, tout en permettant : une résolution élevée, des informations de couleurs de haute qualité et une fréquence d'acquisition importante. Enfin, de par sa maturité, ce type de capteur dispose des méthodes de traitement d'images performantes et qui continuent d'être améliorées, notamment via le domaine du machine learning [18].

Ces capteurs possèdent néanmoins certains inconvénients. En effet, les images produites en haute résolution peuvent représenter un grand volume d'informations pouvant impacter le temps des traitements. Les caméras RGB sont également très sensibles aux conditions d'éclairage ainsi qu'aux conditions météorologiques (pluie, neige, brouillard,...). Les pics de lumière saturant l'image, tels qu'observables à la sortie d'un tunnel, où à l'inverse le manque de luminosité rendant l'image trop sombre, peuvent altérer les images produites, au point de les rendre inexploitable (Fig 2.6). Cela rend leur utilisation en conditions nocturne très limitée. Les caméras infrarouges de par les longueurs d'ondes qu'elles exploitent sont moins sensibles aux conditions lumineuses et météorologiques. Cependant, les valeurs que contiennent les pixels sont limités à un seul canal. On ne retrouve pas les informations RGB des caméras opérant dans le visible. On a donc une perte d'information qui peut rendre la distinction de certains objets plus complexe et qui peut diminuer les possibilités d'applications.



FIGURE 2.6 – Limites de la perception basée vision : Illustrations de phénomènes pouvant affecter la perception basée sur la vision par caméra : (a) Saturation de l'image à cause d'un pic de luminosité ; (b) Présence de gouttes sur l'objectif empêchant la bonne visibilité de la caméra ; (c) Manque de luminosité qui impacte également le bonne visibilité de la caméra. (Images de contexte issues d'internet).

Les caméras monoculaires souffrent également d'une ambiguïté d'échelle due à la perte d'une dimension lors de la projection du monde tridimensionnel sur un plan d'image bidimensionnel. Cela signifie que la transformation entre la caméra et son environnement peut être estimée avec précision, mais à un facteur d'échelle près [16]. Les distances entre une caméra monoculaire et les objets d'une scène perçue pourront être estimées, sous certaines hypothèses, en utilisant le modèle sténopé [16]. Des erreurs d'estimations de

distance pourront cependant être présentes, à cause de l'ambiguïté d'échelle. Enfin, il est difficile de leur attribuer une portée de mesure, celle-ci pouvant dépasser plusieurs centaines de mètres, ou se limiter à quelques mètres en fonction des conditions d'éclairage ou de l'obstruction du champ de vision par des objets.

Du fait de ces limites, une caméra est rarement exploitée seule dans le cadre des systèmes autonomes et ADAS. Pour contrebalancer ces limitations, plusieurs solutions ont vu le jour telles que l'association de caméras RGB et infrarouges ou l'utilisation de plusieurs caméras réparties sur l'ensemble du véhicule [19]. Les caméras peuvent aussi être combinées avec d'autres types de capteurs (LiDAR [20], radar [21]...). L'utilisation de plusieurs capteurs apporte cependant de nouvelles complexités liées à l'augmentation de la quantité d'informations à traiter ou à la mise en correspondance des données issues de différents capteurs.

### 2.2.1.2 Caméras stéréoscopiques

**Technologie.** Dans l'objectif de compenser le manque d'information des caméras monoculaires en ce qui concerne les mesures de distances, l'utilisation des caméras stéréoscopiques est apparue pour la perception des véhicules (Fig 2.7). Basée sur la technique de stéréovision, utiliser ce type de capteur permet d'apporter une information de profondeur à l'image capturée. Le fonctionnement de cette technique repose sur la combinaison de deux caméras monoculaires, parfaitement parallèles et séparées par une base : l'entraxe, dont la longueur est connue avec précision.



FIGURE 2.7 – Modèles de caméras stéréoscopiques : (a) Bosch Stereo video camera ; (b) StereoLabs ZED2.

**Modélisation du capteur.** Ces deux caméras possèdent une plage commune, correspondant à la zone sur laquelle les images issues des deux caméras se recouvrent. L'application du principe de triangulation permet ensuite d'estimer les distances situées entre le capteur et les différents pixels situés dans cette zone commune. Les distances, mesurées entre le capteur et les objets représentés par les pixels de la plage commune aux images, permettent ainsi d'apporter une troisième dimension aux données, en y intégrant des informations de distance.

Conformément au raisonnement présenté dans [16], on retrouve la distance capteur-pixel  $Z$  qui peut être estimée par l'équation (Eq 2.4).

$$Z = \frac{\text{entraxe} \cdot f}{\text{disp}} \quad (2.4)$$

où  $f$  représente la focale des caméras utilisées et  $\text{disp}$  représente la disparité donnée par l'équation (Eq 2.5).

$$\text{disp} = x'_1 - x'_2 \quad (2.5)$$

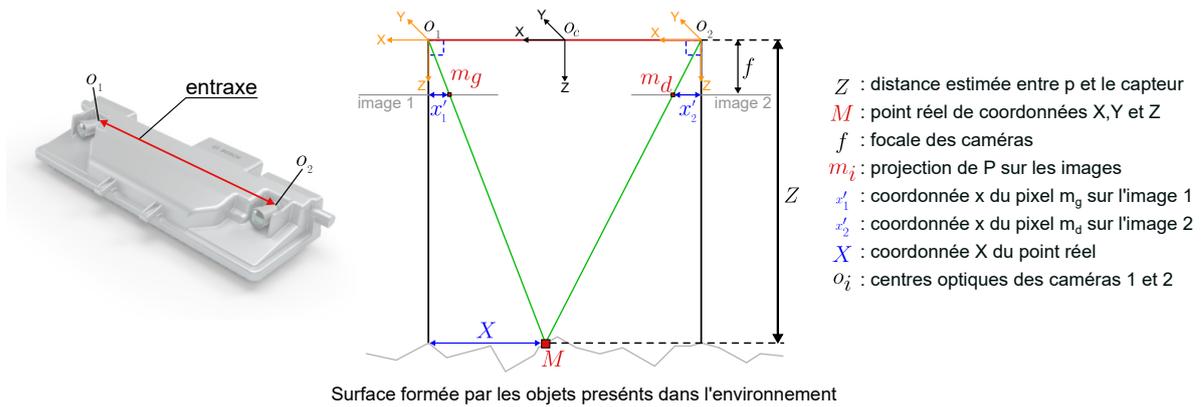


FIGURE 2.8 – Principe de la stéréo-vision.

**Points forts et limitations.** Les pixels disposent alors d'un canal supplémentaire au RGB, communément appelé D pour Depth (profondeur) et qui correspond aux distances estimées. Les données peuvent être représentées sous la forme d'un nuage de points 3D ou d'images RGB-D pouvant être colorisées soit en fonction des couleurs RGB, soit en fonction de niveaux de couleurs liés au canal D. La portée de mesures de distances des caméras stéréoscopiques dépend de leur entraxe. Les capteurs produits par les industriels disposent en général d'une portée de mesure située entre 10 et 50 mètres. La précision des distances mesurées varie cependant en fonction de l'entraxe utilisée par rapport à la distance mesurée [22]. Enfin, les caméras stéréo se basent sur l'utilisation commune de deux caméras 2D. On retrouve par conséquent les mêmes forces et faiblesses présentées précédemment pour les caméras 2D, notamment du point de vue de la sensibilité aux conditions environnementales.

### 2.2.2 Capteurs actifs

Les capteurs actifs se composent a minima d'un émetteur et d'un récepteur. Via l'émetteur, ils produisent leur propre onde pour « éclairer » artificiellement les objets de l'environnement avant de mesurer, via le récepteur, le signal que ces objets renvoient vers le capteur. En produisant leurs propres ondes, ces capteurs ont l'avantage d'être moins dépendants des conditions d'éclairage, ce qui réduit leur sensibilité aux conditions météorologiques. Dans le cadre de la perception pour la robotique, les ADAS ou la cartographie, les ondes produites pour sonder l'environnement sont exploitées pour mesurer les distances séparant le capteur des objets détectés dans cet environnement.

Parmi les capteurs actifs exploités dans le cadre de la perception embarquée, on dénote deux principes de fonctionnement différents et majoritairement utilisés : il s'agit des mesures basées sur le « temps de vol direct » et des mesures basées sur la « différence de phase » (ou Effet Doppler).

**Mesures basées « temps de vol direct ».** Ce principe de fonctionnement mesure le temps de parcours d'une onde entre sa source d'émission (le capteur), et son point d'impact avec un objet rencontré. Pour ce faire, les capteurs utilisent un émetteur produisant un signal par impulsion pour chaque mesure ou trame d'acquisition en fonction du capteur utilisé. Le signal émis peut être une onde de différente nature et est dirigée par l'émetteur afin de connaître son orientation de départ. Lorsque l'onde rencontre un objet, celle-ci est réfléchiée et renvoyée en direction de sa source, où elle est captée par un récepteur (Fig 2.9).

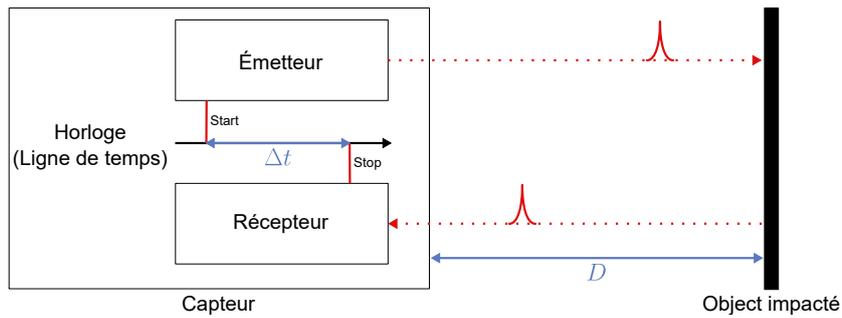


FIGURE 2.9 – Principe de mesure basé sur le temps de vol.

En mesurant, via une horloge de haute précision, le temps  $\Delta t$  séparant l'émission de la réception, il est possible d'en déduire la distance séparant le capteur de l'objet impacté par l'équation (Eq 2.6) :

$$D = \frac{1}{2} \cdot c \cdot \Delta t \quad (2.6)$$

où  $D$  est la distance mesurée,  $c$  est la vitesse de la lumière et  $\Delta t$  le temps mesuré entre l'émission et la réception, soit le temps mis par l'onde pour parcourir deux fois la distance séparant le capteur de la cible impactée : émetteur-cible puis cible-récepteur.

**Mesures basées sur la différence de phase.** Les mesures basées sur la différence de phase se fondent sur le même principe global que pour les mesures basées sur le « temps de vol direct » : une onde est émise, elle impacte un objet détecté, puis est renvoyée vers la source avant d'être captée par le récepteur. La différence est qu'avec ce second principe, c'est le décalage de phase entre l'onde émise et celle réfléchi qui est mesuré, et non le temps de parcours de l'onde. (Fig 2.10).

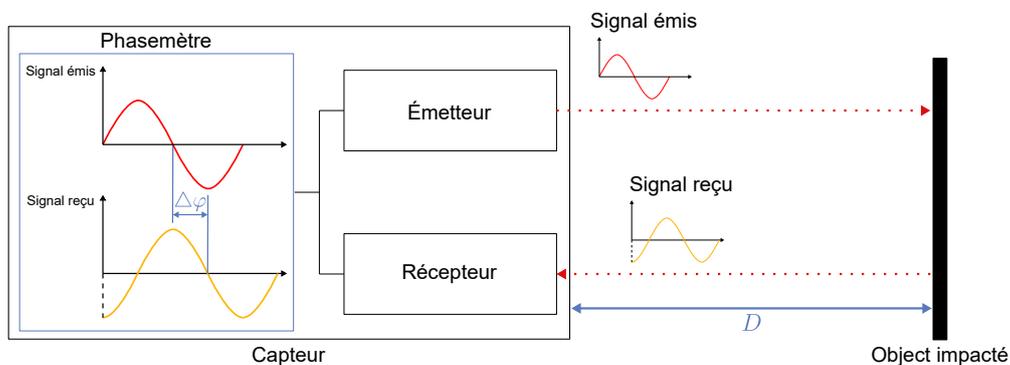


FIGURE 2.10 – Principe de mesure basé sur la différence de phase.

Pour ce faire, l'onde émise est modulée par un signal sinusoïdal défini. Lorsque l'onde impacte un objet qui la renvoie vers le capteur, une comparaison va être faite entre l'onde émise et celle renvoyée. Le récepteur va ainsi recevoir un signal ayant les mêmes caractéristiques, mais qui est déphasé par rapport au signal initialement émis. Un phasemètre peut alors être utilisé lors de cette comparaison pour mesurer le déphasage  $\Delta\varphi$ . Ce même déphasage peut également être exprimé par l'équation (Eq 2.7).

$$\Delta\varphi = 2\pi \cdot F \cdot \Delta t = 2\pi \cdot F \cdot \frac{2D}{c} \quad (2.7)$$

Connaissant la valeur du déphasage  $\Delta\varphi$  grâce à l'utilisation du phasemètre, ainsi que la fréquence du signal, il est possible d'en déduire la distance séparant le capteur de l'objet détecté grâce à l'équation (Eq 2.8).

$$D = \frac{\Delta\varphi \cdot c}{4\pi \cdot F} \quad (2.8)$$

où  $D$  la distance capteur-objet mesurée,  $c$  représente la célérité,  $F$  représente la fréquence du signal émis et  $\Delta t$  le décalage temporel entre le signal émis et signal reçu.

On notera toutefois que la portée maximale des mesures va dépendre de la valeur de la phase.

### 2.2.2.1 LiDAR

**Technologie.** Depuis un peu plus de dix ans, les capteurs utilisant le laser de plus en plus présents dans le cadre de la perception embarquée. Bien que les LiDARs peuvent utiliser différents principes de fonctionnement, ceux exploités dans le cadre d'acquisition sur plateforme terrestre (robotique, véhicule autonome ou cartographie), sont majoritairement basés sur le principe de mesure par temps de vol direct (Fig 2.11). Pour exploiter ce principe, les LiDARs utilisent un émetteur laser pour produire une impulsion lumineuse dont la longueur d'onde varie en fonction du capteur : entre l'infrarouge proche et l'infrarouge moyen [905nm - 1500nm]. Cette impulsion émise par le laser sous la forme d'un faisceau lumineux de petit diamètre (de l'ordre du millimètre) possède une divergence très faible. Cela permet à l'onde lumineuse émise d'être cohérente spatialement et temporellement.

Ce type de capteur mesure ainsi le temps de parcours de chaque impulsion entre l'émission par l'émetteur laser et la réception via un récepteur optique (photodétecteur ou photodiode). Chaque mesure est représentée par un point, dont les coordonnées sont exprimées dans un repère propre au capteur. Elles sont calculées à partir de l'orientation initiale du faisceau laser (azimut et angle vertical) et de la distance déterminée à partir du temps de vol.

|                             |  |  |  |  |
|-----------------------------|---|---|---|---|
| Caractéristiques            | Velodyne HDL-64E  | Velodyne VLS-128  | Sick LMS5xx   | Sick LMS5xx   |
| Type / Nombre de nappes     | 3D / 64 nappes  | 3D / 128 nappes   | 2D / 1 nappe  | 2D / 1 nappe  |
| Champs de vision hz / v     | 360° / 26.9°  | 360° / 40°  | 190° / X  | 360° / X  |
| Portée de mesure            | 1m - 120m   | 1m - 220m   | 1m - 80m  | 1m - 420m   |
| Fréq. max / nombre de pts/s | 15Hz / ~2.2M  | 20Hz / ~4.8M  | 100Hz   | 250Hz / ~1M   |
| Résolution horizontale      | 0.08°   | 0.11°   | 0.04°   | 0.001°  |
| Résolution verticale        | ~0.4°   | ~0.4°   | N/A   | N/A   |

FIGURE 2.11 – Modèles de LiDARs : Exemples et caractéristiques de capteurs LiDARs couramment embarqués sur des véhicules terrestres.

Les mesures réalisées par ce type de capteur se traduisent sous la forme de points 2D, ou 3D. Les trames de données vont représenter l'ensemble des mesures réalisées pour une rotation complète et forment ainsi des nuages de points. Dans le cadre des LiDARs 2D, un seul faisceau laser est utilisé conjointement à un miroir monté sur un système rotatif, lui permettant de balayer la scène à 360° et perpendiculairement à l'axe de rotation (Fig 2.12-a). Les points mesurés sont définis dans un repère cylindrique par un angle

et une distance. Les LiDARs 3D exploitent également un système rotatif, mais utilisent plusieurs diodes laser simultanément, ce qui leur permet de couvrir un champ de vue plus important. L'axe de rotation étant généralement placé verticalement, ces LiDARs disposent d'un champ de vision de  $360^\circ$  à l'horizontal et pouvant varier de  $20^\circ$  à  $40^\circ$  à la verticale en fonction du nombre de faisceaux et de leur répartition. On retrouve ainsi des capteurs utilisant de 4 à 128 faisceaux lasers avec des architectures semblables à celle représentée sur la figure (Fig 2.12-b). Dans les chapitres suivants du manuscrit, nous nous focaliserons sur les LiDARs 3D, qui offrent un champ de vision plus important et nous utiliserons la dénomination générale LiDAR.

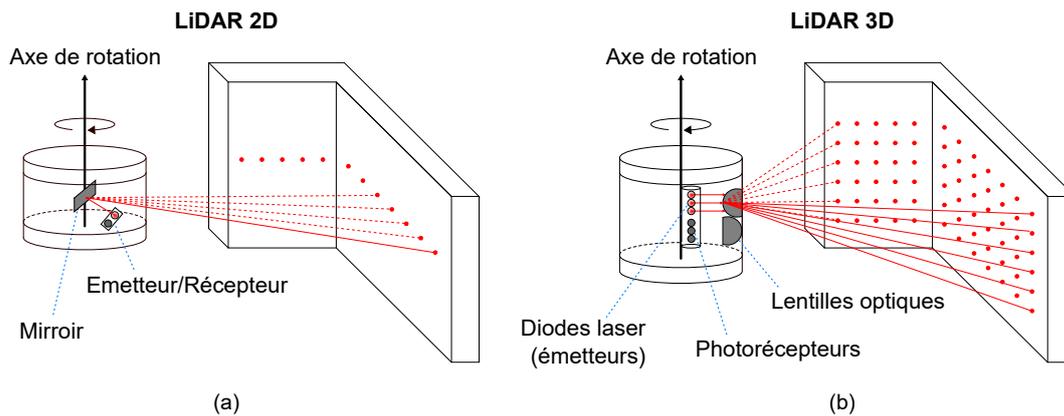


FIGURE 2.12 – Types de LiDARs à balayage : (a) LiDAR 2D ; (b) LiDAR 3D.

**Modélisation du capteur.** Pour modéliser un capteur LiDAR, on peut s'appuyer sur la modélisation d'un télémètre. Via un émetteur laser, il émet un faisceau dans une direction et mesure dans une dimension, la distance séparant le point d'émission du premier élément rencontré sur la trajectoire du faisceau. Un LiDAR peut être assimilé à un ensemble de télémètres répartis verticalement grâce à différents angles d'élévations. Le modèle de base du télémètre peut être complété par plusieurs éléments pour établir le modèle du capteur LiDAR noté  $M_L$  :

- le bruit de mesure de distance : déjà présent pour le modèle du télémètre et qui permet de caractériser la précision de mesure de distance ;
- le nombre de nappes, correspondant aux nombres d'émetteurs lasers répartis verticalement et donc au nombre de faisceaux exploités ;
- la répartition verticale des nappes : qui correspond aux angles d'élévation  $\mathcal{V}$  propres à chacune des nappes ;
- le bruit de mesure d'angle d'élévation : correspondant à la précision avec laquelle sont définis les angles d'élévation propres à chaque nappe ;
- la résolution azimutale  $r_{Az}$  : correspondant à l'angle horizontal qui sépare deux mesures ;
- le bruit de mesure d'angle azimutal : correspondant à la précision des azimuts mesurés.

Chaque mesure du LiDAR se traduit par un point 3D et est composée par 3 informations : une distance  $D$  (calculée sur le principe du temps de vol direct), un azimut  $Az$  et un angle d'élévation  $\mathcal{V}$ . A partir de ces informations, il est possible de positionner le point dans l'espace 3D en utilisant un système de coordonnées sphériques local, dont l'origine est centrée sur le capteur (Fig 2.13).

Bien que les coordonnées des points soient initialement sphériques, elles sont en général (mise à part pour des traitements spécifiques) converties en coordonnées cartésiennes  $(X, Y, Z)$  d'unité métrique.

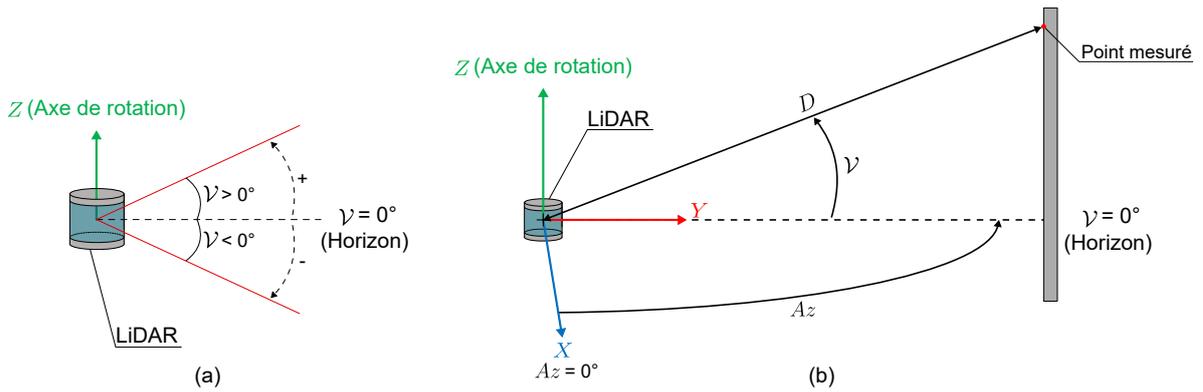


FIGURE 2.13 – Modélisation du capteur LiDAR : (a) Sens d'évolution de l'angle d'élévation  $\mathcal{V}$  par rapport à l'horizon, où  $\mathcal{V} = 0$ ; (b) Représentation des informations composant la mesure et permettant de positionner les points en 3D dans un repère sphérique.

Pour cela, on utilise le modèle de capteur LiDAR ( $M_L$ ) inversé. On obtient alors l'équation (Eq 2.9).

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = M_L^{-1}(D, Az, \mathcal{V}) = \begin{bmatrix} D \cos Az \cos \mathcal{V} \\ D \sin Az \cos \mathcal{V} \\ D \sin \mathcal{V} \end{bmatrix} \quad (2.9)$$

Chaque mesure LiDAR se traduit par un point 3D dans un référentiel local cartésien, avec un repère orthonormé où l'axe de rotation coïncide avec l'axe Z et où les X et Y sont orientés selon le modèle de capteur utilisé. Un point  $p$  est au minimum constitué des coordonnées X, Y et Z. D'autres informations et attributs peuvent compléter ce point, tels que l'intensité  $I$  interpolée sur une échelle de valeurs  $[0; 255]$  en fonction de la valeur obtenue du signal renvoyé [23], l'index de la nappe *ring* auquel il appartient (Fig 2.14) ou encore l'horodatage *time* de la mesure.

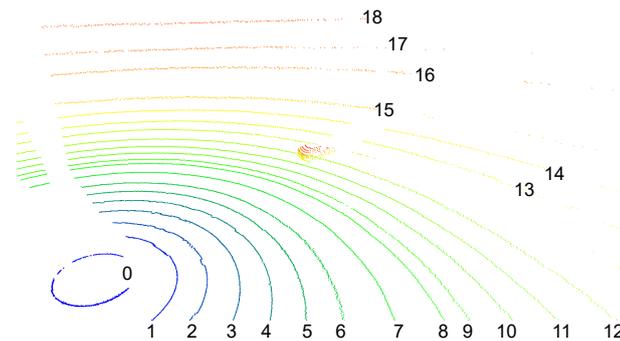


FIGURE 2.14 – Scan LiDAR : Extrait d'un scan LiDAR 32 nappes. Les points constituant les scans sont colorisés en fonction de la valeur de leur attribut *ring*.

Ce type de capteur permet de disposer d'une grande capacité de perception. Cela est dû à leur champ de vision ( $360^\circ$  horizontalement et jusqu'à  $40^\circ$  verticalement en fonction des modèles de LiDAR), couplé à une portée qui est généralement de l'ordre de 150 à 250m. Ils peuvent fonctionner à des fréquences de rotation atteignant 20Hz et permettent de mesurer jusqu'à plus d'un million de points par seconde. La résolution angulaire verticale du LiDAR est fixe et va directement dépendre de la répartition verticale des nappes et de leur angle d'élévation  $\mathcal{V}$ . En revanche, la résolution azimutale  $r_{Az}$  va varier en fonction de la fréquence d'acquisition paramétrée.

Étant donné la grande quantité d'informations que peut fournir un LiDAR, à fréquence soutenue, les points ne sont pas transmis unitairement, mais par « paquets de points » que l'on nommera *scan*. En partant de l'azimut de référence ( $Az = 0^\circ$ ), un scan est transmis par le capteur après chaque rotation complète autour de l'axe  $Z$ . Les scans représentent ainsi des nuages de points répartis sur  $360^\circ$  (Fig 2.14).

**Points forts et limitations.** L'importante capacité de perception des LiDARs permet de scanner l'environnement en 3D et d'en créer une carte détaillée, où des objets de petites dimensions peuvent également être détectés. L'utilisation d'ondes lumineuses cohérentes, permet d'atteindre une précision centimétrique. Ces ondes sont également peu divergentes, ce qui contribue à limiter le bruit de mesure. Les mesures étant indépendantes les unes des autres, cela permet d'éviter qu'une trame entière de données soit bruitée comme ça peut être le cas avec les images issues de caméras. Grâce à ces avantages, les cartes obtenues à partir des nuages de points représentent l'environnement numérisé fidèlement et de manière détaillée (Fig 2.15). Étant des capteurs actifs, les LiDARs ne sont pas impactés par les conditions d'éclairage ou les ombres, comme peuvent l'être les capteurs passifs. Ils peuvent être exploités de jour comme de nuit. Enfin, bien que chaque scan décrive l'environnement de façon figée, à un instant  $t$  daté, la comparaison entre plusieurs scans acquis successivement permet de différencier les objets statiques des objets dynamiques.

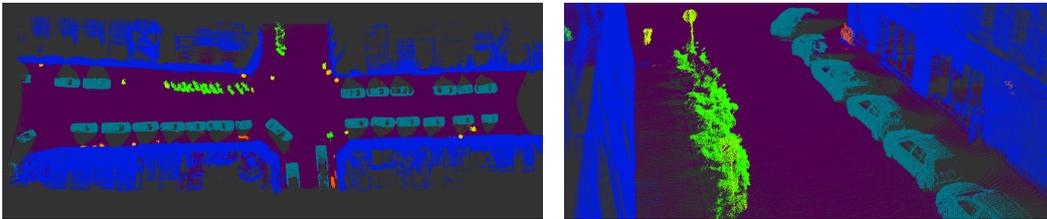


FIGURE 2.15 – Exemple de données type mesurées par un capteur LiDAR : Nuage de point 3D issue de l'accumulation de scans et représentant la rue Madame située à Paris [24].

Les capteurs LiDARs ne sont néanmoins pas parfaits. Cette technologie reste relativement nouvelle et représente, pour le moment, un coût important vis-à-vis de technologies plus matures (caméras ou radars), ainsi qu'une taille et un poids pouvant être plus imposants. Un second inconvénient est que ce type de capteur ne mesure pas les informations traditionnellement mesurées par les caméras telles que les couleurs. On perd donc les informations couramment utilisées dans le cadre de la perception, comme par exemple la reconnaissance de panneaux, la détection d'objets ou plus généralement, pour une meilleure compréhension contextuelle de la scène. Enfin, dans certaines conditions météorologiques, l'usage du LiDAR peut être limité. Les mesures peuvent être bruitées par certains phénomènes : pluie, neige, poussière ou brouillard, ce qui peut amener à de « fausses détections ». Les éléments réfléchissants ou les surfaces mouillées telles que le sol peuvent également impacter les mesures et la portée du capteur. En réfléchissant l'onde de façon spéculaire, cette dernière n'est pas redirigée vers le récepteur qui ne peut alors pas terminer la mesure. L'élément réfléchissant n'est alors pas perçu par le capteur (Fig 2.16).

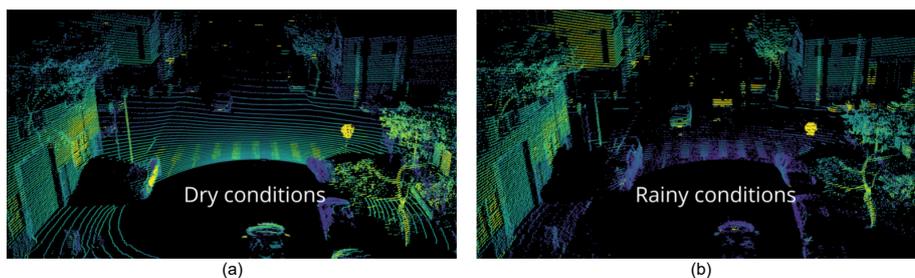


FIGURE 2.16 – Représentation de l'impact de la pluie sur des données LiDAR : (a) Scan LiDAR mesuré en conditions météorologiques favorables ; (b) les données mesurées au même lieu en conditions de pluie [25].

**Remarque :** les capteurs présentés dans la suite de ce chapitre n'ont pas pu être exploités dans le cadre de ces travaux de thèse. Ils sont introduits de façon plus synthétique que les précédents capteurs et pourront être approfondis lors de futurs travaux. Il reste néanmoins important d'avoir un aperçu de ces capteurs couramment exploités dans le cadre de la perception.

### 2.2.2.2 Radar

De par leur utilisation dans des domaines très variés et depuis le début du 20<sup>ème</sup> siècle, les capteurs radars représentent une technologie de perception qui est plus mature que le LiDAR. Elle est par conséquent moins coûteuse et a ainsi été intégrée plus tôt dans le domaine automobile.

C'est pourquoi on retrouve déjà ce type de capteur sur de nombreux véhicules de série. Il y est exploité par différents systèmes tels que le régulateur de vitesse adaptatif, le système anti-collision ou le freinage d'urgence. Son rôle réside principalement dans la détection de cibles statiques ou dynamiques autour du véhicule, ainsi que dans la mesure de vitesse et direction du déplacement des cibles dynamiques.



FIGURE 2.17 – Exemple de radars exploités pour la perception.

A l'inverse des LiDARs, le radar exploité dans le domaine automobile est principalement basé sur les mesures par différence de phase. Pour ce faire, leur antenne émet des micro-ondes radar en continu, à différentes fréquences communément utilisées : 24, 77 ou 79GHz. Une fois l'onde entrée en contact avec un objet, elle est renvoyée vers la source. Le récepteur capte cette onde réfléchie, qui est appelée « écho radar » et le déphasage est alors exploité pour déterminer la distance séparant le capteur de la cible impactée.

Le principal avantage des capteurs radar est qu'ils peuvent atteindre une portée de plusieurs centaines de mètres, sans représenter un coût important comme c'est le cas pour les LiDARs. Grâce aux mesures réalisées en continu, ils apportent une information supplémentaire en estimant la vitesse et la direction des cibles impactées. Ils sont également moins sensibles aux conditions météorologiques (pluie, brouillard...) vis-à-vis des autres capteurs présentés jusqu'à présent. Ils ne dépendent pas non plus des conditions d'éclairage.

Ils sont en revanche très sensibles aux objets métalliques, ce qui bruite les données et cause de nombreuses fausses détections. De plus, leurs mesures peuvent être impactées par des phénomènes de rebonds ou de multi-trajets, ce qui impacte également la précision des mesures. Par ailleurs, les échos radars n'apportent pas suffisamment d'informations permettant de distinguer clairement la forme des objets détectés et limitent ce type de capteur à une résolution spatiale faible, de l'ordre du mètre. Enfin, bien que leur portée soit intéressante, ils possèdent un champ de vision horizontal réduit par rapport aux possibilités que peut offrir un LiDAR, comme on peut le voir (Fig 2.18).

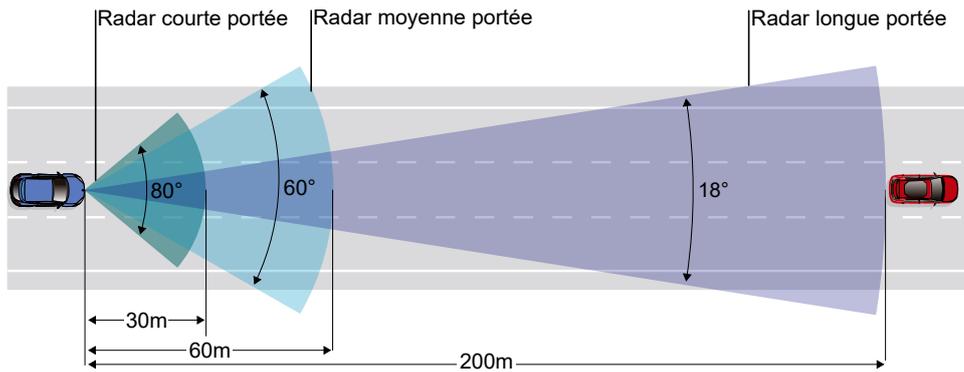


FIGURE 2.18 – champs de vue des différents types de radar.

### 2.2.2.3 Ultrasons

Un autre type de capteur dont la technologie est maîtrisée, mature et qui équipe la quasi-totalité des véhicules de série actuels, est le capteur à ultrasons. Les mesures de distances sont réalisées via l'utilisation d'ondes sonores de fréquence située entre 20kHz et 40kHz. Ces capteurs reposent sur le temps de vol direct (Eq 2.6), à la différence qu'ici,  $c$  représente la vitesse de l'onde sonore et non la vitesse de la lumière. Ils sont principalement utilisés pour détecter des obstacles à très faible distance. On les retrouve notamment pour les systèmes d'assistance au stationnement ou la détection d'obstacles situés dans un environnement proche et à basse vitesse. Récemment, ils ont également pu être exploités dans le cadre des avertisseurs d'angles morts.



FIGURE 2.19 – Exemple de capteurs à ultrasons.

Les avantages de ce type de capteur sont qu'il est à très faible coût, tout en restant fiable dans le temps. Il peut interagir avec tout type de matériaux et n'est pas impacté par la poussière, les conditions d'éclairage ou les conditions météorologiques.

En revanche, les mesures peuvent être sujettes à des phénomènes de rebonds amenant à de fausses détections. De plus, pour rester à bas coûts, ces capteurs sont généralement équipés d'horloges peu précises, ce qui impacte la précision des mesures réalisées. Le principal handicap de cette technologie du point de vue de la perception est qu'ils possèdent un champ de vision ainsi qu'une portée de mesure qui sont très limités. Du fait de la perte d'énergie lorsque la distance mesurée augmente, la portée de mesure est en général de quelques mètres, allant jusqu'à une dizaine de mètres pour certains capteurs. On retrouve également une zone blanche située entre le capteur et sa portée minimale, généralement autour de 15 centimètres. Enfin, ce type de capteur souffre également d'un temps de réaction important, dû à la vitesse de propagation des ondes sonores qui est nettement inférieure à celle des ondes lumineuses. Cela réduit davantage les possibilités d'usages de ce type de capteur dans le cadre de la perception.

### 2.2.2.4 Caméras 3D / LiDARs flash

Enfin, parmi les capteurs actifs, on notera également l'apparition plus récente des LiDARs flash, ou « solid-state », ainsi que des caméras 3D. Leurs fonctionnements respectifs sont comparables l'un de l'autre. Tous deux fonctionnent sans système mécanique, ce qui permet de réduire leur coût, leur taille, ainsi que les risques liés aux systèmes mécaniques tels que les pannes. Majoritairement destinés à être des capteurs accessibles à bas coût, ils se basent sur le principe de différence de phases qui ne nécessite pas d'horloge précise.



FIGURE 2.20 – Exemples de LiDARs flashs (a,b) et de caméras 3D (c,d,e) : (a) Velodyne Velarray ; (b) Valeo Scala ; (c) Microsoft Kinect ; (d) Intel RealSense.

Les données produites par ces capteurs sont des nuages de points 3D à l'image de ceux produits par les LiDARs « classiques ». Ne pas dépendre de système mécanique permet à ces capteurs de disposer d'une fréquence élevée de mesures. Ils disposent d'un champ de vision comparable, de l'ordre de 100° horizontalement et 50° verticalement. En revanche, ils se différencient par leur portée de mesure qui s'étend sur une dizaine de mètres pour les caméras 3D. Les LiDARs flash quant à eux peuvent atteindre une portée variable en fonction du modèle et située entre 50 et 200 mètres.

Bien que ces types de capteurs présentent de nombreux avantages vis-à-vis de l'intégration sur véhicules en série, ils correspondent à des technologies récentes, apparues au cours des années 2010. Elles ne sont donc pas suffisamment matures et restent encore peu explorées pour des utilisations dans le cadre des ADAS. Les avancées prochaines pourraient cependant leur permettre de prendre l'avantage sur les autres capteurs présentés précédemment du point de vue de leur rapport qualité/prix.

### 2.2.3 Synthèse comparative des capteurs

Nous avons pu voir dans les sous-sections 2.2.1 et 2.2.2 que les capteurs présentés peuvent être liés à différentes fonctionnalités dépendantes de leur capacité de perception mais également de leurs limitations. L'objectif étant de détecter les éléments de l'environnement le plus en amont possible afin d'anticiper au maximum les actions à réaliser et d'avoir de la redondance d'informations. Pour répondre au mieux à cet objectif, deux critères primordiaux caractérisent ces capteurs : leur champ de vision et leur portée de mesure. On peut les représenter de manière synthétique par la (Fig 2.21).

Leur champ de vue et leur portée ne sont cependant pas les seuls critères à prendre en considération. Afin de synthétiser cette première partie dédiée aux sources de données de perception, nous nous sommes inspiré des travaux réalisés dans [9] pour évaluer ces capteurs. Nous nous sommes basés sur une liste de critères repris dans le tableau (Tab 2.1). Pour comparer de façon objective les différents capteurs, ces critères ont été choisis de manière à représenter des aspects comparables des capteurs. Ils sont quantifiés par un score numérique : 1 - faible, 2 - moyen et 3 - élevé, et sont réparties dans trois catégories distinctes :

- Capacité de perception : comprenant le champ de vision du capteur, sa portée de mesure et sa fréquence d'acquisition ;
- Pertinence des données : précision de mesure, résolution, sensibilité aux conditions météorologiques et sensibilité aux conditions d'éclairage ;
- Facilités d'usage : prix du capteur, facilité de lecture des données (poids, compréhension contexte,...) et facilité d'intégration (tailles, poids, positions possibles...).

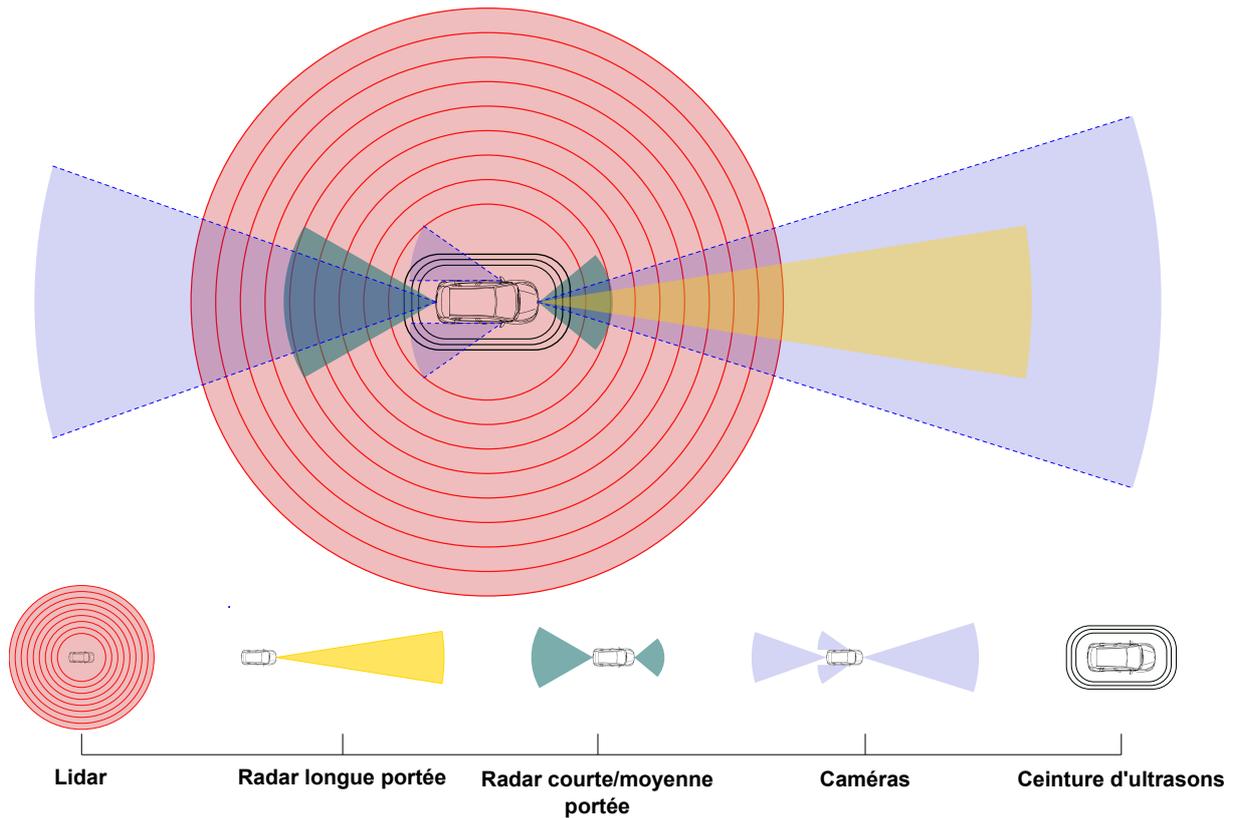


FIGURE 2.21 – Champs de vue et portée des différents types de capteurs.

A partir de ces critères, nous pouvons dresser le tableau (Tab 2.1) permettant de comparer ces différentes sources de données de perception.

| Critères \ Capteurs        | Caméras |    |        | LiDARs |    | Radar | Ultrasons | Lidar flash | Caméras 3D |
|----------------------------|---------|----|--------|--------|----|-------|-----------|-------------|------------|
|                            | 2D      | IR | Stéréo | 2D     | 3D |       |           |             |            |
| Champ de vue               | 2       | 2  | 2      | 1      | 3  | 2     | 1         | 2           | 2          |
| Portée                     | 2       | 2  | 2      | 3      | 3  | 3     | 1         | 2           | 1          |
| Fréquence d'acquisition    | 3       | 3  | 3      | 2      | 2  | 2     | 2         | 3           | 3          |
| Précision de mesure        | 1       | 1  | 1      | 1      | 3  | 1     | 1         | 1           | 1          |
| Résolution                 | 3       | 3  | 3      | 3      | 2  | 1     | 1         | 2           | 2          |
| Sensibilité météorologique | 3       | 1  | 3      | 3      | 2  | 1     | 1         | 2           | 2          |
| Sensibilité d'éclairage    | 3       | 1  | 3      | 3      | 1  | 1     | 1         | 1           | 1          |
| Prix                       | 1       | 2  | 2      | 2      | 3  | 1     | 1         | 2           | 2          |
| Facilité d'intégration     | 2       | 2  | 2      | 2      | 1  | 3     | 3         | 2           | 2          |
| Facilité de lecture        | 3       | 2  | 2      | 2      | 2  | 1     | 2         | 2           | 2          |

TABLE 2.1 – Évaluation comparatives des capteurs extéroceptifs : Tableaux des scores obtenus par les différents capteurs en fonction des critères présentés.

A partir des scores obtenus pour les différents critères, on peut représenter de manière graphique les résultats afin de les comparer entre eux mais également à un capteur « idéal » (Fig 2.22). Ce qui ressort de ces graphiques est que, si l'on prend l'ensemble des critères en compte, ces différents capteurs possèdent chacun des avantages et inconvénients. Dans l'état actuel, et pour pallier les limites observées et propres à chaque capteur, les systèmes de perception peuvent être composés par plusieurs capteurs identiques répartis sur l'ensemble du véhicule ou par la combinaison de différents types de capteurs. Ces deux types de configurations peuvent apporter une complémentarité permettant au système de perception de se rapprocher du capteur de perception idéal.

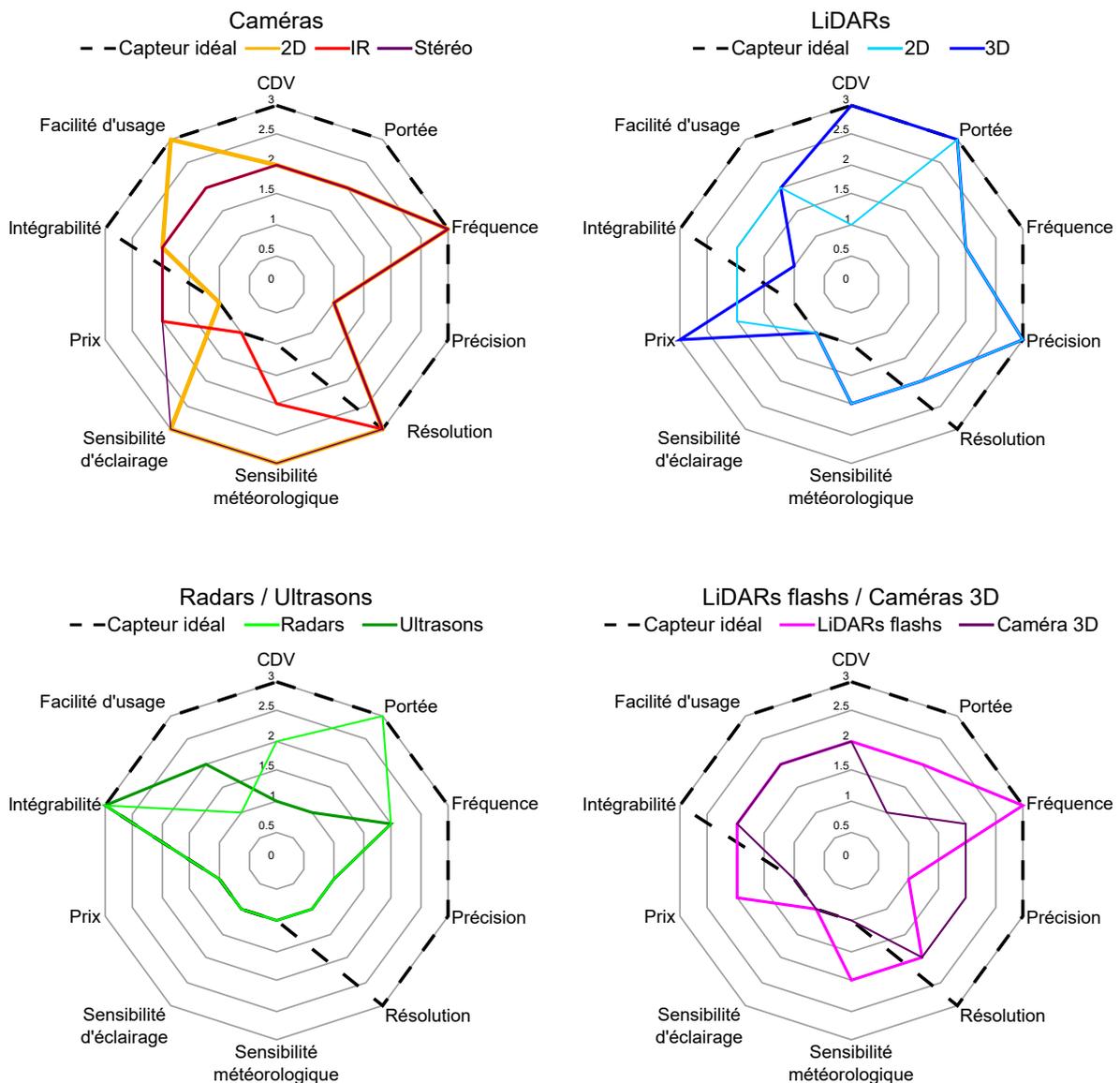


FIGURE 2.22 – Analyse comparative des différents types de capteurs : Comparaison des différents capteurs extéroceptifs vis-à-vis d'un capteur idéal, en fonction des critères présentés.

Cependant, d'un point de vue purement perception, sans prendre en compte les limites liées aux critères de facilités d'usage, on remarque que le type de capteur se rapprochant le plus du capteur parfait correspond au LiDAR 3D. Il représente actuellement la source de données la plus intéressante mais reste limité par son prix peu compétitif et les difficultés de son intégration sur un véhicule de série. Cependant, la démocratisation de cette technologie devrait lui permettre d'améliorer ces points dans les prochaines années et faciliter son exploitation sur des véhicules en série.

## 2.2.4 Moyens de représentation des données de perception

### 2.2.4.1 Images RGB et RGB-D

On l'a vu avec la description des caméras monoculaires, les images RGB correspondent à une représentation d'une scène 3D projetée sur un plan 2D : le plan image. Les données sont représentées sur un plan image et stockées dans une matrice de pixel. La représentation des données par images RGB est celle qui se rapproche le plus de la vision humaine.

En exploitant les couleurs visibles par l'œil, il s'agit du moyen de représentation le plus facilement interprétable pour l'Homme. De plus, la représentation d'une scène basée sur deux dimensions permet de réduire la complexité de calcul et des traitements par rapport à ceux appliqués sur des données 3D.

Les images RGB-D, également appelées image de profondeur « depth image » est un moyen de représentation généralement utilisé avec des capteurs caméras stéréoscopiques. Il s'appuie sur les images RGB composées des trois canaux  $R$ ,  $G$  et  $B$ , auxquels on ajoute un nouveau canal  $D$  (Depth) [26]. L'information contenue dans ce canal représente la profondeur associée au pixel, soit la distance calculée entre capteur et l'objet représenté par le pixel. On a donc une représentation appelée 2.5D, qui contient des informations 3D, mais sous une forme 2D augmenté par un attribut. Cela permet de conserver un avantage vis-à-vis de la complexité des calculs opérés en 3D. Cependant, comme pour les images RGB, une étape de projection sur un plan image est nécessaire.

#### 2.2.4.2 Grille d'occupation

Les grilles d'occupation, introduites dans [27], sont utilisées pour discrétiser l'environnement sous la forme d'une grille 2D régulière. Les cellules qui composent cette grille sont associées à une valeur de probabilité d'occupation. Cette valeur permet de quantifier la confiance dans le fait que l'espace représenté par la cellule soit occupé ou non par un obstacle, et si cet obstacle est fixe ou dynamique [28]. Cette représentation des données est principalement ciblée sur les capteurs qui mesurent des distances. Les mesures de distances sont ensuite projetées sur un plan horizontal et permettent de mettre à jour la probabilité d'occupation des différentes cellules. Comme pour les images, l'environnement 3D est projeté sous la forme d'une représentation 2D, cette fois-ci horizontale.

Cette représentation de l'environnement perçu apporte de l'efficacité dans les traitements en réduisant la complexité, mais éloigne les données de leur moyen de représentation initial, ce dernier pouvant être plus représentatif de la scène perçue. En fonction de la résolution fixée pour la grille, cela va engendrer une simplification de la scène et potentiellement une perte des détails de la scène qui peut être conséquente si cette résolution fixée est trop faible. Lorsque plusieurs capteurs de différents types équipent une même plateforme, les grilles d'occupation représentent un moyen efficace pour rassembler l'ensemble des données autour d'un seul moyen de représentation.

#### 2.2.4.3 Nuage de points 3D

Le nuage de points 3D est un moyen de représentation des données principalement utilisé par les LiDARs [29]. Il a cependant aussi pu se développer avec les caméras stéréoscopiques, les caméras 3D ou encore certains Radars. Comme on a pu le voir via l'étude du LiDAR, les données sont représentées par des points 3D, avec des coordonnées  $(X, Y, Z)$  cartésiennes. Ce moyen de représentation permet d'avoir des informations au plus proche de la scène et évite de devoir appliquer des projections vers l'espace 3D. De façon native les nuages de points ne permettent pas de retrouver les informations de couleurs dont disposent les images. Ils peuvent néanmoins être colorisés en fonction de l'intensité des points et utiliser l'intensité dans les traitements [30]. Une autre méthode consiste à combiner une caméra dont le champ de vision couvre le nuage et permet, par projection d'associer les points du nuage aux pixels de l'image pour en récupérer les informations RGB. Cela implique une combinaison de capteur adéquate et un traitement supplémentaire des données pour opérer cette association.

#### 2.2.4.4 *k*-d tree et dérivés

Un *k*-d tree (ou arbre *k*-d) correspond à une structure permettant de partitionner spatialement l'espace et de structurer les données en fonction de leur répartition dans l'espace [31]. Il inclut le cas spécifique des arbres binaires, où les nœuds contiennent un point en dimension *k*. La structuration des données apportée par ce type de structure permet d'optimiser la recherche locale de données multidimensionnelles (par ex : recherche des plus proches voisins pour des points 3D). Il n'est plus nécessaire de parcourir linéairement l'ensemble des données, la recherche peut être ciblée localement. A partir de cette structure de base, d'autres types de structures sont apparues avec notamment les octrees et les quadtrees.

**Octree.** Les octrees correspondent à un type de structure utilisé pour partitionner l'espace 3D en sous-espaces [32], nommés par la suite voxels<sup>3</sup>. L'espace est subdivisé de façon récursive en huit sous-espaces. L'arborescence de la structure octree va être définie par le lien de parenté entre un voxel parent et les 8 sous-espaces qui le constituent : les voxels enfants (Fig 2.23-a). Un octree est défini par son niveau de profondeur maximal  $d_{max}$ . Au niveau le plus profond, l'espace est divisé en  $N = 2^{d_{max}} \cdot 2^{d_{max}} \cdot 2^{d_{max}}$  tel qu'illustré (Fig 2.23-b,c).

Dans la pratique, les octrees sont principalement exploités pour structurer les nuages de points 3D afin d'optimiser les traitements opérés sur ces nuages. Les bibliothèques telles qu'OctoMap [33] utilisent un paramètre  $d_{max}$  fixe mais permettent de définir la résolution maximale de l'octree  $r_{max}$  (m).  $r_{max}$  représente la « plus petite taille de voxel » et correspond à la longueur d'arête des voxels enfants situés au niveau de profondeur  $d_{max}$  (le plus bas de l'arbre octree). (Fig 2.23-b).

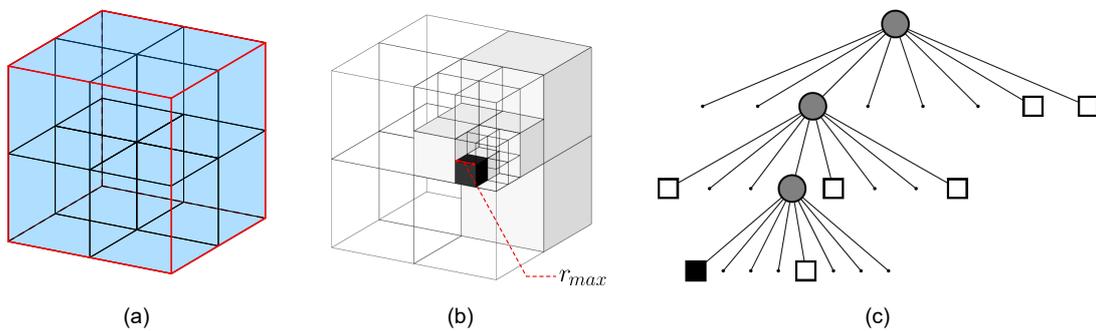


FIGURE 2.23 – Structure d'un Octree : (a) Représentation d'un voxel parent (en rouge) et de ses 8 voxels enfants (en bleus) ; Structure en octree définie avec  $d_{max}=3$ . La structure est représentée sous deux formes différentes : (b) La représentation volumétrique et (c) la représentation sous forme d'arbre.

Le niveau de profondeur d'un voxel dans la structure (depth) est nommé  $d = \{1 \text{ à } d_{max}\}$ . Plus  $d$  augmente, plus la taille du voxel diminue jusqu'à atteindre  $d_{max}$  et une taille de voxel égale à  $r_{max}$ . Pour connaître  $t_{V_d}$ , la taille d'un voxel  $V$  situé au niveau de profondeur  $d$  de l'octree, le calcul peut être généralisé par l'équation (Eq 2.10).

$$t_{V_d} = r_{max} \cdot 2^{(d_{max}-d)} \quad (2.10)$$

Chaque voxel qui compose la grille est représenté par un nœud. Ce nœud contient deux attributs :

1. *key* : un identifiant unique à chaque voxel et associé aux coordonnées ( $X, Y, Z$ ) du point central de l'espace représenté par le voxel. Cette clé permet en général de renseigner également sur le niveau de profondeur  $d$  à laquelle il se situe ;
2. *value* : une valeur stockée (nom, label, valeur numérique, etc.) ;

3. Un voxel correspond à un pixel en 3D d'où son nom issu de l'assemble de volume et pixel.

Les nuages de points pouvant être très denses et composés par des millions de points, l'utilisation de ces structures peut permettre d'optimiser les données. A titre d'exemple, si un grand nombre de points est localisé dans un espace réduit, de volume inférieur à celui représenté par un voxel donné, l'ensemble des points ne seront plus représentés que par une seule entité : un voxel. En paramétrant la résolution  $r_{max}$ , on intervient sur cette optimisation. Plus la résolution  $r_{max}$  paramétrée est faible, plus les données seront optimisées. En contrepartie, les données perdront en niveaux de détails et en représentativité vis-à-vis de la données initiales (Fig 2.24).

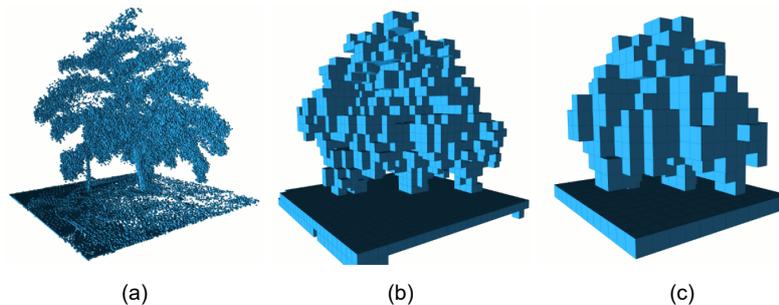


FIGURE 2.24 – Représentation de différents niveaux d'échelles d'une structure octree : Représentation d'un octree représentant un arbre à trois niveaux de profondeur  $d$  différents : (a)  $d = 16$  correspondant à une résolution de  $r = 0.08m$ ; (b)  $d = 13$  correspondant à une résolution  $r = 0.64m$ ; (c)  $d = 16$  correspondant à une résolution de  $r = 1.28m$  [33].

Au-delà de l'optimisation obtenue sur le nombre d'information (nombre de points initial / nombre de voxels), les octrees permettent de structurer les nuages de points qui sont à la base, désorganisés. Les traitements de données sur une zone localisée du nuage sont également optimisés par la structuration apportée par les octrees. Les informations sont obtenues de manières plus efficaces qu'en parcourant linéairement l'ensemble des points qui composent le nuage.

**Quadtree.** Les quadtrees sont une structure utilisée pour partitionner un espace bidimensionnel en le subdivisant récursivement par quatre nœuds [34]. Ils sont une analogie des octrees en deux dimensions et sont principalement utilisés en traitement d'images. Chaque nœud parent permet de subdiviser l'espace qu'il représente en quatre sous-espaces, qui donneront lieux aux quatre nouveaux nœuds enfants dans l'arborescence. Comme pour les octrees, les quadtrees sont définis grâce au paramètre  $d_{max}$  qui représente la profondeur de l'arbre utilisé. Le voxel parent couvre l'ensemble de l'image. Au niveau de profondeur le plus élevé  $d_{max}$ , l'image est décomposée par un nombre de sous-espace  $N = 2^{d_{max}} \cdot 2^{d_{max}}$ .

## 2.3 Traitement des données de perception

Comme évoqué dans l'introduction générale (Sec 2.1), les capteurs permettent de produire des données utiles à la description et la retranscription de l'environnement dans lequel le véhicule évolue. Cependant, pour être capable d'interpréter cet environnement, plusieurs types de traitements (Fig 2.25) doivent être appliqués sur les mesures brutes. Un premier traitement est le géoréférencement des données permettant de les positionner dans l'espace et d'apporter des informations contextuelles. On retrouve aussi différents types d'algorithmes chargés de traiter ces données pour les interpréter et en retirer les informations pertinentes vis-à-vis de la scène perçue. Enfin, lorsqu'un système de perception exploite différentes sources de données, une étape de fusion de données peut également être nécessaire pour associer les informations issues des différentes sources de données.

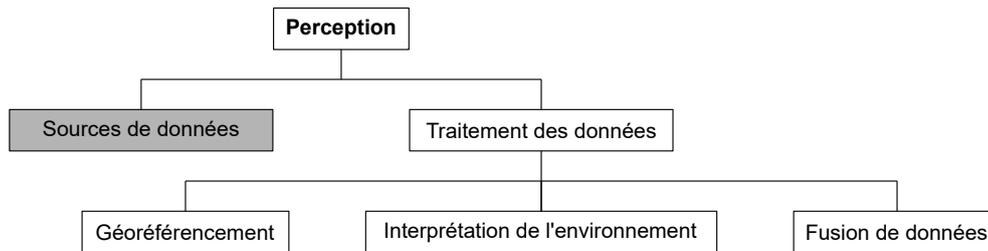


FIGURE 2.25 – Organisation de la section dédiée aux traitements des données : Après avoir étudié les sources de données dans la première partie de ce chapitre, on s'intéresse à présent aux traitements des données répartis en trois sous-catégories : géoréférencement, interprétation de l'environnement et fusion de données.

### 2.3.1 Géoréférencement des données de perception

Avant de traiter les données pour interpréter l'environnement qui entoure le véhicule, il peut être nécessaire de les géoréférencer. Certains algorithmes s'appuient sur le géoréférencement pour traiter les données tels que les algorithmes de map-matching [35]. Dans le cadre de la cartographie, ce géoréférencement représente l'objectif principal car il permet de créer les cartes via l'utilisation des données de localisation ou via l'utilisation d'algorithmes. Différents systèmes types de la littérature et utilisés dans le cadre de la localisation sont représentés (Fig 2.26).

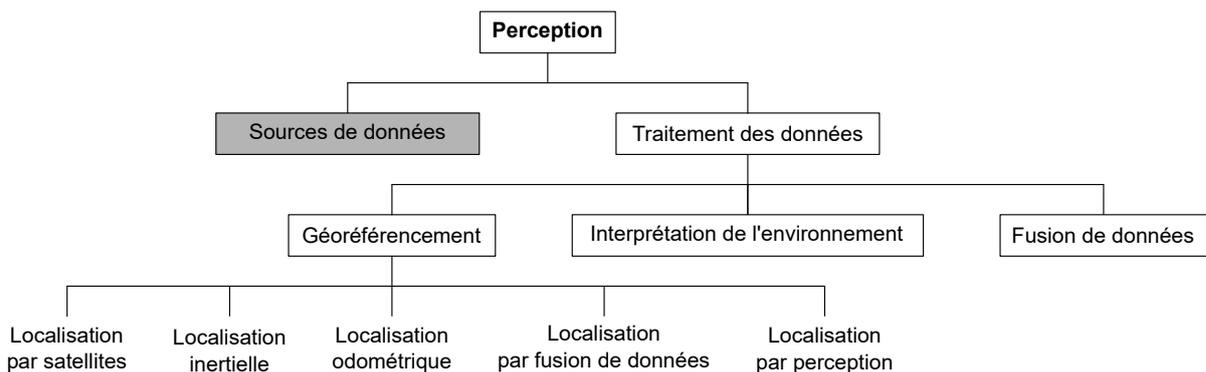


FIGURE 2.26 – Systèmes types pour la localisation

**Remarque :** ce travail n'étant pas centrés sur la localisation, nous présentons ici de façon synthétique les quelques solutions les plus répandues ainsi que leurs avantages et limites auxquelles elles sont sujettes. Davantage de détails sur ces solutions peuvent être retrouvés dans les travaux [36].

#### 2.3.1.1 Méthodes de localisation

**Localisation par satellites.** Le positionnement par satellites, généralement réduit sous le terme GNSS<sup>4</sup>, permet grâce à un récepteur de connaître sa position sur la Terre de façon absolue, à n'importe quel moment et quel que soit l'endroit où l'on se situe. Il est aujourd'hui largement démocratisé grâce aux récepteurs présents dans les véhicules, les téléphones ou encore les montres connectées. Ce type de positionnement exploite le principe de la trilatération [37]. Il consiste à utiliser les distances calculées entre le récepteur et au moins 4 satellites, pour déterminer sa position. Ces distances sont appelées pseudo-distances et sont calculées grâce au décalage temporel observé entre le signal reçu du satellite et ce même signal répliqué par le récepteur. Le décalage correspond au temps nécessaire au signal pour parcourir la distance satellite-récepteur et permet donc de la déduire.

4. Global Navigation Satellite System

Plusieurs phénomènes peuvent impacter la précision de la position calculée en impactant les signaux émis par les satellites [36]. On trouve d'un côté les erreurs atmosphériques, d'éphémérides ou d'horloge qui sont faibles ou pouvant être corrigées. De l'autre les erreurs liées aux masques<sup>5</sup> de l'environnement et aux multi-trajets<sup>6</sup> sont difficiles à identifier et corriger.

La localisation par satellites peut être opérée selon deux méthodes :

1. le positionnement absolu qui exploite le principe de la trilatération directement pour déterminer la position du récepteur ;
2. le positionnement relatif, communément appelé RTK<sup>7</sup>, qui utilise une station de base dont la position est connue avec précision. Cette station transmet au récepteur les corrections des erreurs de trilatération et permet d'améliorer la précision du positionnement absolu.

Ainsi, le positionnement absolu permet de déterminer la position du récepteur avec une précision d'ordre de grandeur 5m lorsque les conditions sont favorables : ciel ouvert, sans masques, ni effets de multi-trajets [38]. En revanche, les corrections reçues en utilisant le positionnement relatif permettent d'obtenir une position de précision centimétrique : +/- 1cm horizontalement et +/- 10cm verticalement.

Le positionnement par satellites propose l'avantage de fournir un positionnement direct et absolu. L'utilisation en relatif permet d'atteindre une précision importante. Cependant, le GNSS ne permet de déterminer qu'une information de position. Pour déterminer l'attitude d'un récepteur, il faut appliquer d'autres moyens de localisation. Les phénomènes de masques ou de multi-trajets pouvant impacter les signaux satellites et donc la précision du positionnement, il est nécessaire de porter une attention aux conditions, ainsi qu'à l'environnement, dans lequel il est exploité. Un dernier inconvénient lié à la localisation par satellites est la fréquence des mesures qui s'étend jusqu'à 10Hz et qui est bien plus faible que celle pouvant être atteinte par d'autres systèmes tels que les centrales inertielles. Par conséquent, ces fréquences de mesures peuvent se montrer insuffisante pour géoréférencer des données issues de capteurs extéroceptifs pouvant utiliser des fréquences d'acquisition égales ou bien supérieures à 20Hz.

**Localisation inertielle.** La localisation inertielle repose sur des mesures fournies par des ensembles d'accéléromètres et de gyroscopes. Ces capteurs sont contenus dans une centrale inertielle, aussi appelée IMU<sup>8</sup>. A la différence de la localisation par satellites qui permet de d'obtenir une position absolue d'un objet, les IMU permettent de mesurer sa position, son attitude<sup>9</sup> ainsi que sa vitesse de déplacement relativement à un point de départ, à une orientation et une vitesse initiale connues. L'utilisation combinée d'accéléromètres et de gyroscopes permet ainsi d'avoir des informations de localisation complètes, appelées « poses » et correspondant à un couple de position et d'attitude associé à l'objet.

Un autre avantage des IMUs est leur totale indépendance vis-à-vis de systèmes extérieurs, à la différence du GNSS avec les satellites. Enfin, un dernier atout est leur fréquence de mesure qui est bien supérieure au GNSS et pouvant atteindre jusqu'à 200Hz (Voir l'exemple de l'Annexe A). Les IMUs souffrent cependant de leur forte dérive dans le temps comme on le retrouve dans les travaux [39].

5. Les masques représentent les obstacles physiques qui se trouvent placés entre le récepteur et les satellites (arbres, bâtiments élevés, tunnels, etc...). Ils peuvent alors affecter au complètement intercepter les signaux satellitaires.

6. Le multi-trajet peut intervenir lorsque des éléments proches du récepteur sont susceptibles de réfléchir ou diffracter le signaux émis par les satellites. Cela a pour effet de rallonger le chemin parcouru par le signal jusqu'au récepteur, le décalage temporel est rallongé, ce qui a pour conséquence d'induire une erreur de positionnement.

7. Real Time Kinematic

8. Inertial Measurement Unit

9. L'attitude représente l'orientation d'un objet dans l'espace selon trois angles : le roulis (*roll*), le tangage (*pitch*) et le lacet (*yaw*).

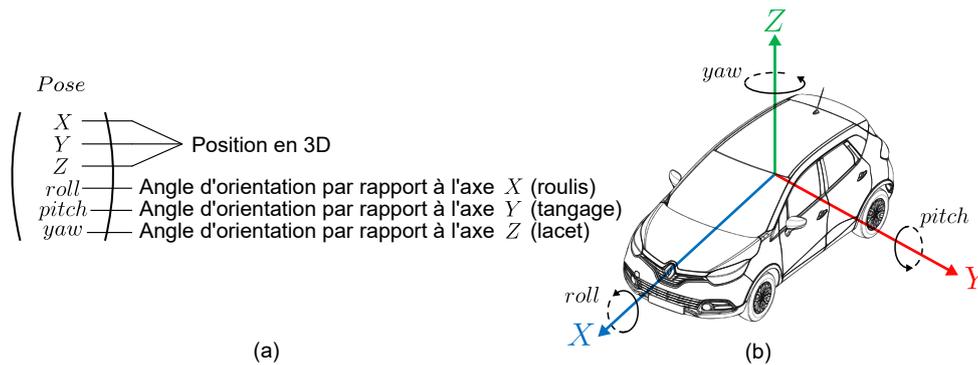


FIGURE 2.27 – Pose d'objet ou de capteur : (a) Présentation des informations minimales composant la pose d'un objet ; (b) Représentation des axes de rotations définis pour calculer les angles d'inclinaison qui composent l'attitude pour un objet véhicule.

A travers ces travaux, il est possible de constater que la précision des données fournies par les IMUs peut grandement varier d'un modèle à l'autre. Cette précision varie en fonction de leur qualité de fabrication et par conséquent, de leur prix comme il l'est démontré dans les travaux [40].

**Localisation odométrique.** L'odométrie consiste à mesurer le déplacement d'une plateforme mobile, à partir d'un point de départ, en fonction des données issues de capteurs : par exemple les odomètres, positionnés sur les roues de cette plateforme. Un odomètre mesure le nombre de tours effectués par une roue grâce à un codeur incrémental. Le déplacement peut alors être déduit en fonction du nombre de tours et des dimensions de la roue. En couplant plusieurs odomètres sur les différentes roues qui composent la plateforme, il est possible de déterminer, en plus de la position, l'orientation relative au point de départ.

De par sa simplicité de mise en œuvre, son faible coût ainsi que sa fréquence de mesure, cette solution est intéressante et répandue. Cependant, les informations obtenues par cette méthode se limitent à une position en 2D (coordonnées  $X$  et  $Y$ ) ainsi qu'à l'angle de tangage. Les informations sont donc moins complètes que pour les méthodes étudiées précédemment. Sa résolution dépend directement de la résolution du codeur occupant les odomètres. A l'image des IMUs, elle est sujette aux dérives en fonction des défauts du terrain où elle est appliquée. Deux types d'erreurs interviennent principalement [41] :

1. Les erreurs systématiques : erreurs mécaniques du système pouvant être corrigées par calibration et erreurs de mesures, non corrigibles, mais qui peuvent être prises en compte dans le modèle cinématique de la plateforme mobile ;
2. Les erreurs aléatoires dépendant du contact roue/sol et qui évoluent en fonction du type de surface sur laquelle la méthode est appliquée.

**Localisation par fusion de données.** Nous avons pu voir que les 3 approches présentées jusqu'ici possèdent leurs propres limitations. L'utilisation combinée de plusieurs approches, sur une même plateforme, peut alors représenter une alternative intéressante pour améliorer la précision et la robustesse de la localisation. En utilisant un filtre de Kalman [42], permettant la fusion des données de localisation de plusieurs sources différentes, l'objectif est de fusionner plusieurs méthodes pour en tirer les forces propres à chacune d'entre elles et obtenir un système de localisation ayant des performances supérieures à chacune des solutions utilisées individuellement. Plusieurs types de fusion peuvent être évoquées comme la fusion odométrie-IMU ou le GNSS-odométrie [36], mais nous nous attarderons ici sur la fusion GNSS-IMU qui offre le plus d'information et qui est très répandue [43].

En effet, de par ce couplage, le GNSS nous permet d'avoir un positionnement absolu grâce à des coordonnées mesurées dans un repère global, d'un côté, de l'autre, l'IMU complète ces informations avec l'attitude et les vitesses de déplacement. On a donc un positionnement et une orientation en 3 dimensions. Différents couplages sont possibles [36], un choix approprié permet de corriger les estimations de positions, de vitesse et d'attitude, et ainsi réduire la dérive dans le temps. Pour le GNSS, cela permet aussi de détecter les multi-trajets. La détermination et la correction permet, dans un premier temps, de grandement améliorer la précision du positionnement. Dans un second temps, ce couplage permet au GNSS de profiter des fréquences d'acquisition des IMUs, bien plus élevées. L'IMU est ainsi utilisée pour prendre le relais entre deux mesures GNSS. En estimant le déplacement relatif à une première mesure GNSS, jusqu'à disposer d'une nouvelle position GNSS, on peut alors calculer des poses absolues, de précision centimétrique et produites à très haute fréquence.

**Localisation par perception.** Une dernière méthode de localisation répandue repose sur l'utilisation des capteurs extéroceptifs et l'estimation de trajectoire en se basant sur les données de ces capteurs et par rapport à un repère initial. Les approches existantes peuvent être regroupées selon trois familles.

**Le SLAM<sup>10</sup>** : la cartographie et localisation simultanée consiste à créer une carte de l'environnement incrémentalement à partir des d'une position de départ et de données de perception. Cette carte est ensuite utilisée pour la localisation et les nouvelles mesures permettent de la mettre à jour. Cette méthode est applicable sur des capteurs imageurs, mais également sur des capteurs type LiDAR. Cette méthode présente néanmoins des inconvénients liés au temps de traitement nécessaire pour construire une carte de grande ampleur.

**La reconstruction 3D** : aussi appelée SfM<sup>11</sup>, il s'agit d'une approche visant à construire une carte 3D de l'environnement grâce à l'exploitation de la trajectoire d'une caméra en mouvement. La reconstruction, effectuée en post-traitement, est faite à partir d'une collection d'images. Le principal inconvénient de cette approche est que la reconstruction à grande échelle reste encore difficile à appliquer [44].

**L'odométrie visuelle** : comme pour la localisation odométrique, l'objectif de cette approche est d'estimer la trajectoire d'une plateforme mobile relativement à un point de départ. L'odomètre est ici remplacé par des capteurs de vision. L'idée générale est de détecter les similitudes entre plusieurs images successives issues d'une même caméra. L'étude des similitudes permet par la suite de déduire le déplacement effectué par la caméra entre les prises de vues et indirectement les déplacements de la plateforme à laquelle elle est rattachée [36].

**Synthèse** : ces différentes méthodes cherchent à répondre à un objectif commun : déterminer la trajectoire de façon relative à un point de départ. On peut remarquer que les deux premières approches cherchent à répondre à un objectif proche : reconstruire une carte 3D d'un environnement inconnu en estimant la trajectoire du capteur utilisé et afin de localiser ce dernier au sein de la carte créée. Cela rend les approches de plus en plus proches et similaires. Cependant, certaines limitations peuvent être observées du point de vue calculatoire ou des échelles auxquelles elles peuvent être appliquées. Ces approches restent incrémentales et peuvent souffrir de dérives dans le temps.

---

10. Simultaneous Localization And Mapping

11. Structure from Motion

L'utilisation de calcul de fermeture de boucle peut néanmoins permettre de corriger et d'optimiser dans une certaine mesure la dérive. Une autre contrainte est que ces approches nécessitent d'avoir un nombre d'amers<sup>12</sup> important pour obtenir une mise en correspondance, automatique et précise, des données. L'utilisation de LiDAR peut cependant résoudre partiellement ce problème.

### 2.3.1.2 Référentiels

On a pu le voir jusqu'ici qu'il existe différents types de capteurs extéroceptifs et différents systèmes de localisation. Une même plateforme peut être équipée par plusieurs de ces capteurs et systèmes, or, chacun possède son propre référentiel. On va donc distinguer trois types de référentiel différents :

**Le référentiel capteur.** C'est le premier type de référentiel, il est propre à chaque capteur. Il s'agit ici du référentiel local et relatif au capteur auquel il correspond (Fig 2.28-a et b). L'origine du repère attaché à ce type de référentiel est en général centrée sur le centre du capteur. En revanche, des variations quant à l'orientation du repère existent en fonction des capteurs. Dans la suite du manuscrit on nommera  $\mathcal{L}$  le référentiel propre à un LiDAR, et  $\mathcal{C}$  le référentiel propre à une caméra.

**Le référentiel *Ego*.** Il correspond au référentiel généralement exploité lorsqu'une plateforme embarque différentes sources de données. Une fois les paramètres de calibrations extrinsèques définis pour l'ensemble des capteurs embarqués, l'objectif est de réunir toutes les données produites dans un seul et unique référentiel. *Ego* est généralement exploité pour cet objectif. Il s'agit d'un référentiel local et relatif à la plateforme (Fig 2.28-c). Son origine est en général matérialisée par le centre de l'essieu arrière du véhicule, avec l'axe  $X$  orienté vers l'avant, l'axe  $Y$  vers la gauche et l'axe  $Z$  vers le haut. L'unité de localisation est également calibrée et paramétrée pour que les mesures de positionnement correspondent à l'origine de ce repère.

**Le système de référence.** C'est le troisième type de référentiel. Il correspond aux systèmes géodésiques  $\mathcal{G}$  (Fig 2.28-d). Ce sont des systèmes de référence qui permettent de positionner un objet sur la Terre, en lui associant des coordonnées d'un repère lié à la Terre et qui sont définis par des points fondamentaux<sup>13</sup>. On considère deux catégories :

1. Les systèmes de coordonnées géographiques dont les coordonnées sont exprimées en valeurs angulaires : longitude  $\lambda$  et latitude  $\varphi$  ;
2. Les systèmes de coordonnées projetées qui projettent les coordonnées géographiques sur une surface plane. Cela permet d'obtenir des coordonnées cartésiennes  $X$  et  $Y$ . Concernant la troisième dimension, l'altitude  $Z$  est obtenue en se basant sur le niveau moyen des mers [45].

### 2.3.1.3 Transformations des données

L'objectif du géoréférencement étant de projeter l'ensemble des données produites lors de l'acquisition dans un seul et même référentiel, il est nécessaire d'appliquer des transformations. Là encore, on distingue deux types de transformations.

12. Les amers correspondent à des éléments de l'environnement suffisamment distinguables pour que, lorsqu'ils sont retranscrits à travers des données de perception (ensembles de pixels ou ensembles de points), ils puissent servir de point d'intérêts sur lesquels l'estimation du déplacement est calculée.

13. Points de références permettant de définir un système de coordonnées tel que le méridien de Greenwich, positionné sur son observatoire et où la longitude est définie comme égale à  $0^\circ$ , servant ainsi de référence au système géodésique mondial WGS-84 (World Geodetic System 1984).

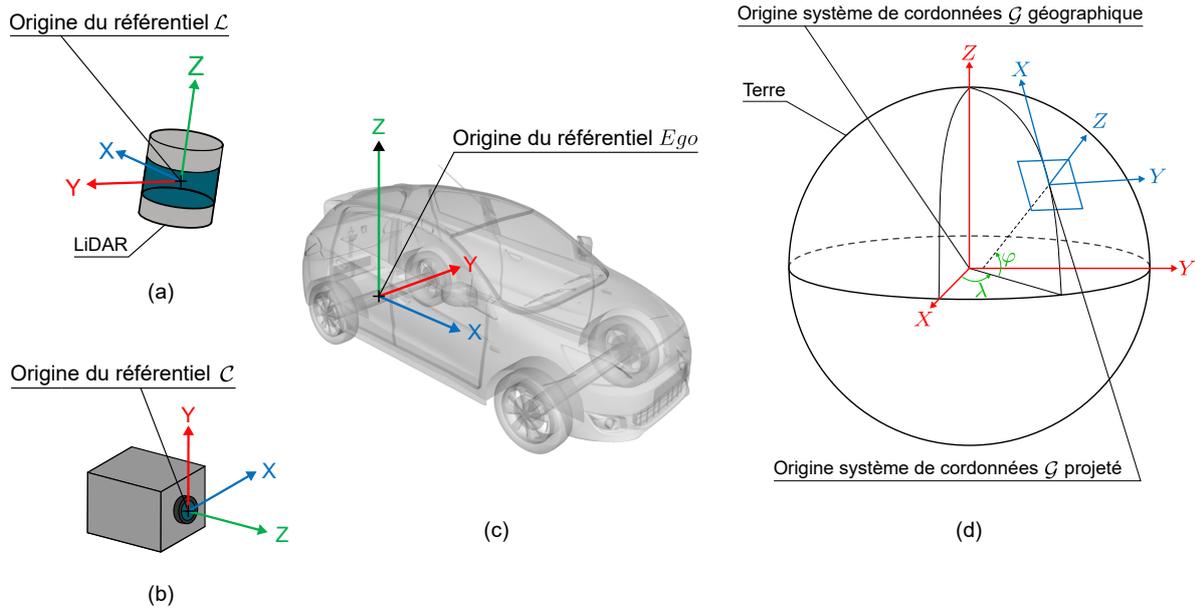


FIGURE 2.28 – Représentation des différents référentiels généralement exploités dans le cadre de la perception : (a) et (b) représentent respectivement le référentiel  $\mathcal{L}$  propre au capteur LiDAR et  $\mathcal{C}$ , le référentiel propre aux caméras ; (c) représente le référentiel  $Ego$  ; (d) est une représentation du référentiel global  $\mathcal{G}$  dans le cadre d'un système de coordonnées géographiques (en vert), ou pour un système de coordonnées projeté (en bleu) et dans lesquels les données sont géoréférencées de façon absolue. Les axes en rouges représentent les axes du système de coordonnées ECEF (Earth Centered Earth Fixed).

**Les projections.** Les mesures de positionnement par GNSS<sup>14</sup> permettent d'obtenir des coordonnées géographiques dans le système WGS-84. Pour simplifier le format de données et faciliter les traitements, ces mesures sont projetées pour obtenir des coordonnées cartésiennes. Les projections sont alors exploitées pour projeter des coordonnées WGS-84 vers des projections de référence, précisément définies, telles que l'UTM (Transverse universelle de Mercator) [46].

**Les transformations rigides.** Elles sont exploitées pour projeter des mesures d'un référentiel euclidien  $A$  à un référentiel euclidien  $B$ . Ce type de transformation se compose d'un couple de translations et de rotations en trois dimensions (Fig 2.29), détaillé dans (Annexe B).

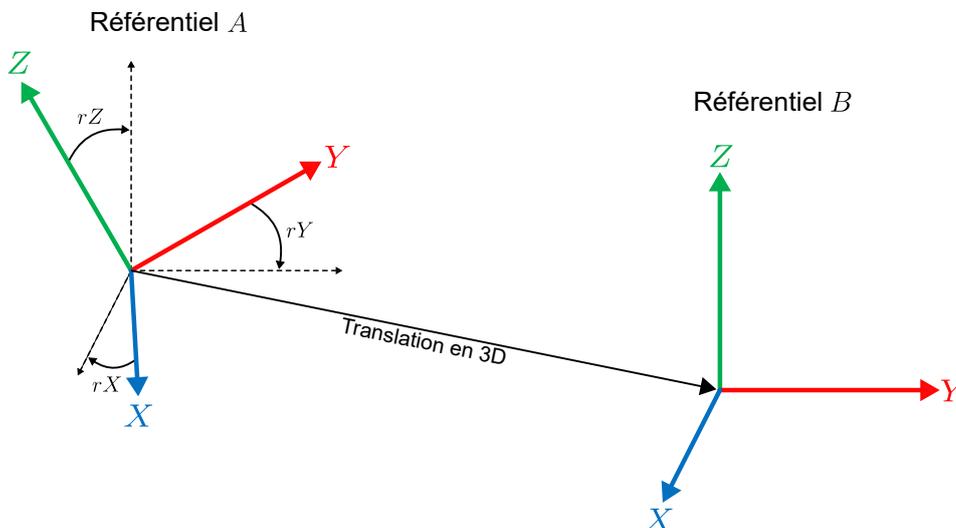


FIGURE 2.29 – Représentation d'une transformation rigide.

14. Le GNSS, pour Global Navigation Satellite Systems, correspond à un système de positionnement par satellites.

Dans le cadre des systèmes de perception embarqués, les transformations rigides sont appliquées, dans un premier temps, pour réunir les données de l'ensemble des capteurs embarqués dans un seul et même référentiel : *Ego*. Une première transformation est déterminée pour chaque capteur afin de projeter les mesures propres aux capteurs de leur référentiel local vers le référentiel *Ego*. Un exemple de cette transformation pour un capteur LiDAR est représenté (Fig 2.30-a). Cette transformation est déterminée par un processus de calibration extrinsèque pour chaque capteur. Notre méthodologie de calibration est détaillée Chapitre 3. Les paramètres de ces transformations sont fixes et restent identiques jusqu'à ce que la configuration de la plateforme évolue : changement d'orientation ou de position des capteurs.

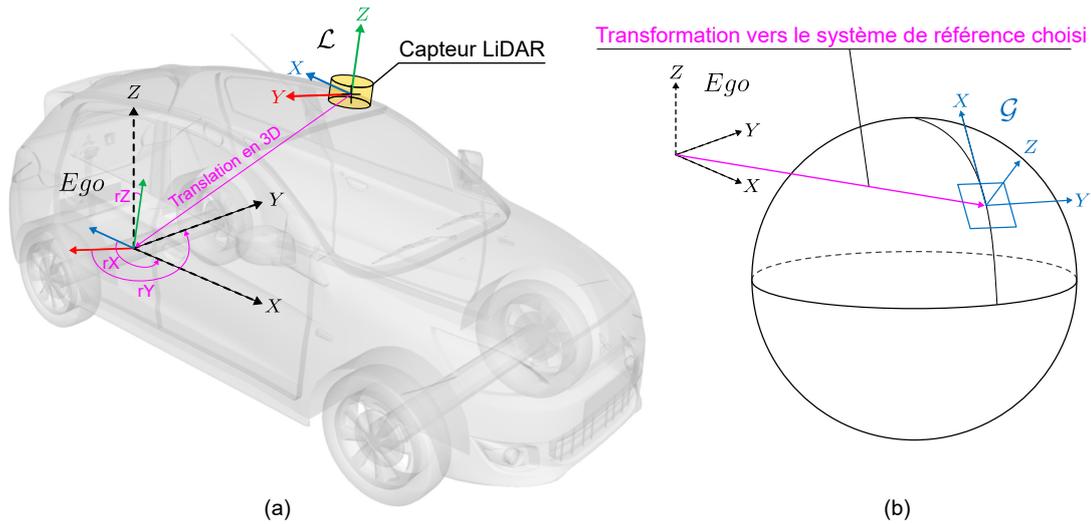


FIGURE 2.30 – Représentation des différentes transformations utilisées : (a) Transformation pour projeter les mesures du capteur LiDAR de leur référentiel local  $\mathcal{L}$  vers le référentiel *Ego* ; (b) Transformation pour projeter le référentiel *Ego* vers un système de référence connu  $\mathcal{G}$ .

Une fois les coordonnées mesurées par le système de localisation projetées en Cartésien, il est possible de calculer une seconde transformation rigide entre *Ego* et  $\mathcal{G}$ , afin de les géoréférencer de façon absolue, dans le système de référence choisi (Fig 2.30-b). Les données de chaque capteur ayant déjà été référencées dans le référentiel *Ego* par la première transformation, il suffit d'appliquer cette seconde transformation sur l'ensemble des données pour qu'elles soient à leur tour géoréférencées dans  $\mathcal{G}$ .

### 2.3.2 Algorithmes d'interprétation de l'environnement

Les algorithmes présents dans la littérature des données des différents types de capteurs introduits dans la (Sec 2.2). Les résultats des algorithmes de traitements peuvent être représentés sous différentes formes comme nous pouvons le voir avec quelques exemples (Fig 2.31).

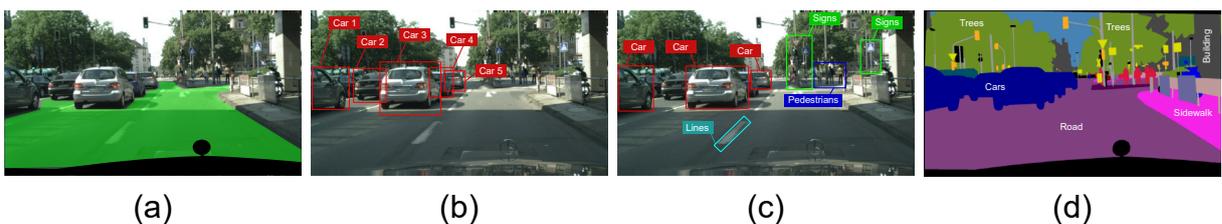


FIGURE 2.31 – Exemples de résultats d'algorithmes de perception [14] : (a) Extraction du sol ; (b) Algorithme de détection d'objets ; (c) Algorithme de classification sémantique et (d) Algorithme de segmentation sémantique.

Parmi les différents algorithmes liés à la perception, on s'intéresse ici aux plus utilisés. Ces algorithmes de traitements peuvent être appliqués sur des nuages de points 3D, des images 2D. On les retrouve généralement dans les outils de benchmark tels que le KITTI Vision Benchmark Suite [4], où ils sont évalués. Nous avons fait le choix de les structurer selon quatre catégories illustrées (Fig 2.32).

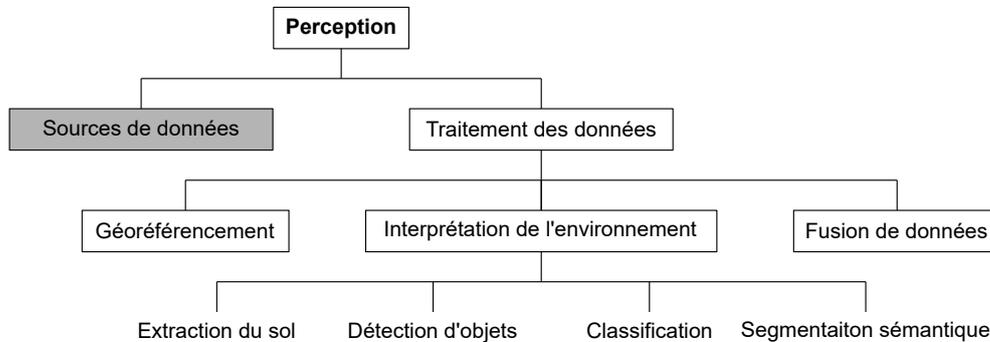


FIGURE 2.32 – Organisation de la section dédiée aux algorithmes de traitement

### 2.3.2.1 Algorithmes d'extraction du sol

Ce premier type d'algorithme a pour rôle d'identifier les mesures correspondant à la structure de la route et de les étiqueter comme tel. Cette dernière est communément associée au nom « sol » dans la littérature (ground detection) et peut englober la route ainsi que les trottoirs. Le sol est l'élément structurant de la scène et permet de relier entre eux les différents objets et éléments de structures qui occupent la scène perçue par le véhicule. Inclure une première étape dédiée à cet élément dans la chaîne de traitement offre différents avantages qui sont discutés dans les paragraphes suivants.

Cet élément fait l'objet de traitements ciblés car il possède des caractéristiques spécifiques qui permettent de le distinguer plus aisément que les autres objets et structures composants l'environnement. Dans de nombreuses situations, le sol est l'élément le plus représenté quantitativement dans les données. Lorsqu'il s'agit de route ou de trottoir, il possède également des caractéristiques homogènes telles que la couleur ou la forme. Il peut être assimilé à une surface plane, proche de l'horizontale, en fonction de la pente de la route. A contrario, la majorité des objets présents dans l'environnement ont des formes variables et irrégulières. Une grande partie d'entre eux s'orientent verticalement (Fig 2.33).

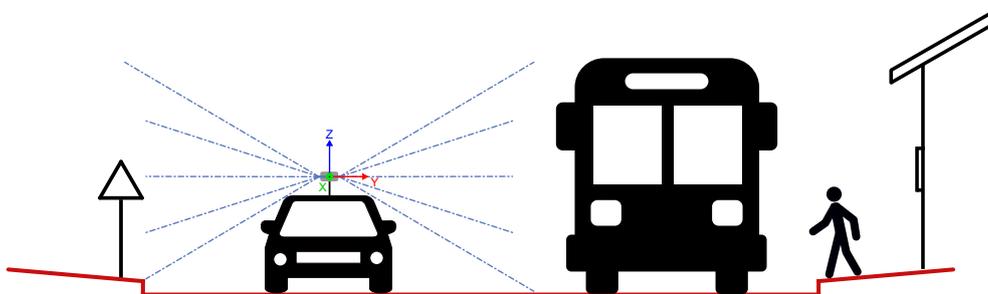


FIGURE 2.33 – Représentation de la distinction entre le sol et les objets : Distinction entre le sol (représenté en rouge) et les objets (représentés en noir).

L'extraction des données du sol a plusieurs applications : tout d'abord, il permet de donner des informations au véhicule sur la zone de « free-space<sup>15</sup> », lui indiquant où il peut évoluer sans danger (Fig 2.34).

15. Les zones free-space correspondent aux zones navigables et non occupées en tenant compte des contraintes imposées

Il est également possible de simplifier certains traitements de données spécifiques à cet élément structurel tels que l'identification des marquages. En identifiant le sol, cela permet de délimiter la voie dans laquelle le véhicule circule ainsi que d'identifier les voies adjacentes, ce qui contribue à définir cette zone de « free-space ».

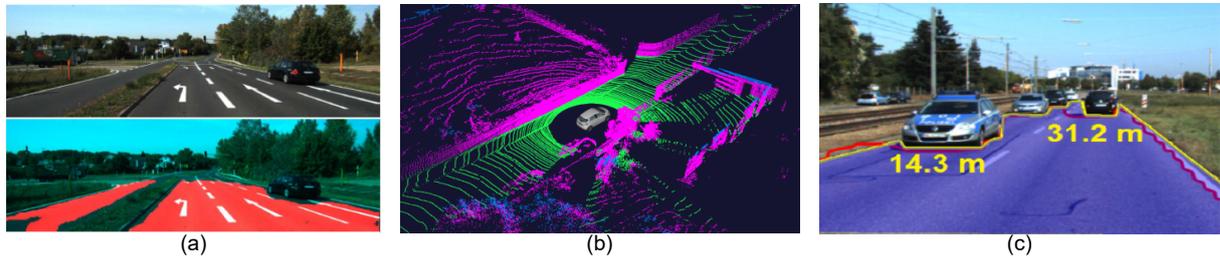


FIGURE 2.34 – Illustration des résultats d'algorithmes de détection du sol : (a) Détection de la route basée sur la vision monoculaire [47]; (b) Détection du sol basée sur des données LiDAR [48]; (c) Algorithme d'identification de zone « free-space » [49].

Cela permet également d'améliorer et de faciliter les tâches des autres algorithmes (détection d'objets, segmentation des objets,...) pour différentes raisons. Les données identifiées comme appartenant au sol sont exclues des traitements, ce qui permet de les accélérer en réduisant considérablement le volume d'information pris en compte. Le sol étant l'élément qui lie les différents objets de l'environnement, en soustrayant cet élément au reste des données, les objets peuvent être plus facilement distingués les uns des autres. Les objets sont ainsi plus faciles à identifier et à segmenter via les autres algorithmes de traitement présentés.

### 2.3.2.2 Algorithmes de détection d'objets

Les algorithmes de détection sont restreints à un ensemble de classes sémantiques (panneaux de signalisation, véhicules, piétons, etc). Ils sont utilisés pour détecter si au sein d'une trame de données, une ou plusieurs instances d'objets, correspondant à la classe sémantique ciblée, sont identifiées. Ces détections permettent d'établir si des objets sont présents dans l'environnement du véhicule et peuvent représenter un potentiel obstacle à éviter pour le véhicule.

En fonction des algorithmes, les objets détectés dans les données traitées peuvent être modélisés sous différentes entités géométriques : des boîtes englobantes 2D ou 3D, en fonction du type de données exploitées (Fig 2.34). A partir de ces boîtes englobantes, différents paramètres décrivant ces objets détectés peuvent être estimés tels que leurs dimensions (hauteur, longueur, largeur,...). L'analyse temporelle des boîtes englobantes entre des trames successives peut également servir à inférer si un objet est statique ou en mouvement. Ces boîtes sont définies par une liste de coordonnées représentant les sommets ce qui permet de les re-projeter par la suite dans l'espace de représentation des données.

Les données traitées peuvent également avoir d'autres représentations géométriques pour certaines classes d'objets. Si un élément ponctuel peut facilement être délimité par une boîte englobante, pour certains éléments linéaires tels que les marquages au sol, cela n'est pas pertinent. A la place, ces éléments détectés peuvent être représentés par d'autres modèles. Dans le cas des marquages au sol discontinus séparant les voies, elles peuvent être représentées sous la forme de courbes. On trouve notamment des représentations sous formes de polynômes ou de splines. C'est notamment le cas dans l'article [52].

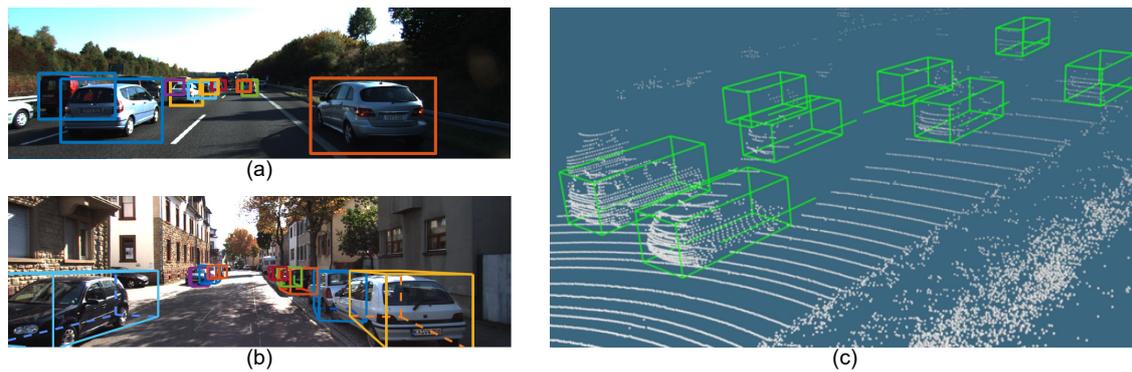


FIGURE 2.35 – Exemples d'algorithmes de détection d'objets : Après traitement, les boîtes englobantes des objets détectés sont projetées sur les données traitées. Cela permet de visualiser les détections obtenues pour un type d'objet (ici des véhicules). (a) Algorithme basé vision avec identification par boîtes englobantes 2D ; (b) Même algorithme avec identification par boîtes englobantes 3D ; (c) Algorithme basé sur des données LiDAR avec identification des objets par boîtes englobantes 3D [50][51].

Les données qui n'appartiennent pas à une classe ciblée par l'algorithme de détection ne sont pas prises en compte dans les résultats en sortie de l'algorithme mais peuvent cependant être utilisées par d'autres traitements dans l'objectif de détecter des objets appartenant à d'autres classes.

### 2.3.2.3 Algorithmes de classification

Ces approches sont exploités pour classer sémantiquement les différents éléments qui composent les données. On distingue deux types de classification : la classification binaire ou la classification multi-classes. La classification binaire se rapproche des algorithmes de détections dans la mesure où une unique classe sémantique est ciblée. Les données sont alors séparées en deux classes : les éléments qui correspondent à la classe et ceux qui ne font pas partie de la classe.. A titre d'exemple, dans le cadre d'un algorithme de classification orienté sur la classe sémantique « véhicule », les données traitées pourront être associées à deux classes : « véhicule » ou « non-véhicule ». Dans le cadre, des classifications multi-classes, au moins trois classes sémantiques différentes sont considérées lors de la classification des données comme dans [53]. Les algorithmes sont alors plus complexes et les données peuvent être étiquetées avec les différentes classes sémantiques prises en compte.

Les algorithmes de classification sont couramment combinés avec des algorithmes de segmentation [54], qui interviennent en amont. Une des approches que l'on trouve dans la littérature consiste à exploiter la distance euclidienne au sein de groupes de points. Concernant les images, des méthodes exploitées par la vision par ordinateur peuvent être utilisées pour former des groupes de pixels. On retrouve par exemple les composantes connexes [55, 56]. Les données sont alors regroupées par segments, représentant les objets de l'environnement et qui sont plus facilement distinguables les uns des autres. Les groupes qui ne contiennent pas suffisamment de pixels ou de points, ainsi que les pixels et points isolés ne sont pas pris en compte dans la suite des traitements. La dernière étape est ainsi consacrée à l'analyse des segments pour les classifier et leur attribuer une classe.



FIGURE 2.36 – Représentation des étapes de traitements jusqu'à la classification des données.

La segmentation des données consiste donc à les regrouper selon des caractéristiques. Celles-ci peuvent être étudiées au stade précédant la classification. Celles-ci peuvent correspondre à des critères géométriques simples (dimensions, surface, volume...). D'autres critères mathématiques calculés sont également

exploitables tels que la normale aux points pour les nuages de points 3D ou la régularité des contours des segments. Enfin, on peut également étudier les valeurs d'intensité ou de couleurs en fonction des données traitées. On peut diviser les méthodes de classifications selon deux niveaux de dépendance : supervisée et non-supervisée.

- Classification supervisée : on sélectionne un ensemble de données référencées et de même nature que celles que l'on souhaite classifier par la suite. L'algorithme est entraîné sur cet ensemble et calcule les paramètres statistiques et caractéristiques permettant d'identifier les différentes classes rencontrées dans le jeu d'entraînement. A l'issue de cet entraînement, un modèle de classification est créé et peut être appliqué sur de nouvelles données non-référencées.
- Classification non-supervisée : aucune donnée référencée au préalable n'est exploitée. L'algorithme s'entraîne sur un jeu de données non labellisées et produit de manière automatique le modèle permettant d'attribuer les différentes classes sémantiques aux données traitées. Il définit lui-même les caractéristiques sur lesquelles se baser ainsi que le nombre de classes identifiées au sein du jeu de données fournis. Là encore, le modèle de classification sera appliqué sur de nouvelles données pour les classifier.

Dans les deux cas, les algorithmes entraînés se basent essentiellement sur le domaine de l'apprentissage automatique (machine learning). Les modèles de classifications peuvent ainsi être de plusieurs types : séparateurs à vaste marge (S.V.M), arbres de décisions ou réseaux de neurones.

#### 2.3.2.4 Segmentation sémantique

Enfin, depuis quelques années, de nombreux travaux se sont orientés vers les algorithmes de segmentation sémantique. Ces algorithmes se basent sur le domaine du machine learning pour traiter les trames de données dans leur globalité. On parle ainsi de prédiction dense car tous les éléments composant les données (pixels ou points) sont pris en compte et labellisés à la fin du processus. Ces algorithmes de traitement regroupent deux étapes. D'abord, ces éléments sont regroupés sous la forme de segments : région de pixels pour les images, ou groupe de points pour les nuages de points. Dans un second temps, les données sont traitées segment par segment et sont classées sémantiquement. Cette classification s'appuie sur l'utilisation de réseau de neurones entraînés préalablement sur des jeux de données annotés. Les caractéristiques permettant d'identifier la classe sémantique d'un segment donné sont analysées par ces réseaux lors de l'entraînement, avant d'être appliquées dans les phases de prédiction.

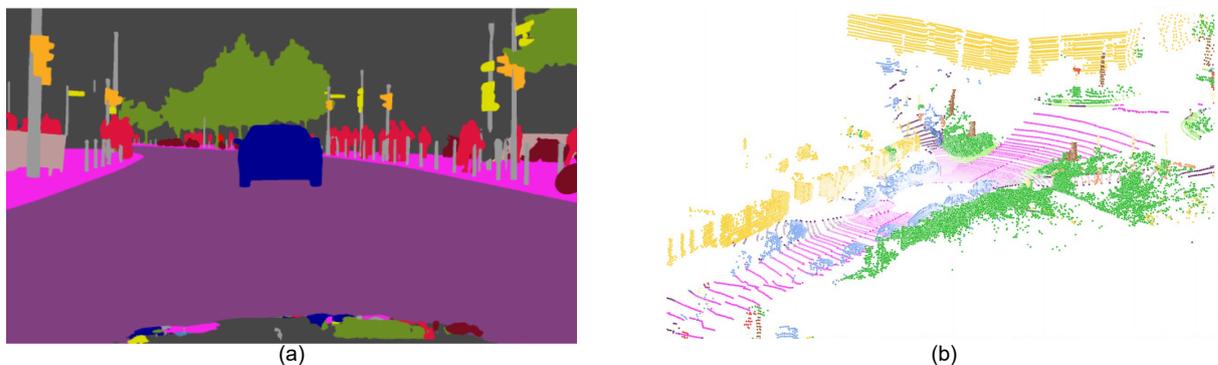


FIGURE 2.37 – Exemples de résultats obtenus à partir d'algorithmes de segmentation sémantique : a) basé vision [57] et b) basé laser [58]. Les différentes couleurs représentent les différentes classes sémantiques assignées par l'algorithme.

Après avoir appliqué ce type d'algorithme, l'ensemble des éléments composant les données possèdent un label sémantique. On distingue deux types de résultats. Le premier est le moins détaillé, les éléments sont labellisés sémantiquement mais ne sont pas rattachés à une instance. Pour le second type de résultats, les éléments sont labellisés mais également identifiés par un indice d'instance, ce qui permet de différencier les différents objets de la scène.

### 2.3.3 Fusion de données

Comme nous avons pu le voir avec les spécificités des différents capteurs, aucun ne permet de satisfaire toutes les exigences requises par la fonction de perception. Pour parer à ces lacunes, la solution actuelle consiste à combiner plusieurs de ces capteurs. Dans la littérature, on peut aussi identifier les approches ayant un TRL<sup>16</sup> industriel avancé. On retrouve principalement le couple radar/caméra qui représente les solutions les plus matures et les moins chères à embarquer sur un véhicule de série. A ce couple peut s'ajouter le capteur LiDAR qui est, pour l'instant, réservé à des véhicules haut-de-gamme (par exemple avec l'Audi A8).

Le rôle des algorithmes de fusion est d'utiliser conjointement les informations obtenues à partir des différents capteurs qui équipent un véhicule pour analyser et représenter de façon plus complète l'environnement [59]. Ils exploitent des données issues des algorithmes de perception, mais peuvent également inclure des informations liées à la localisation du véhicule [60, 61]. L'exploitation des données de perception, associées à l'utilisation d'une carte embarquée permet d'améliorer la localisation du véhicule au sein de cette même carte [35]. Cela permet de disposer d'une nouvelle source contextuelle pour l'interprétation de l'environnement rencontré grâce aux informations contenues dans la carte.

Dans le cadre de la fusion, certains conflits entre les sources de données peuvent apparaître [62]. A titre d'exemple, avec l'utilisation d'un couple radar/caméra, un conflit apparaît si l'un des deux capteurs détecte un objet et que l'autre ne le détecte pas. Ce type d'algorithmes doit alors permettre d'évaluer la pertinence des informations reçues et de définir la ou les sources de données les plus fiables et sur lesquelles s'appuyer pour l'interprétation « finale » de l'environnement. A l'inverse, lorsque plusieurs sources de données s'accordent sur la détection d'un objet, cette fusion de données permet de garantir avec plus de certitudes la présence cet élément dans l'environnement.

## 2.4 Validation de la perception

Nous avons vu au cours de ce chapitre que les solutions de perception reposent sur différents types de capteurs, eux-mêmes associés à différents types d'algorithmes de perception. Le rôle de ces solutions est de percevoir et interpréter l'environnement du véhicule, afin de rendre ses déplacements plus fiables. Étant donné ce rôle crucial que joue la perception, il est nécessaire d'évaluer les solutions de perception lors de tests avant de les exploiter en conditions réelles d'usage. Dans ce but, la validation de solutions de perception cherche à répondre à différents objectifs.

Le premier est d'évaluer les performances d'une solution testée. Cela peut être fait en comparant les données obtenues en sortie de la solution, à des données de référence. Les données de références (pouvant être simplifiées sous le terme « la référence ») correspondent à des données de perception augmentées par différentes informations (sémantiques ou autres).

---

16. « Technology Readiness Level », qui peut se traduire niveau de maturité technologique et qui est associé à une technologie pouvant être mise en application directement.

Les comparaisons menées entre la référence et les données issues de la solution testée, ou de l'algorithme évalué, permettent ainsi de calculer différents indicateurs de performances suivant les critères souhaités.

D'autres objectifs peuvent être menés par la validation des solutions de perception. Pour une solution déjà évaluée, la validation peut également servir à mesurer les progrès réalisés lorsqu'elle évolue (par exemple avec la mise à jour des algorithmes de perception). De la même façon, la phase de test et de validation peut permettre de vérifier la robustesse d'une solution à travers différents scénarios : environnement urbain, rural ou autoroutier ; environnements composés d'objets dynamiques ou exclusivement statiques ; conditions météorologiques favorables ou défavorables ; conditions d'éclairage et de luminosité variables (jour et nuit). La robustesse d'une solution peut également être évaluée en comparant les résultats avec des variations de la configuration : modèles de capteurs, positions et orientations des capteurs. Cela peut permettre d'identifier une configuration pour laquelle un algorithme obtient de meilleurs résultats.

Si jusqu'en 2012 les critères d'évaluation de la littérature scientifique pouvaient varier d'un travail à l'autre [63], l'arrivée d'outils de benchmark tels que le KITTI Vision Benchmark [4] ou Cityscapes [14], a fortement contribué à homogénéiser les critères d'évaluation utilisés. En proposant des critères spécifiques en fonction des différents types d'algorithme évalués, cela a permis d'orienter les chercheurs vers l'utilisation de ces critères pour évaluer les performances de leurs travaux. L'utilisation de critères identiques permet de faciliter la comparaison des performances atteintes par les différentes solutions. Cela permet également aux chercheurs utilisant ces mêmes critères de positionner leur solution vis-à-vis de l'état de l'art existant.

Dans le cadre industriel, les constructeurs automobiles peuvent faire appels à des équipementiers pour intégrer directement leurs solutions de perception sur les véhicules de série. Une grande partie des solutions proposées par ces équipementiers intègrent leurs propres algorithmes de traitement des données et ne proposent en sortie que des données filtrées et traitées. Le détail des algorithmes utilisés est mal connu et ces solutions peuvent être assimilées à des boîtes noires. Dans ce cas, seules les fiches techniques peuvent annoncer des indicateurs de performance. La validation peut alors être utilisée pour vérifier que les performances réalisées lors de tests en conditions réelles correspondent bien à celles annoncées dans les fiches techniques.

Enfin, à travers ces différents objectifs annoncés, la validation peut permettre de garantir, pour une solution donnée, un niveau de performance qui la caractérise en fonction de scénarios spécifiques testés et validés. Elle peut être décomposée en deux éléments distincts qui seront développés dans la suite de cette thèse. D'une part on trouve les données de référence utilisées pour évaluer une solution. D'autre part, la méthode de validation comprend la méthode de comparaison entre les données testées et les données de référence, ainsi que les critères de performance utilisés pour évaluer une solution.

## 2.5 Conclusion

A travers ce chapitre, nous avons dressé un état de l'art des systèmes de perception embarqués sur des véhicules ou autres plateformes terrestres. Le domaine de la perception est un sujet de recherche actif depuis de nombreuses années. Il continue d'évoluer avec de nouvelles technologies de capteurs arrivant à maturité, mais également, avec de nouvelles approches algorithmiques, comme on peut le voir avec le domaine du machine learning qui est très actif.

Il ressort de cette étude que :

- La perception, pour une plateforme mobile, peut être réalisée par différents types de capteurs et que les données brutes produites ne permettent pas de disposer d'informations compréhensibles sur l'environnement.
- Les données brutes mesurées par les capteurs ne permettent pas de disposer d'informations pertinentes sur l'environnement dans lequel la plateforme se situe. C'est pourquoi ces capteurs doivent être couplés à des algorithmes de traitement chargés d'extraire automatiquement des informations compréhensibles et permettant d'apporter les informations nécessaires pour connaître l'environnement. Comme pour les capteurs, ces algorithmes se basent sur des approches de traitement qui peuvent être très différentes et qu'il est nécessaire d'évaluer ou de comparer pour pousser vers l'amélioration future de ces approches ;
- L'étude des capteurs démontre qu'il n'existe pas de capteur parfait et qu'il est nécessaire d'équiper les véhicules avec différents types de capteurs complémentaires les uns par rapport aux autres ;
- Pour garantir le bon fonctionnement des systèmes ADAS, les solutions de perception doivent être évaluées au cours de tests réalisés avant de les utiliser dans le cadre d'un usage réel, non expérimental. Ces tests permettent de répondre à différents objectifs. Le premier est l'évaluation des performances d'une solution donnée. Cependant, ils peuvent également mesurer les progrès réalisés pour une solution après une mise à jour des algorithmes de traitement ou après avoir fait évoluer la configuration des capteurs utilisés par cette solution ;
- Enfin, au cours des années 2010, l'arrivée d'outils de benchmarks a permis de faciliter les comparaisons des performances entre diverses solutions de perception. Ils ont également joué un rôle prépondérant dans l'homogénéisation des critères d'évaluation utilisés pour mesurer les performances des solutions testées. Cela a contribué davantage à simplifier les comparaisons inter-solutions en s'appuyant sur des mêmes critères et permettant aux chercheurs de positionner leur solution vis-à-vis de l'état de l'art.

Pour mesurer les performances d'une solution de perception donnée, deux éléments sont nécessaires : la référence à laquelle cette solution sera confrontée et la méthode de comparaison (incluant les métriques mesurées). Dans l'objectif de proposer une nouvelle approche de validation, répondant à la problématique de ce travail, le chapitre suivant se concentrera sur le premier élément nécessaire grâce à la conception d'une méthode de production des données de référence.

## Chapitre 3

# Conception de la méthode de référencement

### Sommaire

---

|       |  |    |
|-------|--|----|
| 3.1   | Introduction . . . . .                           | 41 |
| 3.2   | Méthodologies de référencement . . . . .         | 42 |
| 3.2.1 | Présentation des approches existantes . . . . .  | 43 |
| 3.2.2 | Synthèse comparative . . . . .                   | 47 |
| 3.3   | Concept méthodologique proposé . . . . .         | 48 |
| 3.3.1 | Analyse des forces méthodologiques . . . . .     | 48 |
| 3.3.2 | Idée générale proposée . . . . .                 | 50 |
| 3.3.3 | Détails du concept . . . . .                     | 52 |
| 3.4   | Travaux préparatoires . . . . .                  | 55 |
| 3.4.1 | Carte sémantique 3D de l'environnement . . . . . | 55 |
| 3.4.2 | Spécification des systèmes . . . . .             | 60 |
| 3.4.3 | Méthode de représentation des données . . . . .  | 63 |
| 3.5   | Synthèse et discussions . . . . .                | 64 |
| 3.5.1 | Positionnement de l'approche . . . . .           | 65 |
| 3.5.2 | Application de la méthodologie . . . . .         | 66 |
| 3.5.3 | Validation méthodologique . . . . .              | 67 |
| 3.6   | Conclusion . . . . .                             | 69 |

---

### 3.1 Introduction

Comme nous l'avons vu dans (Chap 2), la validation des solutions de perception repose sur deux éléments : la référence et la méthode de validation. La référence correspond aux données qui vont servir à évaluer les données issues de la solution testée (données évaluées). Les données de références ont nécessairement une précision plus importante que les données évaluées ou doivent intégrer des informations supplémentaires. Dans le cadre de la perception et de ces travaux de thèses, ces informations correspondent à la sémantique de la scène perçue. Lors d'un processus nommé « référencement », les données de perception acquises sont augmentées par des informations sémantiques. Elles sont annotées par des labels associés aux différentes classes sémantiques traitées. Une fois augmentées, ces données peuvent alors enfin servir de référence pour évaluer différentes solutions de perception ou algorithmes testés.

Ce processus peut être réalisé manuellement, comme évoqué lors du (Chap 2), mais d'autres méthodes peuvent également permettre une automatisation de celui-ci. Bien que le besoin de données de référence soit essentiel pour la validation, ce problème reste cependant relativement inexploré du fait de la complexité qui réside dans l'annotation précise des données afin d'en obtenir des données qualifiables en tant que référence [12]. Ce troisième chapitre débutera par un état de l'art sur les différentes méthodes de référencement de données. Cela nous permettra dans un second temps d'analyser les forces et limitations liées à chacune de ces méthodologies. A partir de cette étude, il nous sera alors possible de conceptualiser une nouvelle méthodologie permettant de répondre au mieux à cette problématique. Enfin, nous discuterons des différents éléments permettant de mettre en application ce concept ainsi que les points méthodologiques sur lesquels une attention particulière devra être portée.

## 3.2 Méthodologies de référencement

En étudiant la littérature ciblée sur l'évaluation des solutions de perception, quatre approches peuvent être distinguées. On les nommera « approches de validation » pour simplifier la lecture. Elles sont représentées (Fig 3.1). Ces approches se différencient sur un point, étudié en détails au cours de cette section : la méthode de référencement des données, simplifiée sous le terme « référencement ».

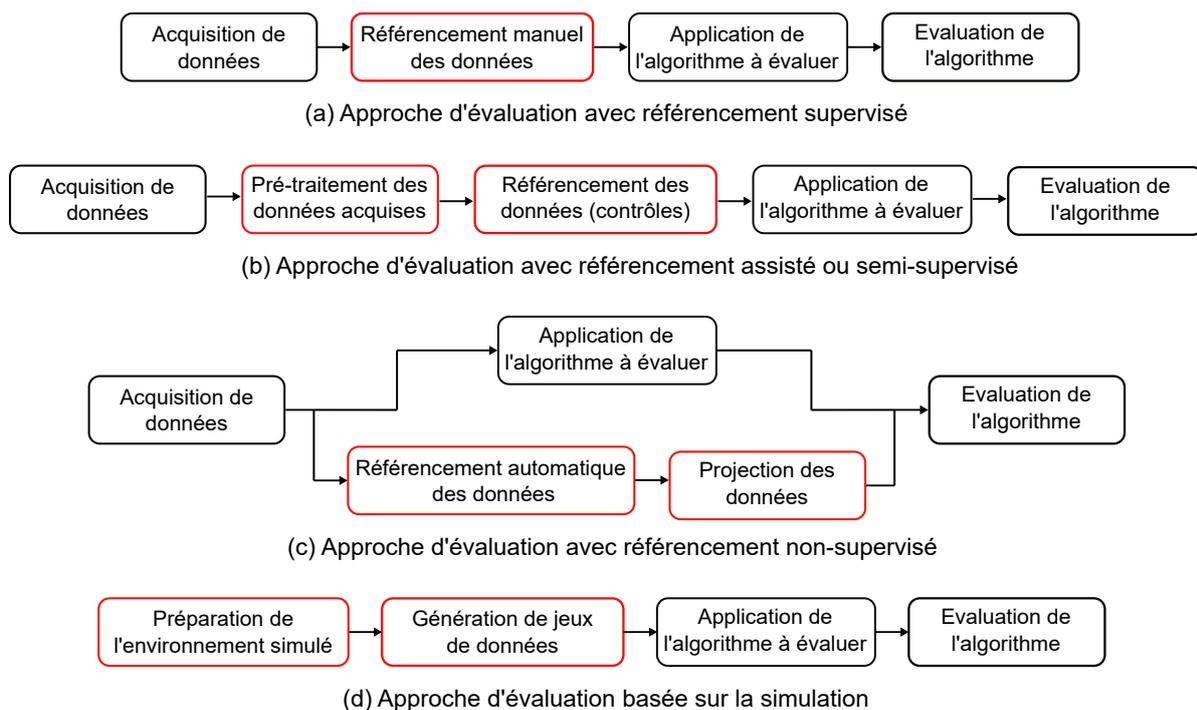


FIGURE 3.1 – Approches de validation étudiées : Représentations des principales approches de validation exploitées dans la littérature. Les différences entre ces approches (en rouge) correspondent aux différents référencements utilisés avec (a) un référencement supervisé ; (b) un référencement assisté ou semi-supervisé ; (c) un référencement non-supervisé ; (d) un référencement basé sur la simulation de données.

Trois étapes sont communes à ces approches de validation :

- L'étape "Acquisition des données" correspond à l'acquisition de données effectuée avec une plateforme équipée de capteurs et sur la base de laquelle les évaluations peuvent être menées. Cette étape est présente dans toutes les approches, exceptée pour celle basée sur la simulation ;
- Lors de l'étape "Application de l'algorithme testé", les algorithmes de traitement associés aux solutions de perception testées sont appliqués sur les données produites lors de l'acquisition des données ;

- L'étape "Évaluation de la solution", les données de sortie des solutions testées sont comparées aux données de référence. A partir de ces comparaisons, des critères d'évaluation peuvent être calculés pour évaluer les performances de la solution.

Les autres étapes qui composent ces approches correspondent aux référencements et sont développées dans la suite de cette section.

### 3.2.1 Présentation des approches existantes

#### 3.2.1.1 Évaluation avec référencement supervisé

Cette première méthode, qui est la plus communément adoptée, se compose d'une seule étape nommée "Référencement manuel des données" (Fig 3.2) et repose exclusivement sur l'intervention humaine.



FIGURE 3.2 – Processus d'évaluation avec méthodologie de référencement des données supervisée.

Au cours de cette étape, les données collectées lors de l'acquisition sont annotées manuellement. L'action réalisée par ces personnes consiste à associer l'ensemble des données à un attribut « label »<sup>1</sup>. Une fois annoté, chaque point d'un nuage, ou pixel d'une image, est associé à une classe sémantique. Par ce type de référencement, de nombreux jeux de données ont été créés puis partagés publiquement sur des plateformes destinées au benchmark d'algorithmes. Ces jeux de données permettent à tout chercheur de pouvoir évaluer des algorithmes, en disposant de données référencées sur la base desquelles ils peuvent comparer leurs résultats en sortie des algorithmes.

On retrouve notamment les jeux de données Cityscapes [14] ou Mapillary Vistas [64], destinés aux capteurs de perception basés vision et pour lesquels les séquences d'images 2D sont segmentées manuellement en régions de pixels puis annotées. Le KITTI Vision Benchmark [65] propose plusieurs séquences composées de données issues de multiple capteurs (caméras et LiDAR) dont les données sont annotées à l'aide de boîtes englobantes 2D pour les images et 3D pour les nuages de points comme on peut le voir sur la page [66]. Un dernier exemple est le jeu de données semantic KITTI [67] basé uniquement sur les données LiDAR du jeu original KITTI Vision Benchmark mais où les nuages de points sont annotés ponctuellement. Au fil du temps ces jeux de données et leur plateforme sont devenus des références dans le domaine de l'évaluation et continuent d'être alimentés par de nouveaux algorithmes qui sont classés en fonction de leurs performances.

Ce référencement est le plus simpliste puisque ce sont les actions manuelles des annotateurs qui permettent le référencement des données. Les données sont sélectionnées manuellement et annotées directement par l'homme, sans algorithme de traitement, ce qui limite la complexité de mise en œuvre. L'interprétation humaine des données permet d'annoter et de référencer les données de façon plus précise. Les annotateurs peuvent interpréter la scène de façon globale et mieux délimiter les objets qui s'y trouvent. Les classes sémantiques associées aux objets bénéficient également de cette interprétation. Pour des traitements automatisés par des algorithmes, une interprétation globale et aussi précise que celle de l'homme reste encore difficile à atteindre [68]. Ces atouts ont contribué à rendre les jeux de données référencés par référencement supervisé populaires.

1. Un label désigne un code (numérique ou alphabétique) permettant d'identifier une classe sémantique.

Malgré ces avantages, la supervision humaine du référencement apporte de nombreuses limitations. Ces limitations sont directement liées aux coûts en termes de moyens et de temps, nécessaires pour référencer un jeu de données [69, 70, 71]. Le nombre d'annotateurs nécessaire pour traiter de grands jeux de données est important. Le traitement manuel, bien que plus précis, demande un temps de traitement bien plus important qu'une analyse automatisée par un algorithme. Par conséquent, une première limitation se trouve dans la taille des jeux de données, ou le nombre limité des séquences, référencées. Une seconde conséquence est de limiter les scénarios (conditions météorologiques, conditions d'éclairage,...) dans lesquels les algorithmes sont testés [12]. Cela pose également un véritable problème avec des séquences de nuits, où les images issues de caméra, sont difficilement interprétables par des humains. De même, les données éparses (par ex : nuages de points 3D) avec une faible résolution, peuvent être complexes à interpréter et à annoter lorsqu'on ne dispose pas d'images issues de caméras de contexte.

De par la production complexe, il est difficile de produire des données de références de façon régulière par cette méthode. On a donc des jeux de données issus d'une période dédiée à l'acquisition, avant d'être post-traités et référencés. Ils sont en général basés sur une configuration de capteur fixe, qui limite les scénarios qui pourraient être analysés en faisant varier la position, l'orientation ou le modèle des capteurs utilisés. Elle est également figée dans le temps, dans la mesure où les jeux de données ne sont pas renouvelés régulièrement, on peut notamment observer cela avec le KITTI Vision Benchmark [4], créé en 2012 et qui n'a pu faire évoluer les jeux de données proposés qu'en 2020 [72]. Cela est également une limitation lorsque l'on tient compte de l'évolution rapide des capteurs de perception. Un jeu de données basé sur une configuration de capteurs précise peut rapidement devenir obsolète.

Enfin, les capteurs utilisés pour produire ces jeux de données sont généralement des capteurs haut de gamme permettant d'avoir un maximum de données exploitables et précises. Cependant, ces capteurs ne sont pas représentatifs des capteurs bas coûts qu'il sera possible d'intégrer sur un véhicule de série. Les évaluations des algorithmes appliquées à des données précises et de haute résolution ne seront donc pas forcément représentatifs des résultats que l'on pourrait obtenir avec des données issues de capteurs à résolution et précision de mesure moindres.

### 3.2.1.2 Évaluation avec référencement semi-supervisé

Dans l'objectif de réduire les limitations du référencement des données supervisé, cette seconde méthode peut être vue comme une extension au référencement supervisé. En introduisant une étape supplémentaire en amont du référencement de données (Fig 3.3), l'objectif est d'optimiser la tâche manuelle des annotateurs, ou de l'accélérer, en limitant leurs interventions.

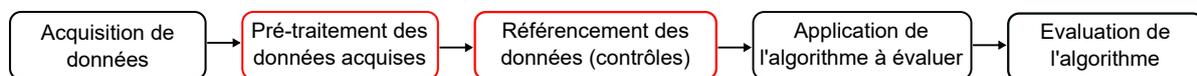


FIGURE 3.3 – Processus d'évaluation avec méthodologie de référencement des données semi-supervisée.

Cette étape introduite se caractérise par l'application d'un outil de pré-traitement des données brutes pour faciliter l'annotation manuelle des données comme on peut le voir dans différents travaux [73, 74, 75, 76, 77]. Les objets identifiés par cet outil sont pré-détectés dans les séquences de données et suivis. La tâche des annotateurs est alors limitée à la vérification des détections et à l'association de chaque détection avec une des classes sémantiques prises en compte. Dans [78], cette étape additionnelle est représentée par l'entraînement d'un réseau de neurones destiné à traiter et annoter automatiquement les données (images 2D ou nuages de points 3D). L'exploitation de l'apprentissage automatique permet, après entraînement, d'automatiser l'étape de référencement des données. Cependant, ces outils de traitement n'étant pas parfaits, une vérification manuelle est toujours nécessaire pour affiner les résultats et/ou corriger les annotations erronées.

Comme pour le référencement supervisé, cette seconde méthode permet de générer des données de référence augmentées par des informations sémantiques précises. Cela est notamment dû aux contrôles ou annotations, toujours effectués sous la supervision des annotateurs. L'apport d'une étape de pré-traitement permet néanmoins d'améliorer la rapidité de l'intervention humaine en réduisant l'action des annotateurs, par rapport à un référencement supervisé sans assistance. En réduisant le temps de traitement pour le référencement, cela apporte la possibilité de produire des données de référence plus rapidement et facilite le traitement de nouveaux jeux de données. Cela permet d'élargir les possibilités quant à la diversité des scénarios pour les évaluations.

La mise en place de cette étape additionnelle peut néanmoins représenter un investissement conséquent tel que l'entraînement de réseaux de neurones. De plus, un entraînement donné n'est pas toujours compatible entre deux jeux de données s'ils se différencient trop (classes sémantiques présentes, densité des nuages de points, résolution des images, etc). Dans ce cas, beaucoup de corrections seront à apporter lors des contrôles. Enfin, l'intervention humaine reste nécessaire avec ce type de référencement. Comme pour l'approche de validation basée sur un référencement supervisé, ne pas totalement déléguer la tâche du référencement des données engendre toujours des limitations pour les évaluations.

### 3.2.1.3 Évaluation avec référencement non-supervisé

Plus récemment, une approche différente (Fig 3.4) est apparue dans la littérature et s'est orientée vers la délégation complète de l'étape de référencement des données.

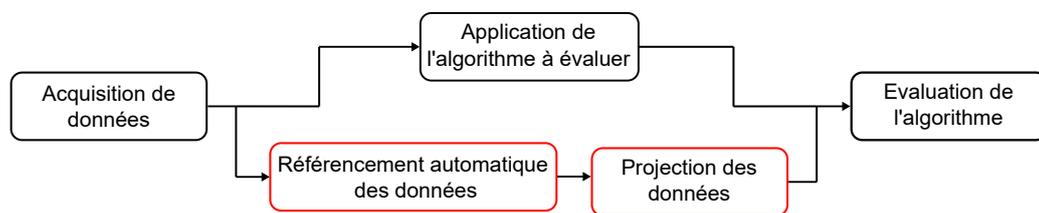


FIGURE 3.4 – Processus d'évaluation avec méthodologie de référencement des données non-supervisée

Cette approche exploite cette fois un « Référencement automatique des données » basé sur une solution de perception indépendante de celle testée. En automatisant le référencement, on passe par une méthode non-supervisée qui ne nécessite plus d'interventions d'annotateurs. Cette approche s'appuie sur l'application de deux solutions de perception en parallèle :  $R$  une solution destinée à produire la référence et  $T$ , la solution de perception testée. Elle est donc applicable sous l'hypothèse que les données en sorties de  $R$  soient plus intègres et précises que les données produites en sortie de  $T$ . Cette hypothèse permet ensuite de considérer les données en sortie de  $R$  comme référence.

A titre d'exemple, on retrouve cette approche dans les travaux [12] et [79], où elle est exploitée pour évaluer  $T$  : une solution de perception pour la détection du sol, basée sur la vision. La solution  $R$  exploitée correspond quant à elle à un LiDAR, couplé à un algorithme de détection du sol. Pour ces références, le prérequis permettant de considérer les données issues de  $R$  comme référence s'appuie sur un niveau de performance garanti par cette solution, et qui repose sur des critères de rappel et de précision [80].

Comme dans le cadre des travaux [12], les données en sortie de  $R$  et celles en sortie  $T$ , peuvent correspondre à deux moyens de représentation distincts. Ici,  $R$  produit des nuages de points 3D où les points sont annotés par les labels « sol » et « non-sol ».  $T$  produit des images dont les pixels sont annotés avec

les mêmes labels. Une étape de « Projection des données » est nécessaire pour projeter les nuages produits par  $R$  sous la forme d'images, par projection projective. Cette seconde étape permet d'avoir des données de référence et des données évaluées représentées dans un espace commun et par un même moyen de représentation. Les comparaisons entre les données peuvent enfin être effectuées pour évaluer les performances de  $T$ .

L'avantage principal de cette méthode de référencement est de faciliter la génération des données de référence. On se détache des limitations en temps et moyens, imposées par l'intervention des annotateurs. Les traitements réalisés, ainsi que les évaluations, peuvent être opérés en temps réel, lors de l'acquisition. En conséquence, n'ayant plus les contraintes recensées par les méthodes de référencement supervisée ou semi-supervisée, cela rend possible l'évaluation d'une solution à travers un nombre de scénarios non limité. Cela permet d'évaluer une solution à travers des conditions d'acquisition variées, ce qui permet de l'évaluer de façon plus concluante. Enfin, cette méthode offre la possibilité de faire évoluer la solution de perception testée  $T$ , sous l'hypothèse qu'elle soit moins performante que  $R$ . On retrouve donc une flexibilité du point de vue de la configuration de capteurs que l'on retrouve difficilement avec les précédentes approches de validation (basées sur un référencement supervisé et semi-supervisé).

Une limite que l'on peut tout de même noter avec cette approche est que la méthode de référencement des données cible un type d'objet ou de structure de l'environnement. Par exemple, dans [12], le référencement est ciblé sur le sol. Dans [79], il est ciblé sur les objets véhicules. On a donc un référencement partiel des données, appliqué sur certaines classes sémantiques. Par conséquence, les données référencées peuvent être utilisées pour évaluer des solutions de perception qui traitent ces mêmes classes sémantiques traitées. Enfin, il reste nécessaire de bien étudier les performances de la solution  $R$  en amont, pour s'assurer que les données qu'elle produit puissent être considérées comme référence. Il sera également nécessaire de combiner plusieurs traitements atteignant un niveau de performances élevé si l'on souhaite référencer l'ensemble des données de la scène perçue. La gestion de ces aspects complexifie l'exploitation de cette méthode de référencement.

### 3.2.1.4 Évaluation basée sur la simulation

Enfin, une dernière approche de validation étudiée dans la littérature se base sur la simulation de données (Fig 3.5).



FIGURE 3.5 – Processus d'évaluation avec données référencées par la simulation.

En exploitant des environnements simulés et différents modèles de capteurs, il est possible de générer des jeux de données de synthèse représentant divers scénarios et configurations de capteurs. Les environnements simulés étant contrôlés, l'ensemble des structures et objets présents sont définis sémantiquement, en amont, lors de l'étape « Préparation de l'environnement simulé ».

Le référencement peut être opéré automatiquement lors de la génération des mesures de synthèse. En fonction de la méthode de calcul pour synthétiser les mesures, celles-ci peuvent être augmentées de l'information sémantique contenue dans l'environnement contrôlé. Les étapes « Acquisition de données » et « Référencement des données », rencontrées au cours des approches étudiées jusqu'ici, sont condensées. Elles sont remplacées par l'étape « Génération de jeux de données » qui produit les données et leurs annotations. Par la suite, les algorithmes évalués sont testés sur les données de synthèse générées. Ces dernières ayant déjà été référencées au cours de leur génération, les labels associés aux données en sortie des algorithmes testés peuvent être comparés aux labels initiaux.

Ces comparaisons permettent d'évaluer les performances des algorithmes à l'image des méthodes présentées précédemment. Cette méthode a récemment été exploitée pour des solutions basées sur la vision [81, 82], comme pour des solutions basées LiDAR [83, 84, 85].

Grâce aux environnements contrôlés issus de la simulation, cette approche est celle qui offre le plus de flexibilité. Les données sont annotées avec précision grâce à différentes techniques telles que le « ray tracing » [86]. Les solutions de perception peuvent être facilement évaluées dans divers scénarios (vitesses, luminosités,...) et divers types d'environnements (urbain, autoroutier, rural, ...). La configuration des capteurs est également très évolutive. Les modèles des capteurs utilisés dans un simulateur peuvent facilement être remplacés ou reconfigurés. Cela signifie que leurs calibrations extrinsèques peuvent être facilement paramétrées en fonction de différentes configurations (positions, orientations,...) vis-à-vis de la plateforme mobile simulée. Il est également possible d'intervenir sur les paramètres intrinsèques en modulant la portée maximale considérée, la résolution ou encore la fréquence des mesures. Il est alors possible de générer des données annotées, sans les efforts nécessaires pour mener une acquisition en conditions réelles et dans un nombre de scénarios illimités et extrêmement variables.

Bien que l'atout principal de la simulation réside dans ses possibilités, cette approche de validation possède cependant certaines limites. La première est que les données de synthèse, issues de la simulation, restent des données idéalisées. Les phénomènes extérieurs, liés à l'environnement « réel », peuvent perturber les données lors des acquisitions en conditions réelles. Ils sont liés aux conditions météorologiques (pluies, brouillard, etc), de luminosités ou de réflectance, qui peuvent intervenir sur les principes physiques de mesures des capteurs, puis impacter et brouter les données. Ces phénomènes, sont difficilement modélisables et reproductibles sur des données générées par simulation. Les données de synthèse ne sont donc pas parfaitement représentatives des données réelles comme le montre les études des travaux [87, 88, 89]. La conséquence directe de baser uniquement l'évaluation d'une solution, sur des données de synthèse, est d'obtenir des résultats biaisés et non représentatifs des performances pour un usage en conditions réelles. Cet aspect est visible dans les travaux [85], où une solution comprenant un algorithme de perception, entraîné sur des données réelles, est évaluée avec des données issues de jeu de données KITTI et performe avec une précision d'environ 70 %. Pour ce même algorithme, entraîné uniquement sur des données synthétiques, la précision obtenue lorsqu'il est appliqué sur les données de KITTI dépasse à peine les 30 %. L'évaluation d'une solution uniquement basée sur des données simulées ne permet donc pas d'évaluer de manière concluante ses performances pour un usage en conditions réelles.

### 3.2.2 Synthèse comparative

Les différentes approches décrites sont apparues dans l'objectif de simplifier l'accès à des données de références et de faciliter l'évaluation d'algorithmes de perception. Afin de les comparer, nous les présentons dans un tableau de synthèse (Tab 3.1). Les scores attribués se basent sur cinq critères représentatifs de la problématique de référencement des données. On retrouve ainsi :

- La précision de la référence : qui détermine si les données produites par les différentes approches et utilisées comme référence sont précises ;
- L'évolutivité des capteurs : qui représente la possibilité de faire évoluer la configuration des capteurs embarqués. Plus il est facile de faire varier cette configuration, plus il est aisé de tester des algorithmes se basant sur différents types de capteurs et différents types de données. Il est ainsi possible de faire évoluer la configuration avec l'arrivée de nouveaux capteurs et de produire des jeux de données avec eux ;

- L'évolutivité des scénarios : qui représente la facilité de produire des données de références basées sur de nouveaux scénarios. Il est ainsi possible de faire évoluer les jeux de données rapidement ;
- L'efficacité des traitements : qui représente la facilité de générer des données de référence pour les évaluations en tenant compte de la consommation en temps et en moyens que le référencement nécessite ;
- La représentativité des données : qui détermine si les données de références et les évaluations menées sont représentatives des données obtenues en conditions réelles.

| Critères \ Approches         | Approches  |                             |                |                  |
|------------------------------|------------|-----------------------------|----------------|------------------|
|                              | Supervisée | Assistée<br>Semi-supervisée | Non-supervisée | Basée simulation |
| Précision de la référence    | ****       | ****                        | ***            | ****             |
| Évolutivité des capteurs     | *          | *                           | **             | ****             |
| Évolutivité des scénarios    | *          | **                          | ****           | ****             |
| Efficacité des traitements   | *          | **                          | ****           | ****             |
| Représentativité des données | ****       | ****                        | ****           | *                |

TABLE 3.1 – Synthèse des forces et faiblesses des différentes approches de référencement des données

Cette analyse nous permet de mettre en évidence qu'il n'existe pas de solution idéale. Les approches supervisées et semi-supervisées, ou assistées, tirent le bénéfice du fait d'annoter des données obtenues en conditions réelles par des annotateurs humains. Cela permet d'avoir des données représentatives de la réalité et dont l'interprétation, ou le contrôle par des annotateurs, permet des annotations précises et correctes. L'inconvénient de ces approches est qu'elles sont limitées par la complexité qu'impose cette supervision et qui limite les possibilités de flexibilité pour le référencement.

Les approches non-supervisées permettent d'augmenter cette flexibilité de par l'automatisation du référencement des données, tout en conservant la représentativité des données obtenues en conditions réelles d'acquisition. Cependant, il faut garantir que l'algorithme de référencement automatique soit suffisamment performant pour être plus précis que celui testé et garantir la fiabilité de la référence. A ce jour, cette approche est principalement ciblée sur un type d'algorithme donné, basé sur les données d'un type de capteur donné. L'approche est donc intéressante, mais manque de flexibilité sur ce point de vue.

Enfin, l'approche basée sur la simulation est celle qui remplit l'ensemble des critères en termes de flexibilité. Son défaut majeur est la représentativité des données. Les données de synthèses sont idéalisées et ne reflètent donc pas forcément les données réelles.

### 3.3 Concept méthodologique proposé

#### 3.3.1 Analyse des forces méthodologiques

L'étude des différentes méthodologies de référencement trouvées dans la littérature scientifiques nous a permis de constater que des efforts sont faits pour réduire l'intervention humaine qui imposent de nombreuses limitations. Les discussions faites sur les forces et limitations liées à chacune de ces approches sont restituées sous forme synthétique par le tableau (Tab 3.1). Elles permettent d'observer qu'en fonction de la méthodologie utilisée, différentes limitations persistent à rendre l'accès aux données de référence complexe.

On peut cependant constater qu'en fonction des critères utilisés pour les départager, ces approches ne sont pas impactées par les mêmes limitations. Une nouvelle approche située à la frontière des différentes méthodes de référencement étudiées (Sec 3.2) pourrait être un moyen d'obtenir une nouvelle méthode de référencement, suffisamment flexible pour répondre à cette problématique et faciliter l'accès aux données référencées. L'objectif est d'analyser les points forts, propres à chaque méthode de référencement, pour tenter de les rassembler au sein d'une seule et même méthode. L'analyse de ces forces pour les différents référencements étudiés est la suivante :

Pour le référencement supervisé ou semi-supervisé, le premier point fort se situe dans la représentativité des données référencées. Celles-ci ont été obtenues en conditions d'acquisition réelles et se situent au plus proche des conditions d'usage des solutions testées. Elles intègrent les bruits de mesures liés aux phénomènes extérieurs (météo, luminosité,...), les sensibilités dues à la dynamique de la scène et de la plateforme ou encore les mesures aberrantes des capteurs. La seconde force correspond à la précision des annotations, atteinte grâce à la supervision et l'interprétation humaine des annotateurs et qui dépasse celle atteinte par des algorithmes de traitements non-supervisés.

Pour les données référencées directement par l'utilisation de la simulation, les forces se situent principalement dans la flexibilité en termes de scénarios et de configurations de capteurs qu'il est possible de paramétrer pour générer des données de synthèse. L'environnement simulé étant contrôlé et parfaitement connu, les données de synthèse sont directement annotées lors de leur génération en récupérant la sémantique de cet environnement. Cela permet d'optimiser les traitements nécessaires et d'obtenir des données de synthèse idéalisées, mais annotées de façon exacte.

Enfin, concernant le référencement non-supervisé, on retrouve la représentativité des données car le référencement est opéré sur des données acquises en conditions réelles. Avec cette approche de validation, on distingue la solution de perception testée de celle utilisée pour produire les données de référence. Ces deux solutions peuvent s'appuyer sur des principes de mesures différents. La projection des données de la solution de référence vers l'espace de représentation des données de la solution testée, permet de renforcer la flexibilité de cette approche grâce à la capacité de franchir les limites de mesures correspondant à une unique modalité. Il est ainsi possible d'évaluer des solutions de perception basées sur différents principes de mesures mais également de faire évoluer la configuration de la solution de perception testée. Enfin, pour les travaux utilisant cette approche, le référencement est opéré par un algorithme de traitement automatique. Il est possible de référencer les données en temps réel, au cours de l'acquisition. Cela renforce la flexibilité de cette approche en permettant de l'appliquer pour n'importe quel scénario ou environnement.

Les points forts liés aux différentes méthodologies de référencement peuvent être synthétisés par la figure (Fig 3.6). En les regroupant au sein d'une même méthode, cela permettrait de générer des données de référence efficacement, en s'adaptant aux différentes acquisitions et scénarios réalisés lors des tests, mais également, pour que ces données de référence puissent être exploitées pour différentes solutions de perception testées. Les limitations, propres à chaque méthode de référencement prise individuellement, ne seraient alors, pas impactantes pour l'évaluation de solutions.

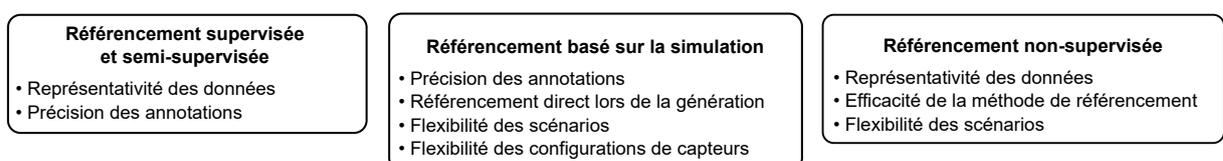


FIGURE 3.6 – Points forts associés aux différents référencements étudiés.

### 3.3.2 Idée générale proposée

Après cette étude des différentes méthodologies de référencement rencontrées dans la littérature, nous avons pu conceptualiser notre propre méthode, tout en cherchant à tirer parti des différentes forces, propres à chaque méthode étudiée et mises en avant dans la (Sec 3.3.1). L'approche de validation que nous proposons est présentée (Fig 3.7). Elle intègre une nouvelle méthode de référencement qui sera présentée en détails dans la suite de cette thèse et qui permet de faciliter l'évaluation de solutions de perception variable et dans différents scénarios. Cette approche se base sur une première hypothèse qui est que les environnements peuvent être décomposés en deux catégories d'éléments distincts :

1. les éléments fixes dans l'environnement, composés par les structures et objets statiques (route, bâtiments, panneaux de signalisation, marquages au sol, etc) ;
2. les éléments dynamiques, correspondant aux objets amovibles qui interagissent dans cet environnement (piétons, véhicules, etc).

Lorsque les performances des solutions de perception sont évaluées, les tests sont généralement opérés sur un site d'essai défini. Bien que les scénarios puissent varier vis-à-vis des conditions météorologiques, des conditions d'éclairage ou des objets dynamiques présents lors des acquisitions de tests, ceux-ci peuvent être réalisés au sein d'un même environnement : le site d'essai, nommé Environnement d'Essai et noté *EE*. Ainsi, pour tout jeu de données correspondant à une acquisition localisée dans *EE*, une partie des données représenteront les mêmes éléments communs : les éléments fixes de l'environnement, quels que soient les scénarios testés.

Nous avons ainsi orienté nos travaux à partir de cette hypothèse. L'aspect novateur de notre approche réside dans la prédiction automatique des annotations correspondant à ces éléments fixes de l'*EE*, pour référencer les données. L'idée générale est d'exploiter une base de données contenant les informations sémantiques correspondant à ces éléments fixes, indépendante des capteurs embarqués dans la plateforme et modélisée sous la forme d'une carte sémantique en 3 dimensions. Le rôle de la méthode de référencement mise en place sera de transférer les informations sémantiques contenues dans la carte 3D vers les données acquises par les capteurs, lors des tests. Ces différents aspects sont retrouvés dans les étapes qui composent l'approche (Fig 3.7).

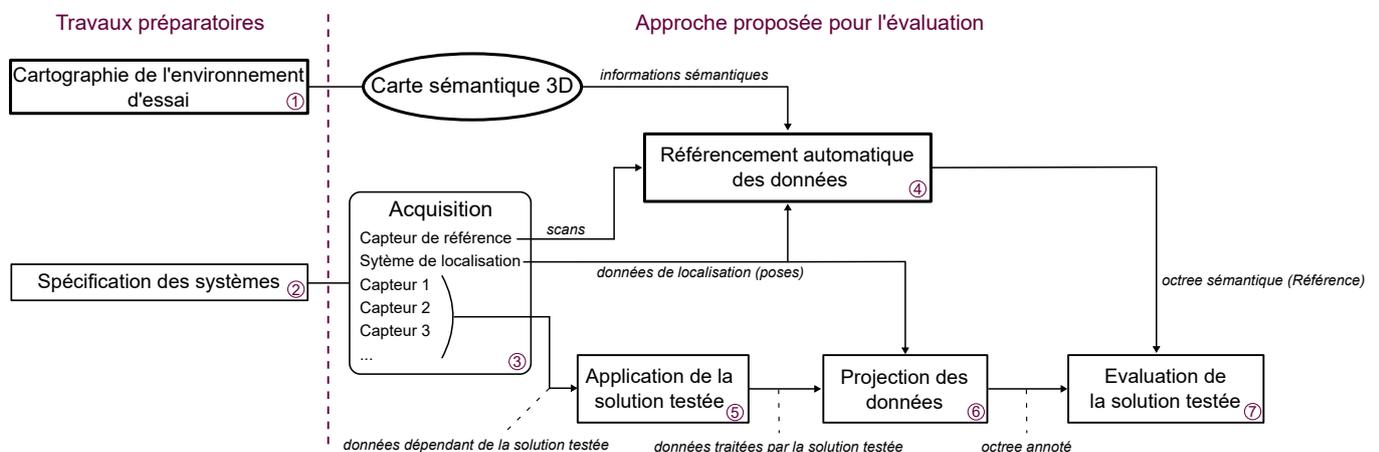


FIGURE 3.7 – Approche de validation des solutions de perception proposée : les numéros assignés aux étapes correspondent aux numéros des paragraphes de description de ces étapes.

1. La première étape au cours des travaux préparatoires est nommée "Cartographie de l'environnement d'essai". Elle représente l'élaboration de la carte sémantique 3D modélisant fidèlement l'environnement d'essai *EE*, dans lequel les scénarios de test sont mis en place. Cette étape correspond à une opération qui n'est réalisée qu'une seule et unique fois, en amont de l'application de l'approche. C'est également la seule opération qui nécessite l'intervention d'annotateurs. Une fois cette opération réalisée, les autres étapes de l'approche peuvent être appliquées pour tout scénario de test ou jeux de données se déroulant dans le même environnement représenté par la carte 3D. Tous les éléments fixes présents dans l'environnement d'essai sont référencés au sein de cette carte 3D et sémantiquement annotées.
2. L'étape « Spécification des systèmes », également réalisée au cours des travaux préparatoires, consiste à définir les capteurs qui seront embarqués sur la plateforme. Ces capteurs serviront de sources de données pour différents objectifs. Le capteur de référence interviendra dans le cadre de la production de la référence. Un système de localisation sera exploité pour le géoréférencement des données. Enfin, d'autres capteurs embarqués seront utilisés pour disposer de données de test, traitées par les solutions de perception testées.
3. La première étape correspondant à l'application de l'approche proposée est "Acquisition". Déjà introduite (Sec 3.2), cette étape consiste à mener une ou plusieurs acquisitions avec la plateforme équipée. Au cours de ces acquisitions les données nécessaires aux évaluations seront produites par les différents capteurs.
4. L'étape « Référencement automatique des données » s'inscrit au sein d'une approche de validation comparable à celles basées sur un référencement non-supervisé (Fig 3.1-c), présentées dans la (Sec 3.2.1), avec les travaux [12, 79]. Les approches sont comparables dans la mesure où aucune intervention humaine n'est opérée lors de l'étape « Référencement automatique des données ». De plus, ce sont des données obtenues en conditions d'acquisition réelles qui sont référencées, et non des données de synthèse. Dans notre cas, le référencement, détaillé dans la suite de cette thèse, s'opère en s'inspirant de l'adaptation de domaine. Les données annotées automatiquement par ce référencement sont ensuite utilisées comme référence.
5. Lors de l'étape "Application de la solution testée", les algorithmes de traitement associés aux solutions de perception testées sont appliqués sur les données produites lors de l'étape d'acquisition.
6. A l'inverse de l'approche (Fig 3.1-c), l'étape de "Projection des données" que nous avons inclus dans notre approche, est utilisée pour projeter les données en sortie de la solution testée vers l'espace de représentation des données de référence.
7. Enfin, la référence produite est confrontée aux données issues des solutions de perception testées au cours de l'étape « Évaluation de la solution testée ». Cela permet d'évaluer les performances de ces solutions selon les différents critères d'évaluation choisis.

### 3.3.3 Détails du concept

#### 3.3.3.1 Inspiration de l'adaptation de domaine

L'adaptation de domaine est un mécanisme activement étudié pour des applications d'apprentissage automatique non-supervisé [90, 91]. Il intervient dans le champ de recherche de l'apprentissage par transfert qui vise à transposer des connaissances provenant d'une tâche source vers une tâche cible. Il peut être défini comme la capacité d'un système à reconnaître et appliquer des connaissances apprises à partir de tâches antérieures, sur de nouvelles tâches, ou domaines, partageant des similitudes.



FIGURE 3.8 – Exemple d'application de l'adaptation de domaine : (a) Image de synthèse obtenue par la simulation d'une caméra monoculaire dans un environnement simulé ; (b) Image d'une caméra monoculaire obtenue en conditions d'acquisition réelles ; (c) Exemple d'application de l'adaptation de domaine permettant de rapprocher des données du domaine « monde réel » vers (b) le domaine de la simulation (a). [92].

Bien que la méthodologie mise en place dans nos travaux ne repose pas sur l'apprentissage automatique, nous nous inspirons de ce mécanisme pour effectuer un transfert d'informations sémantiques d'un domaine source vers un domaine cible.

#### 3.3.3.2 Domaines exploités

Dans notre cas, ce transfert est appliquée entre les domaines *CRA* et *MAP*. Ils peuvent être décrits de la façon suivante :

*CRA* : pour Conditions Réelles d'Acquisition, représente le domaine source. Il s'agit du domaine dans lequel les données correspondent à des mesures faites par les différents capteurs embarqués sur la plateforme, lors des acquisitions. Les données appartenant à ce domaine peuvent correspondre aux données issues de capteurs de perception, comme aux données de localisation. Ce sont des données issues d'une interaction avec le monde réel, soit en cherchant à le représenter via la perception, soit en cherchant à s'y situer par la localisation. Les données appartenant à ce domaine seront nommées « données réelles » dans la suite de cette thèse, pour simplifier la lecture.

*MAP* : correspond au second domaine et représente la carte sémantique 3D de l'environnement *EE*. Il correspond au domaine cible car il contient les informations sémantiques avec lesquelles on cherche à référencer les données issue de *CRA*. Pour modéliser cet environnement par une carte sémantique 3D, le moyen de représentation le plus adapté correspond à un modèle 3D surfacique, où les données sont stockées sous la forme de facettes<sup>2</sup> géométriques et qui sera détaillé dans la suite de ce chapitre.

2. Les facettes correspondent à des surfaces 2D. Elles sont définies par des sommets correspondant à des points 3D permettant de définir l'orientation de la surface plane. Ces surfaces sont associées à des formes géométriques simples. Pour représenter un élément dont les surfaces ne sont pas régulières et homogènes, les facettes utilisées pour le représenter ont la

### 3.3.3.3 Procédé d'adaptation

La méthode de référencement utilisée dans notre approche s'appuie ainsi sur un transfert d'informations sémantiques à partir des données appartenant au domaine *MAP* vers celles appartenant au domaine *CRA*. Cependant, les données qui composent la carte sémantique 3D sont contenues sous la forme de facettes géométriques, qui ne correspondent pas aux moyens de représentation des données, produites de façon native, par les capteurs de perception. Il est donc complexe d'opérer un transfert direct entre les données des domaines *MAP* et *CRA*. Un moyen de répondre à ce problème réside alors dans la simulation de données.

**Parallèle avec la simulation.** Un autre cadre dans lequel les modèles 3D sont utilisés est la simulation. Pour solutionner le problème de représentations différentes entre les domaines, nous avons choisi d'exploiter ce cadre. En utilisant la carte sémantique 3D comme environnement de simulation, ainsi qu'en modélisant le capteur de perception embarqué, il est possible de générer des données de synthèse, représentées de façon identique aux données réelles. Par exemple, en modélisant un LiDAR, les données de synthèse générées correspondront à des nuages de points 3D, comme celles produites par un LiDAR avec des mesures réelles.

Intégrer la simulation au cours du transfert d'informations permet à notre méthode de référencement de bénéficier des points forts liés au référencement basé sur la simulation (Sec 3.2.1). Les données de synthèse sont annotées directement, lors de leur génération, grâce aux informations sémantiques de la carte 3D. Ces données de synthèse issues du domaine *MAP* permettront de faciliter le transfert des informations sémantiques contenues initialement dans la carte 3D, vers les données réelles.

**Application du transfert d'informations au cours du référencement.** Pour référencer les données du domaine *CRA* en s'inspirant des méthodes d'adaptation de domaine, l'idée est de générer, pour chaque trame de données mesurée, une trame de données analogue de synthèse, issue du domaine *MAP*. Les données de synthèse sont générées avec un modèle de capteur correspondant à celui utilisé pour réaliser les mesures dans le monde réel. Enfin, pour générer des données analogues, il faut aussi pouvoir positionner le modèle de capteur simulé à la même position que le capteur réel lors de l'enregistrement des mesures.

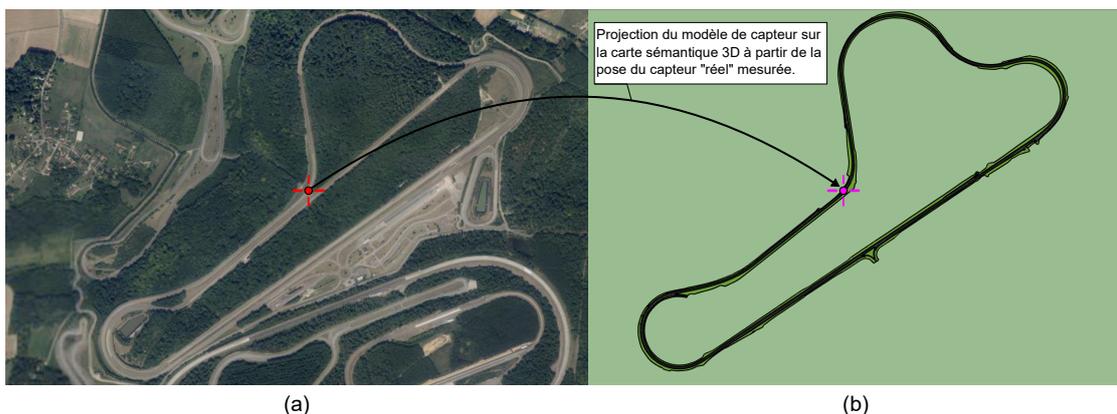


FIGURE 3.9 – Positionnement analogue entre capteur dans le monde réel et capteur modélisé : (a) Position associée au capteur dans le monde réel lors de l'enregistrement de mesures du domaine *CRA* ; (b) Projection du modèle de capteur correspondant, sur la carte 3D et dans un environnement simulé, pour synthétiser les mesures du domaine *CRA* à partir du domaine *MAP*.

forme de triangles. Pour des éléments composés de surface régulières et lisses (par exemple : les murs de bâtiments ou la base des panneaux de signalisation), ils peuvent être simplifiés par l'utilisation de « primitives géométriques » et être modéliser en exploitant des formes géométriques 3D telles que des cubes ou des cylindres.

Pour créer le lien entre les deux domaines, illustré (Fig 3.9), il faut utiliser le même système de coordonnées pour géoréférencer la carte sémantique 3D et les données mesurées du domaine *CRA*. Lorsque des mesures produites lors d'une acquisition sont enregistrées, le capteur, dont elles proviennent, est associé à une pose spécifique. Cette pose est réutilisée ensuite pour positionner le capteur modélisé sur la carte sémantique 3D, au même emplacement que le capteur réel, avant de générer des mesures analogues de façon synthétique.

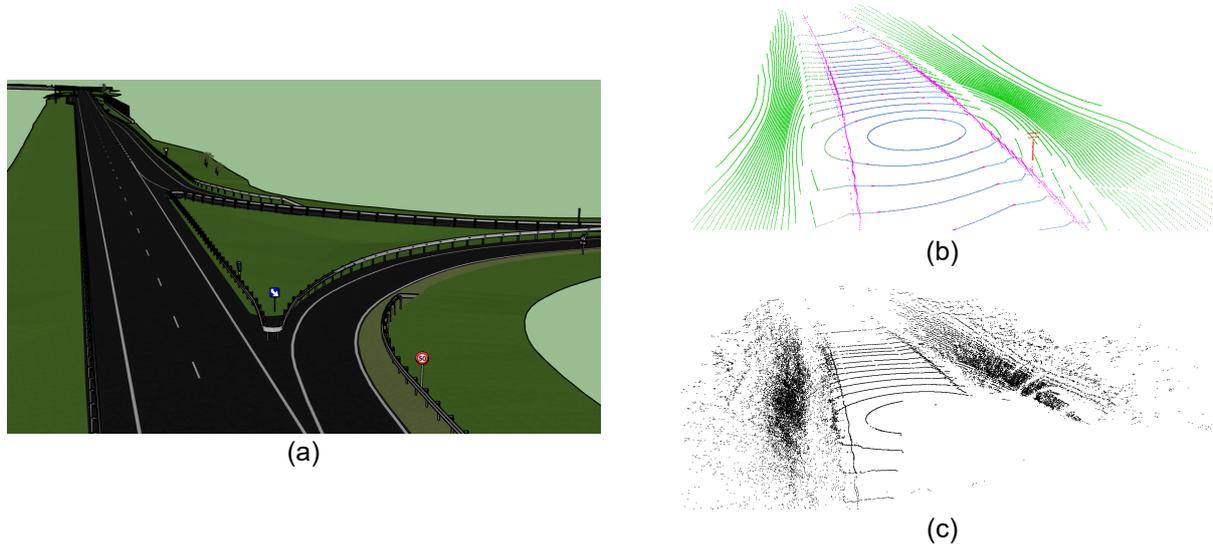


FIGURE 3.10 – Représentation des domaines et types de données utilisés par la méthode de référencement : (a) Extrait de la carte sémantique 3D représentant *EE*; (b) Données de synthèse produites grâce à la carte 3D et au modèle de capteur simulé (représentant le domaine *MAP*); (c) Données de mesures issues du monde réel (représentant le domaine *CRA*).

Cela permet, dans la dernière étape du référencement d'opérer une association entre les données propres à chaque domaine. On dispose d'un côté des mesures réelles représentant les données du domaine *CRA*, et de l'autre côté, des mesures obtenues de façon synthétique et annotées, correspondant au domaine *MAP*. Il est possible d'associer ces données géométriquement (suivant un formalisme classique d'association de données). Cette étape d'association permet de finaliser le référencement. Les données réelles référencées par cette méthode peuvent enfin servir de référence pour évaluer les données obtenues en sortie d'une solution de perception testée.

### 3.3.3.4 Synthèse des avantages

Grâce à cette méthode de référencement, on retrouve les différents points forts qui composent les référencement étudiés dans la (Sec 3.3.1). La supervision manuelle exploitée lors de la première étape « Cartographie de l'environnement », correspondant à l'élaboration de la carte sémantique 3D, permet de retrouver la grande précision des données de référence mise en avant avec la première approche (Fig 3.1-a). Cependant, à l'inverse de celle-ci, le référencement sémantique manuel et coûteux en temps et moyen n'est opéré qu'une seule fois sur la carte 3D, et non pas sur les jeux de données produits après chaque acquisition.

L'automatisation de référencement est obtenu en s'inspirant de l'adaptation de domaine entre les deux domaines identifiés : *MAP* et *CRA*, ainsi que par l'association géométrique entre les données issues de ces deux domaines. Le référencement automatisé permet de traiter des jeux de données complets tout en limitant la consommation de temps et de moyens. L'approche globale se rapproche de celle rencontrée avec le référencement non-supervisé (Sec 3.2.1). L'utilisation d'un algorithme d'association géométrique repose néanmoins sur des principes plus simples que les algorithmes de classification utilisés dans [12, 79]. Cela permet d'enlever les incertitudes propres à ces algorithmes, et conserve la précision des annotations

de la carte sémantique 3D. Ce référencement non-supervisé permet de traiter les données de toutes les nouvelles acquisitions, pour divers scénarios, à condition qu'elles soient localisées dans l'environnement *EE*.

Cette même flexibilité permet également de pouvoir faire évoluer dans le temps la configuration des capteurs embarqués sur la plateforme. Il suffit d'appliquer les mêmes évolutions dans le simulateur en modifiant le modèle de capteur utilisé et/ou ses paramètres de calibration extrinsèques et intrinsèques. On retrouve, grâce à cet aspect, l'adaptabilité en termes de configuration de capteur, précédemment rencontré avec la méthode de référencement basée sur la simulation (Fig 3.1-d).

Enfin, bien que le référencement utilise des données de synthèse parmi les sources de données qu'il utilise, les informations sont bien transférées sur des données réelles. On converse ainsi la représentativité des données tout en apportant l'information sémantique permettant de les référencer.

## 3.4 Travaux préparatoires

Afin d'appliquer notre concept méthodologique pour la validation de perception, on a pu voir au cours de la (Sec 3.3) que plusieurs éléments devaient être mis en place en amont. Ces éléments permettront d'appliquer la méthode de référencement de données conceptualisée jusqu'ici.

### 3.4.1 Carte sémantique 3D de l'environnement

Le premier élément à mettre en place est la carte sémantique 3D de l'environnement. La production de cette carte est représentée au sein de l'approche de validation (Fig 3.7) par l'étape « Cartographie de l'environnement d'essai ». Comme on l'a introduite dans la (Sec 3.3.3.2), cette carte modélise en 3 dimensions l'environnement : la piste d'essai sur laquelle les évaluations de solutions de perception sont réalisées. Les travaux nécessaires pour la mise en place de cette carte sémantique 3D nécessitent l'intervention d'annotateurs, mais ne sont réalisées qu'une seule fois par environnement utilisé. Pour un environnement donné, une fois la carte 3D produite, les autres étapes du traitement qui composent l'approche de validation (Fig 3.7) peuvent être appliquées sans nécessiter de repasser par cette étape de production, ni de nécessiter l'intervention humaine dans les traitements.

#### 3.4.1.1 Description de la carte sémantique 3D

Cette carte sémantique 3D doit être la plus détaillée, précise et fidèle à cet environnement. Chaque élément fixe et présent dans l'environnement doit y être référencé et être associé à une classe sémantique. Comme on l'a évoqué (Sec 3.3.3.2), le monde étant composé d'éléments surfaciques, le meilleur support pour modéliser un environnement de façon représentative à la réalité, correspond aux modèles 3D surfacique tels que ceux exploités dans le cadre de la simulation ou des jeux-vidéos. Ces modèles 3D se composent alors de facettes liées entre elles par des sommets communs. La représentativité et la fidélité de retranscription de l'environnement va directement dépendre de la taille de ces facettes, ce qui correspond à la résolution du modèle 3D. Plus les facettes seront de petites dimensions (et donc une grande résolution pour le modèle 3D), plus elles pourront représenter un objet fidèlement en prenant en compte ses aspérités.

Chaque facette composant les différents objets de la carte 3D peut être annotées en concordance avec la classe sémantique de l'objet auquel elle appartient. Les informations sémantiques associées aux facettes pourront ensuite être exploitées par le simulateur, lors de la génération des données de synthèse, pour les annoter automatiquement et transférer la sémantique de la carte sémantique 3D vers les données de synthèse.

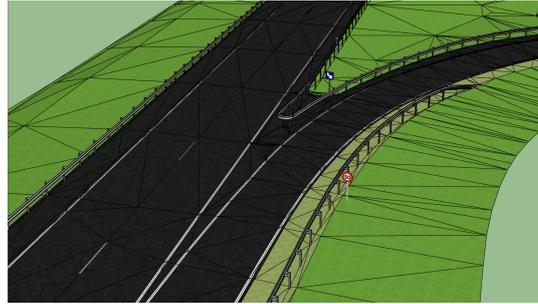


FIGURE 3.11 – Facettes composant un modèle 3D : Extrait d'un modèle 3D avec facettes apparentes (en noir) sur les éléments associés au sol (route et fossés).

### 3.4.1.2 Élaboration de la carte 3D par TT Géomètres-Experts

Pour produire la carte 3D et modéliser l'environnement réel existant, trois opérations sont nécessaires et sont réunies dans l'approche de validation (Fig 3.7) par l'étape « Cartographie de l'environnement d'essai ». Ces trois opérations sont : 1) la numérisation 3D de l'environnement ; 2) le post-traitement des données de cartographie pour obtenir un modèle 3D et enfin, 3) le référencement sémantique du modèle 3D.

Dans notre cas, le modèle 3D dont nous disposons durant ces travaux de thèse correspond au site d'essai du groupe Renault. Cet environnement correspond à un scénario du type autoroutier. Les deux premières opérations citées dans la paragraphe précédent permettent d'obtenir un modèle 3D formant la base de la carte sémantique 3D voulue. Celles-ci ont été sous-traitées à l'entreprise TT Géomètres-Experts<sup>3</sup>.

**Numérisation 3D de l'environnement.** La production de la carte 3D fait appel à l'utilisation d'une plateforme de cartographie mobile. Il s'agit en général de véhicules équipés de plusieurs LiDARs et/ou de multiples caméras (Fig 3.12-a,b), permettant de scanner en 3D l'environnement avec une très haute résolution et une précision des mesures de l'ordre de +/- 1 cm. Les acquisitions réalisées à basses vitesses, combinées à de multiples passages, permettent de capturer un maximum d'informations sur l'environnement scanné. La basse vitesse permet également d'éviter les déformations des scans LiDAR liées à la dynamique du véhicule.

Ces systèmes sont également équipés d'un système de localisation basé sur un couplage GPS-RTK / centrale inertielle permettant un positionnement relatif de précision +/- 1 cm et absolu de l'ordre de 2 cm à 5 cm (Annexe C).

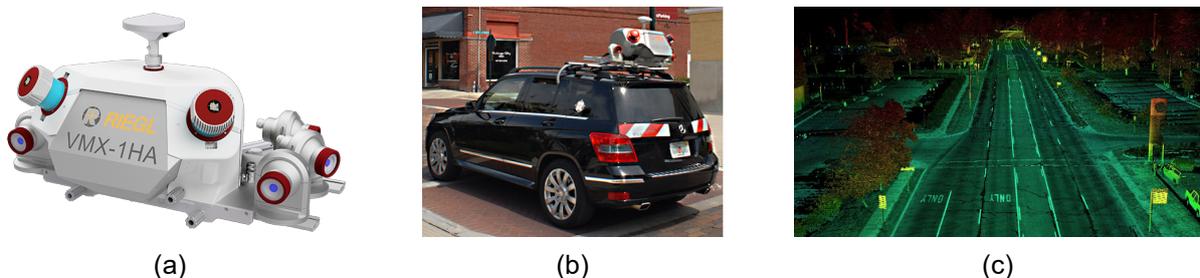


FIGURE 3.12 – Cartographie mobile : (a) Système de cartographie mobile Riegl VMX-250 ; (b) Système Riegl VMX-250 embarqué sur une plateforme ; (c) Exemple de données fournies par ce type de système.

3. <https://www.tt-geomeres-experts.fr/>

**Post-traitement des données.** Une fois l'acquisition réalisée, les données brutes enregistrées subissent une chaîne de traitements qui va permettre de transformer les scans LiDARs en modèle 3D. La première étape est de géoréférencer l'ensemble des données en associant temporellement les données de localisation aux données de perception. Les données obtenues sont réunies sous la forme d'un nuage de points extrêmement dense et couvrant l'ensemble de la zone cartographiée. Ces nuages ont une résolution pouvant atteindre un ordre de grandeur millimétrique (Fig 3.12-c) sur des distances de plusieurs centaines de mètres, voire de quelques kilomètres. Ces données permettent d'obtenir une première représentation numérique, et en 3D, de l'environnement cartographié.

Un des inconvénients du nuage de points est la difficulté de le manipuler du fait de sa densité. Le nuage va donc être segmenté pour regrouper les points appartenant à un même élément. Chaque segment représente un élément de l'environnement. Cette étape permet également de filtrer le nuage en supprimant les points bruités ou associés à des éléments qui n'appartiennent pas à l'environnement cartographié.

Enfin, pour optimiser le traitement d'information et réduire drastiquement le volume de données, une opération de maillage est opérée sur les segments. L'objectif de cette opération est de définir les points qui serviront de sommets pour les facettes et de construire ces facettes [93]. Les données vont alors être composées par les facettes planes qui s'adaptent le mieux aux différentes surface des segments. Lors de cette étape de maillage, il est possible de définir, en fonction du segment, la résolution spatiale adéquate pour représenter l'élément correspondant à ce segment. A titre d'exemple, pour un segment représentant la route, qui est un élément dont la surface est homogène et régulière, les facettes peuvent avoir des côtés dont la longueur est de l'ordre du mètre. Pour d'autres éléments tels que les glissières de sécurité, la résolution spatiale doit être plus importante avec des facettes dont les côtés ont un ordre de longueur de quelques centimètres, afin de conserver la géométrie de l'objet.

Une fois le maillage réalisé, seules les facettes sont conservées pour composer le modèle 3D. L'ensemble des points n'est plus utilisé. On réduit drastiquement le volume de données pris en compte. De plus, on ne travaille plus sur une représentation éparsée composée de points, mais sur une représentation dense composée d'éléments surfaciques, ce qui se rapproche davantage au monde réel. Cela permet de faciliter l'utilisation du modèle 3D en tant que carte 3D, mais permet également d'optimiser la dernière opération : le référencement sémantique de la carte.

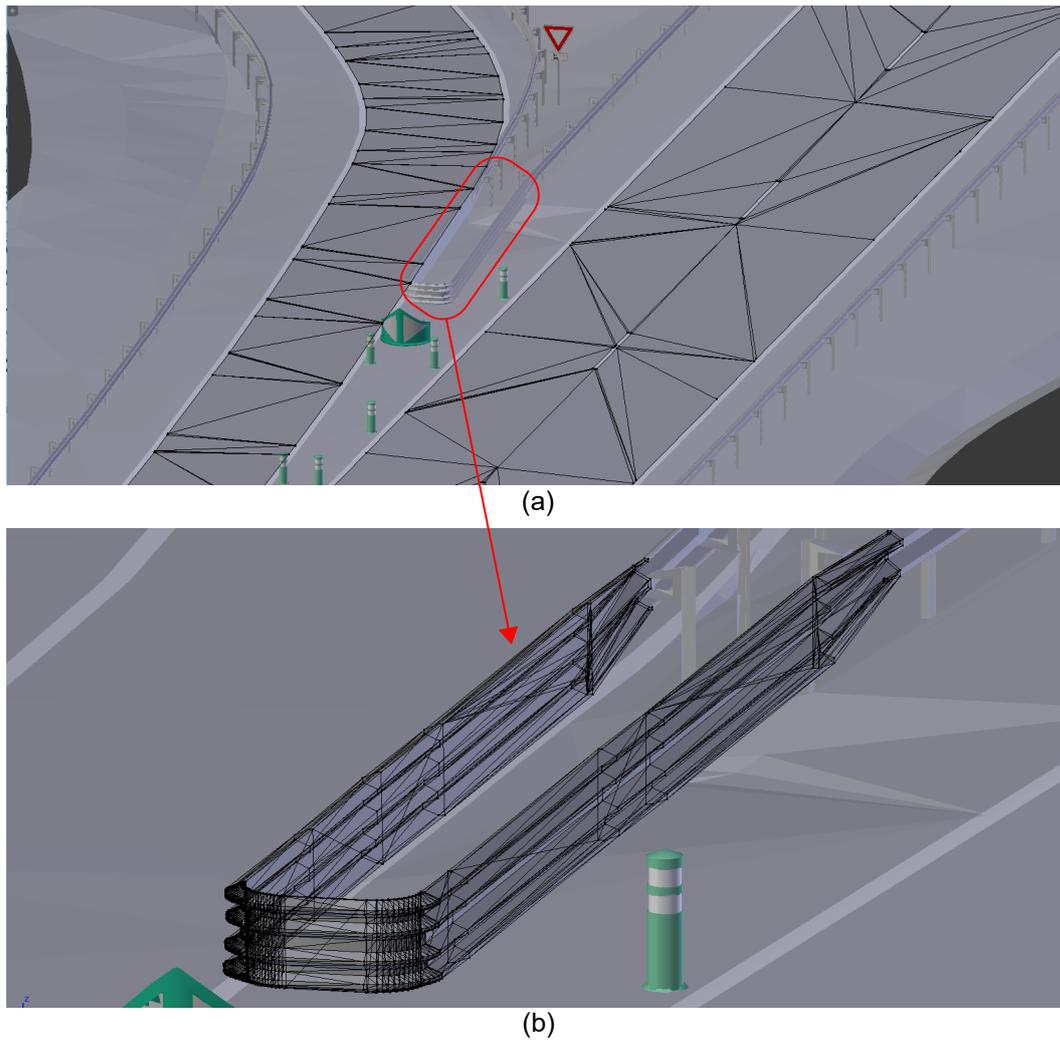


FIGURE 3.13 – Représentation de différents éléments maillés : Différentes résolutions spatiales sont représentées en fonction des éléments auxquelles elles correspondent et visibles grâce aux facettes apparentes (en noir) : (a) Route ; (b) Glissière de sécurité.

### 3.4.1.3 Référencement sémantique de la carte 3D.

Une fois le modèle 3D établi par la société TT Géomètres Experts, nous avons pu intervenir pour réaliser une troisième opération. Cette dernière étape de post-traitement est le référencement sémantique du modèle 3D.

Cette dernière opération permet d'introduire l'information sémantique dans la carte 3D. Chacune des facettes qui composent la carte 3D, et représentent les différents éléments, est annotée. Comme dans le cadre du référencement supervisé (Sec 3.2.1), cette opération est réalisée manuellement par des annotateurs. La tâche reste la même, les annotateurs sélectionnent une ou plusieurs facettes à la fois et leur associent un label correspondant à une classe sémantique. La différence est que le traitement est plus optimisé que sur des points 3D, les facettes couvrant une plus grande surface que les points. Les éléments modélisés en 3D par ces facettes correspondent plus à la réalité que des ensembles de points. Cela permet également de faciliter la tâche d'interprétation des annotateurs lors de l'attribution d'une classe sémantique.

Pour labelliser les facettes qui composent ce modèle 3D, nous avons hiérarchisé les classes en fonction des éléments cartographiés et présents dans le modèle 3D. Celle-ci est présentée (Fig 3.14) et se concentre sur les éléments fixes de l'environnement. Cette hiérarchie pourra évoluer lors de futurs travaux avec la prise en compte des éléments amovibles, non cartographiés dans la carte sémantique 3D et non utilisés au cours de ces travaux de thèse. Lors de l'opération de référencement du modèle 3D, les facettes ont été labellisées par un code numérique propre à chaque classe sémantique et plus facilement intégrable dans le format des nuages de points LiDAR utilisés.

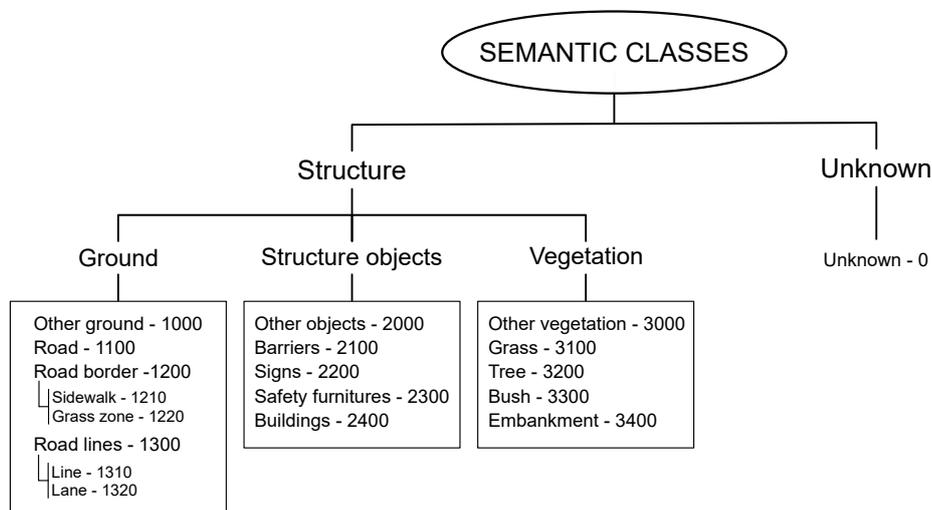


FIGURE 3.14 – Hiérarchie des classes sémantiques : Représentation des classes sémantiques prises en compte et associées à un label, sous forme de code numérique.

Ce procédé est fastidieux et consommateur en termes de temps et de moyens, mais reste plus aisé que l'annotation directe des données brutes produites par les capteurs (points 3D d'un nuage ou pixels d'une image). Le référencement manuel sur facettes de modèle 3D permet néanmoins de bénéficier de la précision des annotations grâce à l'intervention manuelle des annotateurs, tout en limitant le volume d'informations à traiter. Cette dernière opération que nous avons réalisé nous a permis d'obtenir la carte sémantique 3D représentant le domaine *MAP* et qui est utilisée comme source d'informations sémantiques par notre méthode de référencement.

## 3.4.2 Spécification des systèmes

### 3.4.2.1 Capteur de référence

Le capteur de référence correspond à un capteur de perception dont les données correspondent aux mesures du domaine  $CRA$ . Ce seront ces données qui seront augmentées grâce à la méthode de référencement présentée (Sec 3.3) et qui pourront ensuite être utilisées comme référence, lors de l'évaluation d'une solution de perception tierce testée. Pour éviter les biais lors des évaluations, ce capteur est indépendant de ceux utilisés par les solutions de perception évaluées.

Ces données incluent l'ensemble des éléments présents dans la scène lors des acquisitions de tests. Elles prennent également en compte l'ensemble des phénomènes extérieurs difficilement reproductibles par la simulation (conditions météorologiques, conditions d'éclairage,...), ainsi que des objets amovibles, non recensés sur la carte 3D, mais pouvant être présents lors des acquisitions. Il faut donc un capteur permettant de scanner l'environnement de façon précise et avec un certain niveau de résolution, disposant d'un champ de vision et d'une portée de mesure suffisamment importants pour couvrir ceux des solutions de perception testées.

Pour jouer ce rôle, nous avons orienté notre choix vers un capteur de type LiDAR, nommé  $L_{ref}$  pour différentes raisons. Comme on a pu l'étudier au cours du (Chap 2), ce type de capteur possède de nombreux avantages faisant de lui le capteur se rapprochant le plus du capteur idéal. Il permet de disposer d'une portée de mesure de l'ordre de 150 m, tout en proposant une précision de mesure de distances d'ordre centimétrique. Chaque mesure produit un point 3D, décorrélié du reste des mesures qui composent un scan LiDAR. A l'inverse d'une image capturée par une caméra, où l'ensemble de l'image peut être bruitée par un phénomène extérieur, ici seuls les points impactés par le phénomène seront bruités et non l'ensemble du scan. De plus, il s'agit d'un capteur actif, qui est moins sensible aux conditions lumineuses. Il permet d'enregistrer des données à n'importe quel moment de la journée, sans devoir appliquer de modification sur le paramétrage du capteur. Enfin, les données issues de ce type de capteur sont directement représentées dans un espace 3 dimensions. Ainsi, les erreurs et imprécisions, qui peuvent apparaître avec les différentes projections ou hypothèses de transformations, sont évitées.

### 3.4.2.2 Système de localisation

La dernière source de données est représentée par le système de localisation qui permet de géoréférencer, dans un système de référence global  $\mathcal{G}$ , les poses des différents capteurs embarqués ( $L_{ref}$  et capteurs associés aux solutions de perception testées  $T$ ). Ce système est notamment utilisé pour pouvoir positionner et orienter précisément  $L_{ref}$  pour chaque scan enregistré au cours de l'acquisition, noté  $scan_{réel,i}$ . Ces poses permettront de définir la transformation à appliquer pour géoréférencer ces données de mesure appartenant au domaine  $CRA$ , dans le système  $\mathcal{G}$ . Sous l'hypothèse que la carte 3D soit également géoréférencée dans  $\mathcal{G}$ , l'utilisation de ces poses permettra de positionner le modèle de capteur  $M_{L_{ref}}$  correspondant au LiDAR de référence, au sein de la carte 3D. Il sera alors possible de générer les scans de synthèse  $scan_{synth,i}$ , représentant le domaine  $MAP$ , lors de l'étape de référencement. Les mesures produites par ce système de localisation permettront ainsi de créer un lien entre ces domaines et permettra d'utiliser une méthode d'association de données géométriques.

Pour générer des scans  $scan_{synth,i}$  représentatifs des scans mesurés par le capteur de référence  $scan_{réel,i}$ , il faut que la position et l'orientation relatives du capteur modélisé  $L_{ref}$  par rapport à la carte sémantique 3D correspondent rigoureusement à la position et l'orientation relatives du capteur de  $L_{ref}$  par rapport à

l'environnement  $EE$ , comme illustré (Fig 3.9). D'après l'analyse réalisée au cours du (Chap 2), le système de localisation permettant de répondre facilement à ce besoin correspond à un couplage GPS-RTK/centrale inertielle. Le positionnement uniquement basé sur le GPS-RTK permet d'obtenir une précision de  $\pm 1$  cm sur le positionnement en  $(X, Y)$  et de  $\pm 5$  cm en  $Z$ . Selon le modèle de centrale inertielle exploité, il sera possible de conserver ces précisions sur le positionnement, mais également d'orienter le capteur avec une attitude composée par des angles de précision  $\pm 0.01^\circ$  [40]. Nous nous sommes alors orientés sur l'utilisation de ce système de localisation pour répondre aux besoins de géoréférencement de données de notre approche.

### 3.4.2.3 Calibration de la plateforme

Pour pouvoir utiliser conjointement les différents capteurs qui équipent une plateforme, une étape de calibration est nécessaire. Elle permet de définir, pour chaque capteur, les paramètres de transformation rigide entre le référentiel local du capteur et le référentiel  $Ego$ . Un exemple est représentée (Fig 3.15) avec une transformation rigide entre un référentiel  $\mathcal{L}$  propre à un capteur LiDAR et le référentiel  $Ego$  lié à la plateforme, tel que décrit au cours du (Chap 2).

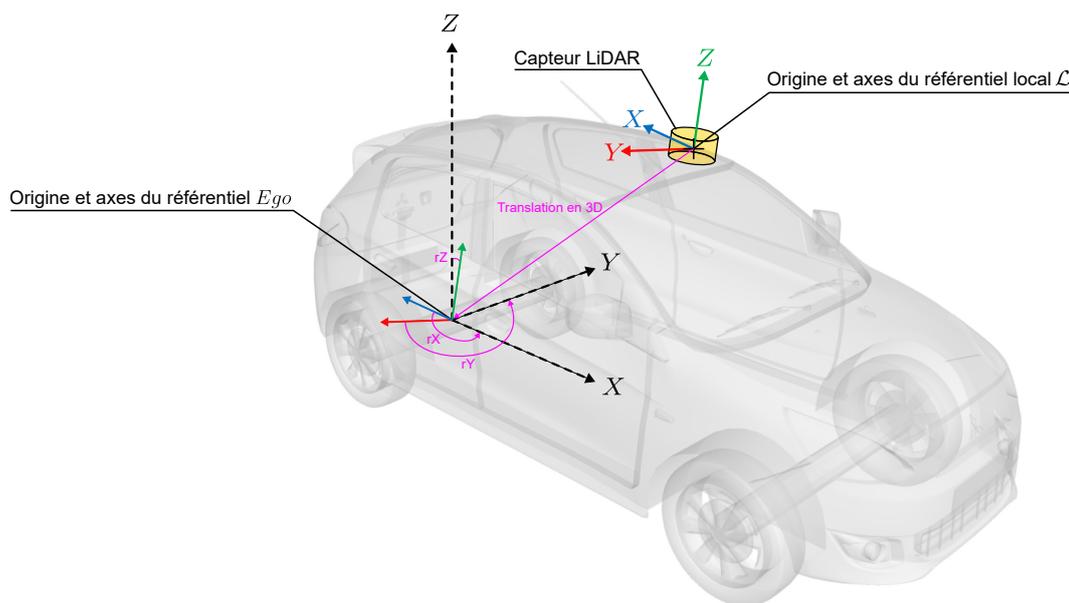


FIGURE 3.15 – Transformation rigide entre le référentiel local d'un capteur LiDAR  $\mathcal{L}$  et  $Ego$  : Représentation de la transformation rigide permettant de projeter les mesures de LiDAR, initialement exprimées dans le référentiel  $\mathcal{L}$ , vers le référentiel  $Ego$ .

La calibration de la plateforme MELO, utilisée au cours des expérimentations, a été opérée par levé topographique. Pour ce faire, un réseau de cibles a été mis en place dans la zone de calibration. Cet ensemble de cibles nous a permis de définir un référentiel local nécessaire au relevé des différents instruments équipant MELO. Pour réaliser ce levé, une station totale Trimble SX10, dont les précisions de mesures de distance et de mesures d'angles sont annoncées (Fig 3.17), a été utilisée.

La station totale a été positionnée sur 3 positions (Fig 3.16). Via le distance-mètre laser intégré à l'appareil, nous avons pu réaliser des mesures sur l'ensemble des éléments voulus : capteur de référence, capteurs associés aux solutions de perception testées, instruments composant le système de localisation et enfin, les extrémités de l'essieu arrière, afin de définir l'origine du référentiel  $Ego$ .

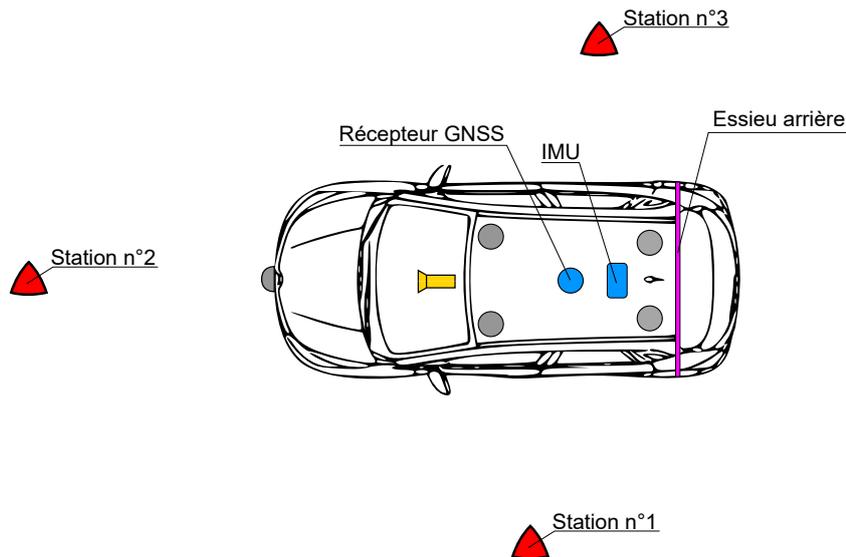


FIGURE 3.16 – Représentation du relevé : Représentation de l'organisation du relevé avec les stations utilisées représentées en rouge et les instruments ou éléments relevés en couleurs avec les différents LiDARs (en gris), la caméra MobilEye (en jaune), la centrale inertielle et le récepteur GNSS (en bleu) et l'essieu arrière (en magenta).



#### Trimble SX 10

Mesures en mode "Direct Reflex" (Mesure de distance laser)

Précision angulaire (azimutale et zénithale) : 1" (0.00027°)

Précision du distancemètre laser : 2mm +/- 1.5ppm

Portée de mesures : 1m-800m

FIGURE 3.17 – Trimble SX10 : Station totale utilisée pour mesurer les positions et orientations relatives entre les différents capteurs équipant la plateforme et le référentiel *Ego*.

Les mesures prises nous ont permis de positionner dans un référentiel local, l'ensemble des instruments équipant la plateforme.

Cet ensemble de mesures a permis d'estimer la position relative de chaque instrument vis-à-vis de l'origine du référentiel *Ego*. Cela nous a permis de définir, avec une précision millimétrique, les translations qui composeront les transformations à appliquer entre chaque instrument de MELO et le référentiel *Ego*.

Au cours de ce relevé, des points ont également été mesurés sur le sol, tout autour de la plateforme, afin de calculer un plan moyen représentant le sol. La multiplication des mesures, réalisées sur les différentes parties des instruments nous a permis d'estimer la hauteur et l'orientation de chaque capteur, vis-à-vis de ce plan moyen du sol.

Les estimations d'orientations pouvant être d'une faible précision, les angles de rotations selon les axes X, Y et Z du référentiel *Ego* ont par la suite été affinés. Nous avons utilisé les données produites au sein du jeu de données  $\mathcal{D}_3$  pour calculer des corrections de cette première estimation des angles. Ce jeu de données correspond à une acquisition menée à l'arrêt, sur une zone de test ayant un sol plat sur un grand périmètre. Les corrections ont alors été calculées progressivement, en superposant les données produites par les différents capteurs de perception embarqués, jusqu'à ce que ces données produites soient correctement alignées.

Grâce à ce procédé, nous avons pu estimer les paramètres de calibration extrinsèques ainsi que les transformations à appliquer pour exprimer entre le référentiel local, propre à chaque instrument et le référentiel *Ego*. Cela nous a permis de réunir l'ensemble des données produites sous un seul et même référentiel avant de les géoréférencer dans le système de référence  $\mathcal{G}$  utilisé.

### 3.4.3 Méthode de représentation des données

Enfin, il est également nécessaire de prévoir un moyen de représentation qui sera employé lors de traitement des données. Les différents moyens de représentations ont été introduits au cours du (Chap 2). Pour appliquer notre approche, le choix du capteur produisant les données de référence s'est orienté vers un LiDAR ( $L_{ref}$ ). On a également vu au cours du (Chap 2) que les LiDARs permettent de mesurer directement en 3D. Pour conserver un maximum d'informations à partir des mesures LiDAR, ce cadre tridimensionnel doit être conservé. Deux moyens de représentations sont alors compatibles avec le modèle de mesure de ce capteur : les nuages de points 3D et les octrees.

**Nuage de points 3D.** Il correspond à la représentation standard des données LiDAR. Il est au plus proches des mesures brutes de ce type de capteur. Les points qui composent un nuage de points peuvent conserver l'intégralité des informations, introduites (Chap 2) et associées à chaque point  $p$  composant le nuage :  $p_i = \{X, Y, Z, I, ring, time\}$ . Ce moyen de représentation évite également la perte de détails et d'informations qui peuvent résulter d'un changement de représentation. A titre d'exemple, lors du changement de représentation vers un octree, on contraint les données à une résolution  $r_{max}$  fixée pour l'octree, qui sera moins importante qu'initialement, où la résolution correspond à la largeur du faisceau laser. L'utilisation des nuages de points permet également de bénéficier des nombreux outils de traitements et algorithmes dédiés à ce moyen de représentation [94].

Ils ont cependant certaines limites d'usages. La première est que les données ne sont pas structurées, ni organisées. Les traitements locaux sur certaines parties d'un nuage ne sont pas possibles directement. Ils nécessitent de parcourir l'ensemble des points et d'analyser leurs informations pour identifier ceux qui sont concernés. Il s'agit donc d'un moyen de représentation non-optimisé, qui nécessite de nombreuses opérations de calculs, même pour réaliser des traitements simples, tels qu'un filtrage des points selon l'une des dimensions. Une autre limitation est que c'est un moyen de représentation rarement utilisé pour d'autres modalités de mesure que celle des capteurs LiDARs. Il est donc plus complexe d'utiliser ce moyen de représentation dans la cadre d'évaluation de solutions de perception basées sur diverses modalités de mesures.

**Octree.** Ce second moyen de représentation, introduit au cours du (Chap 2), est utilisé pour structurer des données 3D. Une implémentation des octrees est exploitable avec la librairie OctoMap [33], dont l'utilisation est répandue pour la représentation de données de perception 3D [95, 96]. Elle utilise ces structures en se basant sur un modèle probabiliste, dont les paramètres sont détaillés dans [33]. Cette librairie s'attache à la conversion de nuages de points 3D non-organisé vers des structures octrees et est efficace dans le découpage de l'espace, l'optimisation des données et la représentation implicite de l'espace libre et occupé. A la différence des nuages de points, cela amène la possibilité d'optimiser les traitements et de cibler localement des données pour les étudier et les traiter. Les octrees peuvent être comparés à une adaptation tridimensionnelle des grilles d'occupations, incluant différents niveaux d'échelles (les niveaux de profondeurs) et donc différentes résolutions. L'utilisation de grille est un moyen efficace et récurrent pour gérer les systèmes multi-modaux [97, 28].

La représentation des données en octrees, qui sont structurés sur la base d'une grille 3D, est donc plus adaptée à la multi-modalité que les nuages de points. Un dernier atout de ce moyen de représentation est d'exploiter différents niveaux de profondeur dans l'arborescence. Cela permet de pouvoir travailler avec des données à différentes résolutions.

Certaines limitations sont également retrouvées avec ce moyen de représentation, notamment liées à la perte d'information vis-à-vis des nuages de points 3D. Ces pertes d'informations sont liées à deux aspects. Le premier dépend de la structure des voxels qui occupent les octrees. Ils sont représentés numériquement par un nœud, généralement associé à deux attributs introduits (Chap 2) :

1. *key* : identifiant propre à chaque voxel permettant de retrouver directement des informations sur ce voxel (coordonnées de son centre, sa position au sein de l'arborescence de l'octree, sa taille, etc.) ;
2. *value* : correspondant à la valeur de probabilité que le voxel soit occupé et qui est calculée en fonction des paramètres du modèle de capteur défini.

On perd donc une partie des informations contenues initialement dans les points  $p_i$  qui composent un nuage 3D. Le second aspect est lié au compromis à réaliser entre le besoin d'optimisation et la résolution souhaitée. On l'a vu (Chap 2), la résolution maximale d'une octree dépend de la valeur du paramètre  $r_{max}$  définie. Une faible résolution apporte une plus grande optimisation mais amène à une perte de détails et une simplification de la scène représentée. À l'inverse, si  $r_{max}$  est défini pour une résolution élevée (par exemple à la taille d'un point  $\simeq 1mm$ ), on conservera les détails de la scène, mais l'optimisation sera moins importante lors des traitements effectués. De plus, l'usage en temps réel sera limité par l'étape de conversion permettant de structurer les nuages non organisés vers des structures octrees. En effet, comme le soulignent différents travaux [96, 98], le traitement de scans composés de plusieurs milliers de points, tout en utilisant une résolution  $r_{max}$  élevée, peut se révéler gourmand en termes de calculs. Par conséquent, la conversion d'un scan en octree peut prendre plusieurs secondes, voire dizaine de secondes, en fonction des données traitées et de la résolution demandée.

### 3.5 Synthèse et discussions

Au cours de ce chapitre, nous avons pu définir notre concept, présenter l'approche méthodologique qui réside dans ce concept et introduire les travaux préparatoires réalisés pour permettre le fonctionnement de l'approche. Cette approche est représentée (Fig 3.18) et est utilisée pour évaluer les performances de différentes solutions de perception prête à l'emploi. Pour mener à bien ces évaluations, il est nécessaire de disposer d'une référence à laquelle comparer les données obtenues en sortie des solutions testées.

L'étape nommée « Référencement automatique des données » répond à ce besoin en produisant automatiquement cette référence, grâce à l'utilisation conjointe de différentes sources de données. La première est un capteur de référence. Son rôle est de produire des données représentant l'environnement dans lequel sont menés les tests. Une seconde source de données correspond à une carte sémantique 3D correspondant à ce même environnement. Elle intègre l'ensemble des informations sémantiques le caractérisant et sert de source d'informations. Enfin, un système de localisation est utilisé pour géoréférencer les données produites par le capteur de référence. Cela permet de les positionner vis-à-vis de l'environnement, et donc, vis-à-vis de la carte sémantique 3D. Grâce à une association géométrique, les informations sémantiques, initialement contenues dans la carte sémantique 3D, sont transférées sur les données acquises par le capteur de référence. Cela permet de les annoter et de disposer de la référence recherchée pour mener les évaluations.

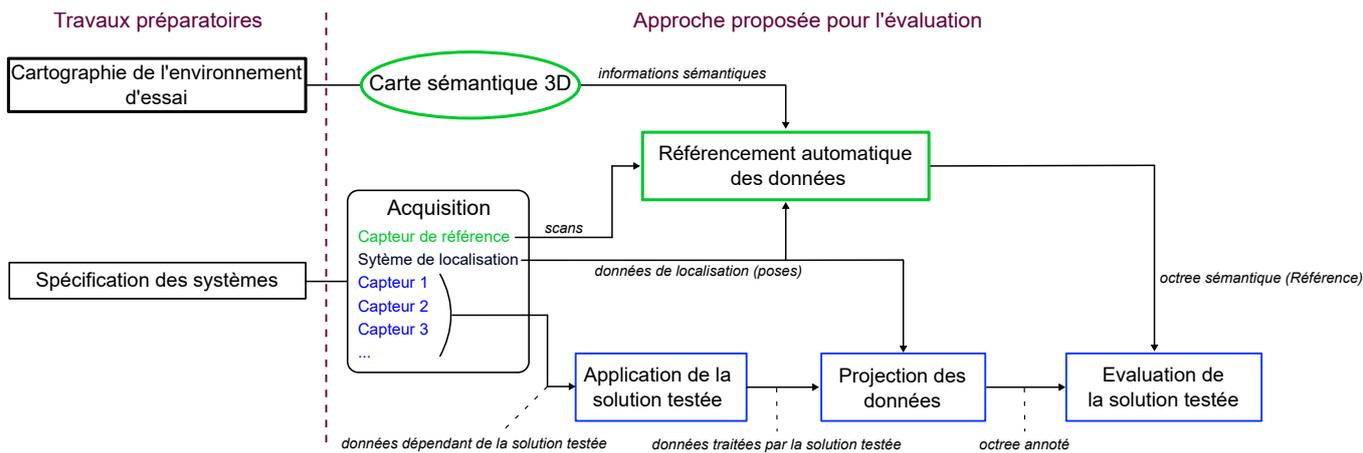


FIGURE 3.18 – Distinction entre référencement et évaluation : Représentation de l'approche proposée avec une distinction entre les éléments associés à la production de la référence (en vert) et ceux associés à l'évaluation des solutions de perception (en bleu).

### 3.5.1 Positionnement de l'approche

Pour positionner ces travaux vis-à-vis de la littérature scientifique, ils se basent sur une approche semi-supervisée, comparable à celles introduites dans les travaux [12] et [99]. Différentes similarités peuvent être retrouvées avec ces approches et celle que nous proposons. Dans [12], on retrouve l'utilisation d'un capteur LiDAR dédié uniquement à la production de la référence, comparable à notre capteur de référence. Les données produites par ce capteur et référencées sont ensuite utilisées pour évaluer des données issues de capteurs indépendants et pouvant reposer sur d'autres méthodes de perception. La différence principale avec nos travaux est que les données de référence sont produites par un algorithme de traitement. Les données produites par le capteur de référence sont ainsi traitées par un algorithme assumé plus performant que celui intégré dans la solution de perception testée. Cette approche peut alors être limitante sur le type de solutions de perception testées ou sur leur complexité.

L'approche que nous proposons peut aussi être comparée à celle proposée dans [99]. On y retrouve la désignation d'un environnement de test en amont. Cet environnement est cartographié en 3D grâce à l'utilisation du LiDAR. Le nuage de points produits représentant l'ensemble de l'environnement est annoté sémantiquement avant d'être structuré sous forme d'Octree. Cet Octree sémantique représente alors la référence. Les solutions de perception exploitant des données en 3 dimensions peuvent être évaluées en exploitant cet Octree. Pour ce faire, une méthode d'alignement est utilisée pour aligner les données produites par ces solutions avec cet Octree de référence. Par association géométrique des données issues de la solution testées sont alignées sur l'Octree et récupèrent l'information sémantique des voxels dont le volume inclut les données traitées. L'évaluation est finalement opérée en vérifiant la cohérence entre les informations sémantiques obtenues par association géométrique sur les voxels et celles issues des prédictions faites par la solution testée. On retrouve donc plusieurs similarités avec notre approche : l'utilisation des structures Octree, la définition d'un environnement de test cartographié puis annoté sémantiquement, ainsi qu'une méthode d'association automatique pour produire des données de référence. Notre approche se différencie cependant sur plusieurs aspects.

Dans les travaux [99], l'environnement d'essai cartographié, puis représenté sous forme d'Octree sémantique, est la seule source de données dédiée uniquement au référencement. Or, ces données ne peuvent évoluer dans le temps et correspondent aux conditions d'acquisition spécifiques, propres au moment où l'environnement a été cartographié. Par la suite, ce même Octree est réutilisé pour annoter les nouvelles données issues des solutions de perception testées.

Dans [99], aucun capteur de référence n'est utilisé. Seules les données issues des solutions de perception testées sont traitées pour les associer à des labels considérés vrais au sein de l'Octree sémantique représentant l'environnement. Au-delà des biais d'évaluation pouvant apparaître lorsqu'on utilise, pour référence et données de test, les mêmes données, plusieurs limites peuvent apparaître :

Une première limite est que les données issues des solutions de perception évaluées ne sont pas acquises en même temps que celles ayant permis de produire l'Octree de l'environnement. Cet environnement peut évoluer et le fait de ne pas avoir les mêmes conditions d'acquisition peut impacter la pertinence des évaluations. Une autre limite liée au fait de n'utiliser que cet Octree sémantique lors du référenciement est de ne pas pouvoir créer de scénarios faisant intervenir des éléments dynamiques. Tous les éléments non-présents dans l'environnement initialement, mais capturés lors des acquisitions avec les solutions de perceptions testées pourront amener des erreurs d'évaluation, impactant négativement les performances évaluées.

Notre approche se différencie en faisant intervenir le capteur de référence. Ce capteur permet de capturer des données acquises dans les mêmes conditions (météo, luminosité, etc) que les capteurs associés aux solutions de perception testées. Les données issues de ce capteur de référence sont traitées indépendamment et servent de référence lors des évaluations. Cela permet d'éviter les biais d'évaluation évoqués. De plus, la référence produite via l'approche que nous proposons, ne repose pas uniquement sur la carte sémantique 3D, mais intègre aussi des données acquises par le capteur de référence. Cela permet de prendre en compte certaines variations vis-à-vis des conditions d'acquisition ou des évolutions de l'environnement d'essai par rapport à sa cartographie initiale. Les données représentant ces phénomènes peuvent alors être mises de côté, ou faire l'objet de traitements approfondis, tous en évitant d'impacter sur la pertinence des performances mesurées lors de l'évaluation. Cette possibilité ouvre la perspective à des acquisitions faisant intervenir des éléments dynamiques lors des acquisitions de test. Ces éléments, bien qu'ils ne soient présents dans la carte sémantique 3D, pourront être capturés par le capteur de référence. Les données correspondant à ces éléments pourront alors faire l'objet de traitements spécifiques, afin d'être référencées correctement. Il sera alors possible d'évaluer des solutions de perception ciblées sur les objets dynamiques comme pour le suivi de cibles en mouvement (piétons, véhicules, etc).

### 3.5.2 Application de la méthodologie

Dans la suite de cette thèse, la mise en application de l'approche sera séparée en deux parties majeures distinguées (Fig 3.18). Le (Chap 4) sera associé au référenciement des données, tandis que le (Chap 5) sera associé à l'évaluation de solutions de perception.

Le (Chap 4) sera donc ciblé sur la production des données de référence et permettra de présenter, de façon approfondie, la mise en application de la méthode introduite au cours de ce chapitre. Les mécanismes méthodologiques, implémentés au cours de ces travaux, pour automatiser le référenciement de données acquises en conditions réelles, seront détaillés. Ils permettront de comprendre comment est opéré le transfert d'informations entre les données du domaine *MAP* et celles du domaine *CRA*. La structure des données obtenues à la suite du référenciement sera présentée avant d'être utilisée dans le cadre du (Chap 5).

Le (Chap 5) sera dédié à l'évaluation des solutions de perception. On présentera les étapes permettant d'exploiter conjointement les données référencées grâce à la méthode présentée (Chap 4) avec celles obtenues en sortie d'une solution testée. Cela permettra de confronter ces données et d'évaluer les solutions testées sur la base de critères de performances présentés au cours de ce chapitre. L'introduction de plusieurs solutions de perception basée sur des principes de fonctionnement et des algorithmes de traitement différents, ensuite évaluées, permettra de mettre en avant la flexibilité de l'approche proposée.

### 3.5.3 Validation méthodologique

Dans l'objectif d'apporter des preuves de concept de notre approche, différentes expériences seront menées au sein de ces deux chapitres. Dans le cadre du (Chap 4), les tests et expériences menés auront pour objectif d'analyser les capacités de la méthode de référencement ainsi que sa robustesse face aux imprécisions dans les données de localisation, ces dernières jouant un rôle important dans le référencement. Dans le (Chap 5), les expériences résident dans les évaluations des solutions de perception introduites. Les résultats obtenus suite aux évaluations menées par notre approche seront alors analysés pour interpréter le comportement des solutions et voir s'il est possible d'en tirer des conclusions.

Pour mener ces expériences, plusieurs jeux de données ont été enregistrés à l'aide de la plateforme MELO (Fig 3.19). Cette plateforme intègre différents capteurs correspondant à différentes modalités. Plusieurs LiDARs équipent MELO, ce qui permet de dédier un des LiDAR comme capteur de référence  $L_{ref}$  et d'utiliser les autres LiDARs dans le cadre de solution de perception à tester. Cela permet de garantir une indépendance totale entre les données utilisées comme référence et celles produites par les solutions testées, et ainsi, d'éviter les biais de mesures des performances lors des évaluations.



FIGURE 3.19 – MELO : Plateforme du Groupe Renault et utilisée pour les acquisitions en conditions réelles, au sein de l'environnement d'essai  $EE$ , correspondant à la carte sémantique 3D.

Trois jeux de données, nommés  $\mathcal{D}_1$ ,  $\mathcal{D}_2$  et  $\mathcal{D}_3$ , ont été acquis. Leurs caractéristiques sont présentées dans le tableau de synthèse (Tab 3.2). Les expériences menées au cours du (Chap 4) seront basées sur ces jeux de données. Cela permettra de s'appuyer sur des données acquises en conditions réelles et intégrant l'ensemble des phénomènes extérieurs difficilement reproductible en simulation.

Les expériences présentées au cours du (Chap 5) seront basées sur  $\mathcal{D}_2$  car il représente le jeu de données le plus complet. En appliquant l'approche décrite dans ces travaux, trois solutions y seront évaluées. Deux d'entre elles sont destinées à détecter les points appartenant à la route au sein de nuages de points acquis par des LiDARs. La troisième solution s'appuie sur la vision. Elle est utilisée pour détecter les lignes des marquages au sol au sein d'images produites par une caméra monoculaire.

|   |   | Jeux de données   |   |  |
|---|---|---|---|--|
| <b>Informations générales</b>   |   |   |   |  |
| Nom   | $\mathcal{D}_1$   | $\mathcal{D}_2$   | $\mathcal{D}_3$   |  |
| Année d'acquisition   | 2018  | 2019  | 2019  |  |
| Vitesse d'acquisition moyenne   | 40km/h  | 70km/h  | A l'arrêt - 0km/h   |  |
| Distance parcourue  | 2.149km   | 2.315km   | 0km   |  |
| Durée   | 214s  | 118s  | 54s   |  |
| <b>Configuration de la plateforme</b>   |   |   |   |  |
| Capteurs de perception<br>(Position / Fréquence de mesure $F$ )                               | 2 LiDARs 32 nappes<br>(Avant du toit / $F = 20Hz$ )<br>2 LiDARs 16nappes<br>(Arrière du toit / $F = 20Hz$ )<br>1 LiDAR 32 nappes<br>(Pare choc avant / $F = 10Hz$ ) | 2 LiDARs 32 nappes<br>(Avant du toit / $F = 10Hz$ )<br>2 LiDARs 16nappes<br>(Arrière du toit / $F = 10Hz$ )<br>1 LiDAR 32 nappes<br>(Pare-chocs avant / $F = 10Hz$ )<br>1 Caméra MobilEye<br>(Pare-brise avant / $F = 36Hz$ ) | 2 LiDARs 32 nappes<br>(Avant du toit / $F = 10Hz$ )<br>2 LiDARs 16nappes<br>(Arrière du toit / $F = 10Hz$ )<br>1 LiDAR 32 nappes<br>(Pare choc avant / $F = 10Hz$ ) |  |
| Système de localisation<br>Fréquence de mesure $F$<br>Ecart-types de positionnement [min;max] | GNSS-RTK / IMU<br>$F = 10Hz$<br>[3.46cm ; 28.58cm]  | GNSS-RTK / IMU<br>$F = 10Hz$<br>[3.46cm ; 5.20cm]   | N/A<br>N/A<br>[N/A ; N/A]   |  |

TABLE 3.2 – Synthèse des jeux de données utilisés : Tableau de synthèse des caractéristiques propres à chacun des jeux de données  $\mathcal{D}_1$ ,  $\mathcal{D}_2$  et  $\mathcal{D}_3$ .

## 3.6 Conclusion

Au cours de ce chapitre, nous avons présenté les études et discussions menées pour permettre la mise en place du concept final élaboré au cours de cette thèse et qui correspond à notre approche de validation. Une combinaison de quatre phases ont permis de parvenir à ce concept final :

- La première est l'étude des méthodes de référencement des données. Nous avons pu voir que la production des données de référence est un élément clé pour permettre de disposer d'évaluations pertinentes et flexibles à divers scénarios. Cette étude nous a permis de d'identifier les forces et limitations propres aux différentes approches trouvées dans la littérature et de les prendre en considération au cours de la mise en place de notre concept ;
- La seconde phase à consister à établir une méthode permettant de référencer des données efficacement et de façon précise. Le concept final s'inspire donc des différentes méthodes étudiées dans le but d'associer différentes forces associées aux approches étudiées lors de la première phase. Pour cela, le concept s'appuie sur un référencement automatique qui s'inspire de l'adaptation de domaine pour opérer un transfert entre des données de synthèse annotées et les données acquises en conditions réelles qui, après référencement, servent de référence pour évaluer les solutions de perception testées ;
- Nous avons ensuite pu discuter des moyens concrets utilisés pour appliquer ce concept sur des données acquises au cours de ces travaux de thèse. Les points méthodologiques devant être clarifiés et expérimentés pour disposer d'une preuve de concept sont énoncés et seront détaillés au cours du (Chap 4) ;
- Une fois le concept défini, les réflexions ont porté sur sa mise en application. Des discussions ont été menées sur les moyens techniques à mettre en place pour permettre d'appliquer la méthode de référencement et opérer ce transfert d'informations sémantiques, entre les données de synthèse et les données acquises en conditions réelles.

A l'issue de ce chapitre, nous disposons d'un concept défini, destiné au référencement des données. Il se base sur une méthode innovante qui permet de faciliter la production de données de référence afin d'évaluer différentes solutions de perception, dans divers scénarios. Dans le chapitre suivant, nous reviendrons plus en détail sur les différentes étapes qui composent cette méthode de référencement. Des expériences permettront la mise en application de cette méthode sur un échantillon des données acquises. Elles permettront d'étudier les capacités de référencement de ce concept proposé.



## Chapitre 4

# Référencement automatique des données

### Sommaire

---

|       |  |     |
|-------|--|-----|
| 4.1   | Introduction . . . . .   | 71  |
| 4.2   | Prédiction des annotations . . . . .                             | 72  |
| 4.2.1 | Méthode et outil de simulation . . . . .                         | 73  |
| 4.2.2 | Modélisation du capteur de référence . . . . .                   | 74  |
| 4.2.3 | Utilisation des sources de données . . . . .                     | 75  |
| 4.3   | Transfert d'informations sémantiques . . . . .                   | 78  |
| 4.3.1 | Méthodes d'association de données . . . . .                      | 78  |
| 4.3.2 | Caractéristiques de la méthode choisie . . . . .                 | 85  |
| 4.4   | Validation de la méthodologie de référencement . . . . .         | 94  |
| 4.4.1 | Analyse de la méthode proposée . . . . .                         | 94  |
| 4.4.2 | Analyse du modèle paramétrique du capteur de référence . . . . . | 95  |
| 4.4.3 | Analyse du référencement . . . . .                               | 100 |
| 4.5   | Perspectives . . . . .   | 109 |
| 4.5.1 | Critères d'association . . . . .                                 | 109 |
| 4.5.2 | Utilisation du géoréférencement . . . . .                        | 110 |
| 4.6   | Conclusion . . . . .   | 111 |

---

## 4.1 Introduction

Au cours du (Chap 3), nous avons pu introduire le concept méthodologique dédié à l'évaluation des performances de solutions de perception. Celui-ci a été développé dans le cadre de cette thèse. Nous avons également introduit les éléments fondamentaux que ce concept nécessite, ainsi que les travaux préliminaires nécessaires à sa mise en place. Les traitements de données résidant au sein de ce concept peuvent être décomposés en deux blocs méthodologiques majeurs :

1. Le référencement de données, nécessaire pour disposer de données de référence auxquelles comparer les données de la solution tierce évaluée ;
2. Le processus d'évaluation, permettant de mener cette comparaison et d'évaluer les performances de la solution sur la base de critères définis.

La suite de cette thèse s'attelle à présenter et détailler les blocs méthodologiques composant ce concept. Au sein de ce chapitre, le premier bloc développé correspond au référencement automatique. Il a pour objectif de permettre aux données issues du capteur de référence d'être annotées et de pouvoir être utilisées comme référence lors des évaluations.

L'étude réalisée dans le chapitre précédent (Chap 3) sur le référencement de données de perception nous a permis de mettre en évidence les avantages et les limites persistantes des approches existantes. Nous avons ensuite conceptualisé une approche qui a pu être développée et qui est présentée de façon détaillée. Pour ce faire, cette approche représentée par l'étape « Référencement automatique des données » (Fig 4.1-a), est décomposée en deux sous-étapes introduites sur la figure (Fig 4.1-b) et qui sont détaillées au cours de chapitre.

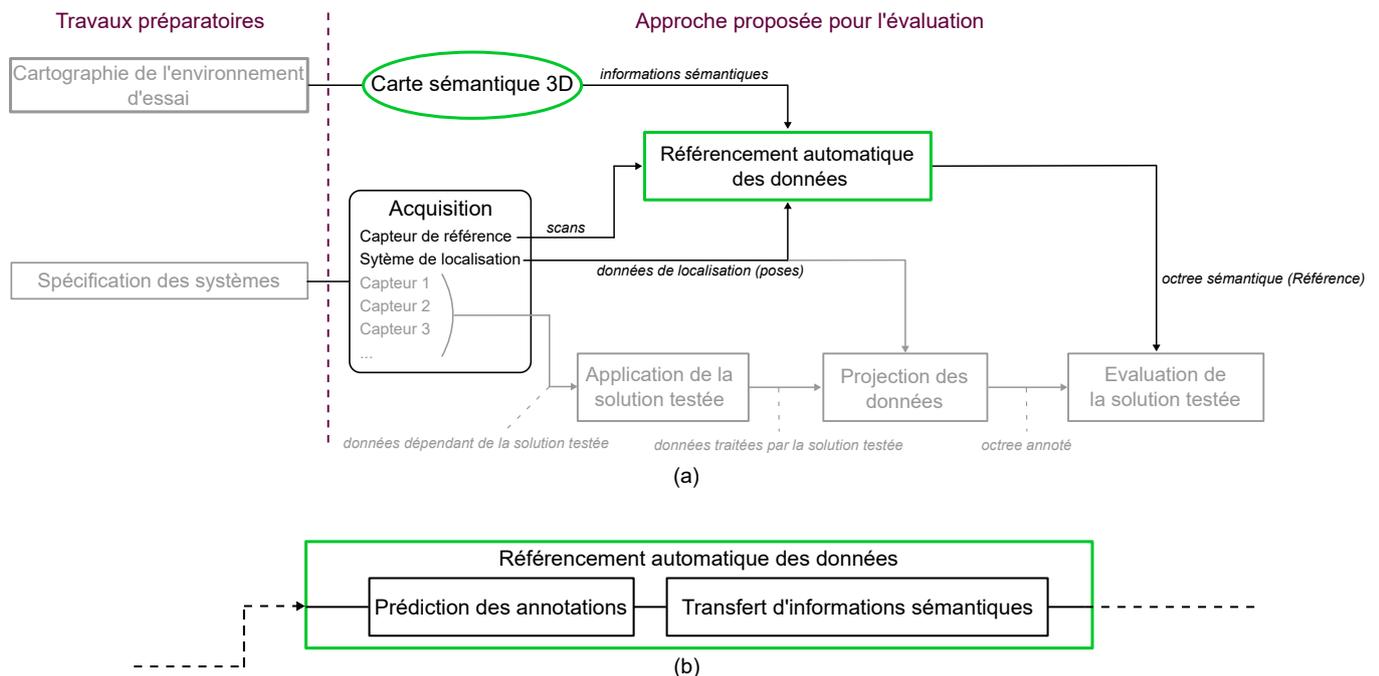


FIGURE 4.1 – Sous-étapes du référencement automatique : (a) Représentation de la méthodologie globale ; (b) Sous-étapes qui composent le bloc « Référencement automatique des données » de la méthodologie globale.

Ce chapitre s'intéresse également au moyen de représentation exploité communément entre les données de synthèse et les données acquises en conditions réelles, celui-ci jouant un rôle dans la sous-étape de « Transfert d'informations sémantiques ». Enfin, l'intégrité de cette méthode de référencement est évaluée en analysant les points clés qui composent son fonctionnement au travers des deux sous-étapes.

## 4.2 Prédiction des annotations

La première sous-étape qui compose le référencement automatique des données est la « Prédiction des annotations » (Fig 4.2). Elle représente la partie s'articulant autour de la génération de données synthétiques annotées. Elles sont obtenues grâce à la modélisation du capteur de référence  $L_{ref}$ , aux informations sémantiques contenues dans la carte 3D représentant  $EE$  ainsi qu'aux mesures de positionnement enregistrées lors de l'étape « Acquisition de données », introduits au cours du (Chap 3).

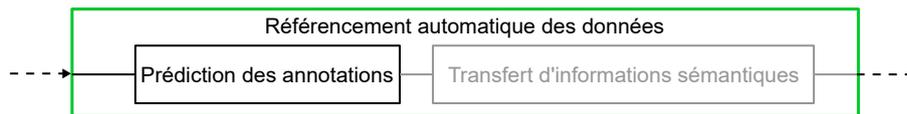


FIGURE 4.2 – Sous-étape de la prédiction des annotations.

### 4.2.1 Méthode et outil de simulation

L'avantage d'utiliser un outil de simulation est double. Premièrement, la production de données de synthèse permet de disposer d'un support pour les informations sémantiques contenues dans la carte sémantique 3D du domaine  $MAP$ . Ensuite, l'utilisation de données de synthèse, obtenues à partir d'un modèle de capteur représentatif du capteur utilisé en conditions réelles, ainsi qu'un moyen de représentation similaire à celui utilisé pour les données acquises en conditions réelles, permet d'avoir des données synthétiques comparables aux données « réelles » (Fig 4.6). Cela permet de simplifier et d'optimiser le transfert d'informations entre ces deux types de données.



FIGURE 4.3 – Logo de l'outil de simulation BlenSor [100]

Pour mener à bien la génération de données de synthèse, nous avons orienté notre choix sur l'outil BlenSor [100]. Cet outil se distingue par une attention portée sur les propriétés spécifiques aux capteurs LiDARs vis-à-vis des simulateurs plus génériques tels que MORSE [101], Gazebo [102] ou Webots [103]. Pour ce faire, BlenSor utilise le ray tracing (ou lancer de rayon). Il s'agit d'une technique de calcul d'optique par ordinateur permettant de simuler le parcours inverse de la lumière [86]. En exploitant, ce principe, il est possible de simuler le trajet réalisé par les faisceaux lasers émis par un capteur LiDAR. Pour pouvoir être exploité, l'outil nécessite deux éléments :

**Modèle de capteur :** Le premier élément est un modèle de capteur  $M_{L,blensor}$ , sur lequel le simulateur va s'appuyer pour contraindre le ray tracing. Il s'agit d'un modèle sphérique paramétrique tel qu'introduit (Chap 2) et défini par des paramètres intrinsèques et extrinsèques, ces derniers étant obtenus par calibration de la plateforme (Chap 3). L'objectif est d'avoir  $M_{L,blensor}$  qui se rapproche le plus possible du modèle  $M_{L,ref}$  correspondant au capteur LiDAR utilisé. Plus le modèle  $M_{L,blensor}$  sera proche du modèle  $M_{L,ref}$ , plus les données de synthèses générées seront susceptibles d'être représentatives des données produites par le capteur  $L_{ref}$  lors d'acquisitions en conditions réelles.

**Environnement simulé :** Le second élément correspond à l'environnement simulé dans lequel on souhaite utiliser le modèle de capteur pour générer des données de synthèse. Cet environnement est retranscrit sous la forme d'un modèle 3D surfacique. Dans notre cas, ce modèle 3D correspond à la carte sémantique 3D introduite au cours du (Chap 3). Cette carte représente l'environnement d'essai  $EE$  sous la forme d'un modèle 3D, représentatif et fidèle à l'environnement réel d'acquisition. En exploitant le ray tracing et les paramètres de  $M_{L,blensor}$ , le simulateur calcule des points d'intersection entre les faisceaux simulés et les facettes qui composent le modèle 3D. Ces points d'intersection constituent les scans qui composent les données de synthèse (Fig 4.4).

La combinaison d'un modèle de capteur synthétisé  $M_{L,blensor}$  proche du modèle du capteur réel, et de l'utilisation d'un environnement simulé représentatif de l'environnement réel  $EE$ , renforce l'hypothèse que les données générées seront représentatives de celles obtenues lors d'acquisitions réelles. Cela permettra de faciliter le transfert d'informations sémantiques lors de la seconde sous-étape du « Référencement automatique des données » (Fig 4.2).

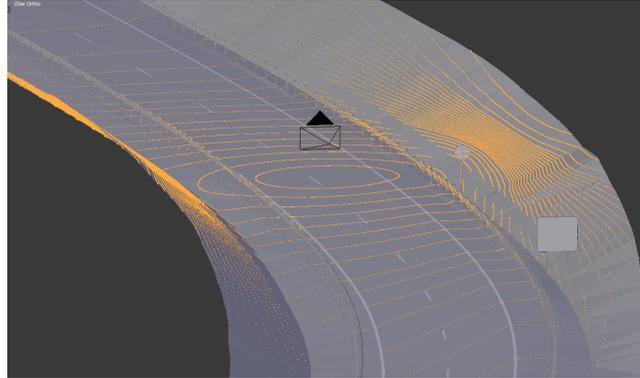


FIGURE 4.4 – Scan de synthèse : nuage de points généré via BlenSor. Les points représentés en orange aux intersections faisceau-facette sont calculés par ray tracing en fonction du modèle de capteur défini dans le simulateur. Le symbole « type caméra » (en noir) est utilisé pour représenter la position du centre du capteur synthétisé. Ce symbole est également orienté en fonction des paramètres extrinsèques définis pour le capteur synthétisé.

BlenSor est développé en langage Python [104] et est open-source. Cela apporte plusieurs avantages liés à la configuration des capteurs, au format de données ou à l'automatisation des traitements comme on pourra le voir dans les (Secs 4.2.2 et 4.2.3).

## 4.2.2 Modélisation du capteur de référence

### 4.2.2.1 Gestion des modèles de capteurs

Du point de vue du simulateur BlenSor [100], les travaux ont été orientés pour l'utilisation de modèles de capteurs basés sur les paramètres intrinsèques de capteurs réels existants. Ces modèles permettent de générer des données de synthèse représentatives des données acquises en conditions réelles. Un premier avantage de BlenSor est qu'il est possible d'accéder aux fichiers de configuration qui contiennent les modèles des capteurs synthétisés. Ces fichiers sont accessibles et peuvent être paramétrés. Il est possible de modifier les modèles de capteur déjà référencés ou d'ajouter un nouveau modèle de capteur. Dans le cadre des capteurs LiDAR, les modèles  $M_{L_{synth}}$  prédéfinis correspondent à des modèles sphériques paramétriques introduits (Chap 2). Pour définir le modèle associé à un capteur LiDAR, les paramètres intrinsèques composant  $M_{L_{synth}}$  et pouvant être définis dans ces fichiers de configurations sont les suivants :

- Angles d'élévation associé à chaque nappe  $\mathcal{V}_{ring}$  ;
- La résolution azimutale  $r_{Az}$  ;
- Fréquence de rotation du capteur  $F$  ;
- Distance maximale mesurable par le capteur de synthèse  $D_{max}$  ;
- Les bruits de mesure de distance :  $\epsilon_{bias,i}$  et  $\epsilon_{mesure}$  ;
- Le type de bruit (Distribution Gaussienne ou Distribution de Laplace).

Comme on peut le constater par ces modèles de capteurs LiDAR utilisés, BlenSor considère les paramètres intrinsèques liés aux angles des faisceaux synthétisés comme parfaits. Cela signifie que de façon native, les données synthèse sont générées sans qu'aucun bruit de mesure ne soit considéré sur les angles d'élévation  $\mathcal{V}_{ring}$  ou sur la résolution azimutale  $r_{Az}$ .

En revanche, les mesures de distance peuvent être bruitées par un modèle d'erreur composé de deux bruits via l'équation (Eq 4.1).

$$D_{bruitée} = D_{simulée} + \epsilon_{biais,ring} + \epsilon_{mesure} \quad (4.1)$$

Les bruits de mesure sont représentés par  $\epsilon_{biais,ring}$  et  $\epsilon_{mesure}$ . Ils suivent une loi de distribution normale centrée réduite  $\mathcal{N}(0, \sigma)$  où  $\sigma$  représente la variance. Le premier bruit de mesure  $\epsilon_{biais,ring}$  est propre à chacun des faisceaux. Le second bruit de mesure  $\epsilon_{mesure}$  repose sur le fait que chaque mesure en elle-même peut être sujette à un certain bruit de mesure identique sur chacune des mesures. Cependant, aucune valeur de variance ou méthode de calcul n'est décrite pour paramétrer ces biais de mesure. Cela doit être mis en place par l'utilisateur.

Les paramètres extrinsèques sont quant à eux définis directement lors de l'étape de génération des données présentées (Sec 4.2.3).

#### 4.2.2.2 Structure des données de synthèse

L'accès au code source nous apporte un second avantage en permettant de modifier la structure initiale des données générées par BlenSor. Les modifications apportées permettent de générer des scans de synthèse avec un format identique à celui utilisé pour les données acquises en conditions réelles et introduit au (Chap 3). Les points  $p_{synth,i}$  qui composent les scans de synthèse sont ainsi composés par les mêmes attributs  $p_{synth,i} = \{X, Y, Z, I, ring, time, label\}$ . L'homogénéisation du format des données renforce encore la facilité d'opérer un transfert d'informations sémantiques dans un second temps.

### 4.2.3 Utilisation des sources de données

#### 4.2.3.1 Génération des données de synthèse

Au cours de cette première sous-étape, deux sources de données sont exploitées conjointement pour générer les données de synthèse : la carte sémantique 3D représentant l'environnement d'essai  $EE$  et le système de localisation. L'objectif est de générer des données de synthèse qui seront associées aux données produites par le capteur de référence  $L_{ref}$ , suivant un critère d'association choisi ou proposé, discuté dans la suite de ce chapitre (Sec 4.3).

**Carte sémantique 3D.** La carte sémantique 3D joue deux rôles au sein du simulateur. Elle est utilisée à la fois comme environnement simulé représentant le domaine  $MAP$ , ainsi qu'en tant que source d'informations sémantiques, grâce aux annotations apportées lors de son élaboration (Chap 2).

**Système de localisation.** Les mesures de localisation enregistrées lors de l'acquisition dans le domaine  $CRA$  sont utilisées pour paramétrer extrinsèquement le modèle de capteur synthétisé  $M_{L_{synth}}$ , au sein de l'environnement simulé : la carte sémantique 3D. Lors de l'acquisition en conditions réelles, chaque scan LiDAR ( $scan_{réel,i}$ ) produit par le capteur de référence ( $L_{ref}$ ) est associé à une pose spécifique  $Pose_i$ .

Telle qu'introduite (Chap 2), elle représente la position et l'orientation du capteur, à l'instant précis où le scan a été enregistré. Grâce aux étapes décrites dans le (Chap 3) sur le géoréférencement des données, les informations qui composent cette pose sont exprimées dans le référentiel  $\mathcal{G}$ , identique à celui utilisé pour géoréférencer la carte sémantique 3D. Cette carte ainsi que les poses étant géoréférencées dans le même système de coordonnées connu  $\mathcal{G}$ , les poses  $Pose_i$  peuvent être directement utilisées pour positionner et orienter le modèle de capteur synthétisé  $M_{L_{synth}}$  dans l'environnement simulé, sans nécessiter de nouvelle transformation (Fig 4.5).

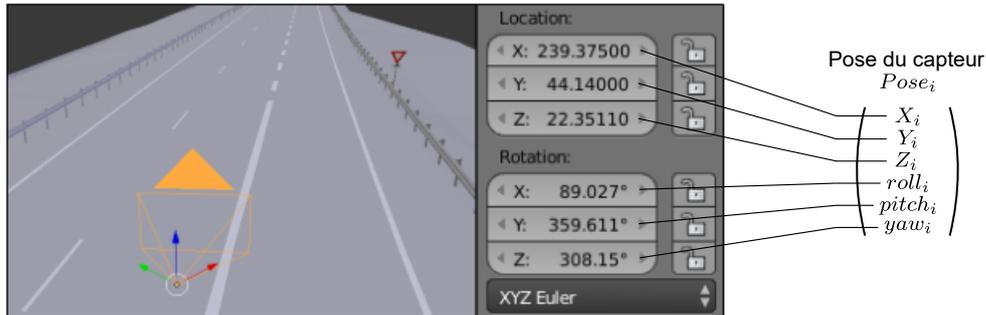


FIGURE 4.5 – Positionnement et orientation du capteur synthétisé dans l'environnement simulé : Pour un scan  $i$  donné, les informations qui composent la pose  $Pose_i$ , auquel il est associé, sont utilisées pour positionner et orienter le capteur synthétisé  $L_{synth}$  au sein de l'environnement simulé. Les paramètres extrinsèques du modèle  $M_{L_{synth}}$  sont redéfinis en fonction des informations de  $Pose_i$ .

Pour chaque scan  $scan_{réel,i}$  enregistré lors de l'acquisition, la pose correspondante  $Pose_i$  est utilisée pour paramétrer le modèle de capteur synthétisé  $M_{L_{synth}}$ . Cela permet de positionner le capteur synthétisé  $L_{synth}$  à l'emplacement précis où se trouvait le capteur de référence  $L_{ref}$  relativement à l'environnement d'essai  $EE$ , lors de l'acquisition. En répétant ce procédé pour chaque scan  $scan_{réel,i}$  qui compose le jeu de données utilisé, il est possible de reproduire un jeu de données analogue, de façon synthétique, composé par les scans  $scan_{synth,i}$ .

---

#### Algorithm 4.1 Algorithme de génération automatisé des données de synthèse

---

##### Initialisation :

$n$  : nombre total de scans LiDAR enregistrés et stockés lors de l'étape "Acquisition de données".

$scan_{réel,i}$  : scan LiDAR d'indice  $i = \{1, \dots, n\}$ , enregistré lors de l'acquisition via le capteur de référence.

$Pose_i$  : pose du capteur associée au scan  $scan_{réel,i}$ .

$scan_{synth,i}$  : scan de synthèse générée par BlenSor et correspondant à  $scan_{réel,i}$

**Entrée :** Nombre de scans  $n$

**for**  $i = 1$  à  $n$  **do**

$Pose_i$  récupère les informations de localisation du capteur ( $X, Y, Z, roll, pitch, yaw$ ) correspondant à  $scan_{réel,i}$ .

$Pose_i$  est utilisée pour paramétrer BlenSor et positionner le modèle de capteur dans l'environnement de simulation.

BlenSor génère le scan de synthèse  $scan_{synth,i}$ .

**end for**

**Sortie :** Scans de synthèse  $scan_{synth,i}$

---

Un dernier avantage qu'offre BlenSor est la possibilité d'exécuter l'outil sans passer par l'interface graphique, via un script Python. Cela permet d'automatiser la génération des scans de synthèse  $scan_{synth,i}$  pour un jeu de données complet, sans nécessiter l'intervention manuelle de l'utilisateur pour modifier les paramètres du capteur synthétisé  $L_{synth}$  à chaque scan généré. Pour un jeu de données issu d'une acquisition donnée, le script mis en place récupère l'ensemble des poses  $Pose_i$  associées aux scans  $scan_{réel,i}$  produits par  $L_{ref}$ . Le script mis en place pour automatiser la génération de données de synthèse est représenté par l'algorithme (Alg 4.1).

### 4.2.3.2 Données de synthèse générées

L'utilisation d'un modèle de capteur  $M_{L_{synth}}$  paramétriquement proche du modèle  $M_{L_{ref}}$  correspondant au capteur de référence utilisé, couplé à un environnement simulé représentatif de l'environnement d'essai réel  $EE$ , permet de générer des données de synthèse proches des données obtenues en conditions réelles, bien que certaines différences peuvent être perçues (Fig 4.6).

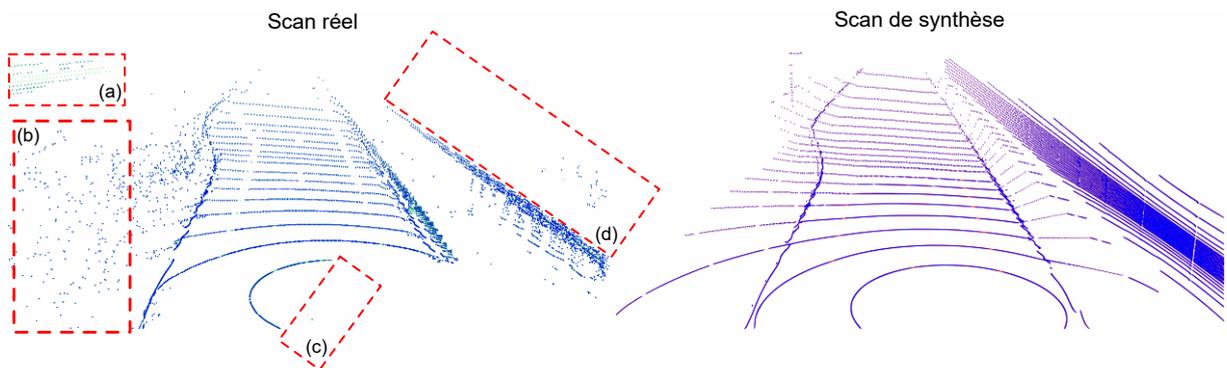


FIGURE 4.6 – Comparaison entre scan réel et scan de synthèse : Pour une même  $Pose_i$  on peut comparer le  $scan_{réel,i}$  associé, ainsi que le  $scan_{synth,i}$  généré avec les informations de cette pose. Certaines différences sont visibles entre les deux scans : (a) Végétation haute (branches et feuilles d'arbres) perçue par  $L_{ref}$ , mais non représentée sur la carte sémantique 3D ; (b) même cas que (a) mais avec des buissons ; (c) La première nappe n'est pas complète sur le scan réel car une partie de ses points intersecte le toit de la plateforme et sont filtrés car inutilisables ; (d) Le mur plus visible sur le scan de synthèse correspond à des parois vitrées. Dans le cadre du scan réel, les mesures passent au travers. Dans le cadre de notre simulation, les facettes composant ces parois ne tiennent pas compte de la transparence, d'où les impacts tels que sur un mur standard.

Les données de synthèse sont annotées automatiquement lors de leur génération, en récupérant les informations sémantiques contenues initialement dans la carte sémantique 3D, et grâce au processus étudié dans le (Chap 3).

Les points  $p_{synth,i}$  qui constituent chaque scan de synthèse  $scan_{synth,i}$  généré, sont issues d'une intersection entre le faisceau simulé par ray tracing et l'une des facettes composant la carte sémantique 3D [100]. Toutes les facettes sont associées à une des classes parmi celles définies (Chap 3). Chaque point  $p_{synth,i}$  créé synthétiquement, et qui représente cette intersection faisceau/facette, est automatiquement annoté. Cela s'opère grâce à la méthode de génération de BlenSor, qui récupère l'information sémantique contenue dans la facette impactée pour compléter la valeur de l'attribut *label* de  $p_{synth,i}$ . Comme on peut le voir avec le scan représenté (Fig 4.7), aucun point ne conserve une valeur *label* par défaut (ne correspondant à aucune classe pré-définie).

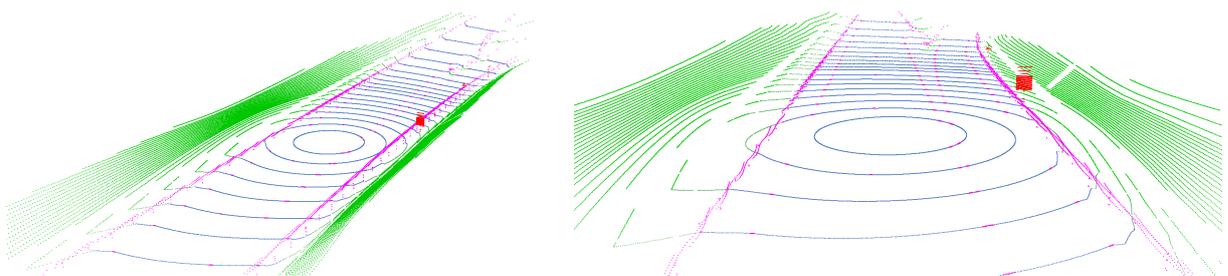


FIGURE 4.7 – Représentation d'un scan de synthèse généré via BlenSor : Scan  $scan_{synth,i}$  labellisé automatiquement lors de la génération et colorisé en fonction des classes sémantiques introduites (Fig 3.14).

Grâce à ces traitements, le premier transfert d'informations sémantiques peut être opéré de la carte sémantique 3D vers les données de synthèse. Ces dernières servent ainsi de support aux informations sémantiques en exploitant un moyen de représentation identique à celui utilisé pour les données acquises en conditions réelles par le capteur  $L_{ref}$  : le nuage de points 3D.

### 4.3 Transfert d'informations sémantiques

La seconde étape du référencement est le « Transfert d'informations sémantiques » (Fig 4.8) et représente concrètement l'adaptation de domaine faite entre les domaines  $MAP$  et  $CRA$ . Nous allons à présent étudier comment exploiter les données à disposition, ainsi que les prédictions d'annotations faites sur les données de synthèse, pour opérer le transfert du domaine  $MAP$  vers  $CRA$  et propager les informations sémantiques.

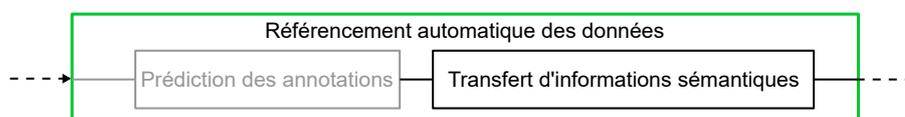


FIGURE 4.8 – Sous-étape « Transfert d'informations sémantiques »

A ce stade des traitements, on dispose de deux types de données distincts, exploitant un moyen de représentation commun, les nuages de points 3D :

1. les scans  $scan_{réel,i}$ , correspondant aux données de perception non labellisées, produites par le capteur de référence au cours de l'acquisition, puis géoréférencés grâce aux données de localisation. Ces données représentent le premier domaine  $CRA$  ;
2. les scans de synthèse  $scan_{synth,i}$ , générés en utilisant conjointement, les mêmes données de localisation, le modèle de capteur et la carte 3D au sein de l'outil de simulation BlenSor. Ces données représentent le second domaine  $MAP$  en servant de support aux informations sémantiques initialement contenues dans le domaine de la carte sémantique 3D.

Pour opérer le transfert d'informations sémantiques, nous allons chercher à associer ces deux types de données. En associant les points qui composent respectivement ces données, nous pouvons récupérer les labels contenus dans  $scan_{synth,i}$  pour les propager sur  $scan_{réel,i}$ . Une méthode d'association appliquée entre les données sert alors de base à ce transfert. Les données bénéficiant d'un géoréférencement dont la précision du positionnement est centimétrique, celle de l'orientation du capteur est de l'ordre de +/- 20 millidegrés sur le cap et +/- 8 millidegrés sur le tangage et le roulis (Annexe A), nous avons cherché à exploiter cet atout et orienter notre étude sur des méthodes d'association géométriques.

#### 4.3.1 Méthodes d'association de données

Dans un premier temps, il est nécessaire de définir une méthode permettant d'opérer ce transfert et propager les informations sémantiques des données de synthèse vers les données produites par le capteur de référence  $L_{ref}$ . Cette méthode s'apparente à des méthodes d'association de données. Dans la littérature scientifique, on retrouve ce type de méthodes pour différentes tâches telles que le suivi de cibles, en associant une donnée de trajectoire mesurée à une prédiction [31].

La segmentation de données exploite également cet aspect : les méthodes d'association s'appuient sur la base de critères pré-définis pour regrouper les données et former les différents segments [105, 106]. Ces différentes applications ont un objectif commun : associer les données provenant d'un jeu de données « source »  $A$ , avec des données d'un jeu de données « cible »  $B$ . Les données sources qui composent  $A$  sont celles qui ont besoin d'être associées et de récupérer des informations contenues dans les données de  $B$ . Le second jeu  $B$  est défini comme « de référence » car il contient les informations auxquelles on souhaite rattacher les données du jeu  $A$ . L'association des données entre  $A$  et  $B$  permet ainsi de propager les informations contenues dans  $B$  sur les données de  $A$  et donc, de réaliser le transfert d'informations souhaité.

Dans notre cadre d'application, le jeu de données source  $A$  correspond aux scans  $scan_{rel,i}$  mesurés par le capteur de référence  $L_{ref}$ . Les données du jeu cible  $B$  correspondent aux scans de synthèse  $scan_{synth,i}$  qui intègrent les informations sémantiques utilisées pour référencer  $A$  (Chap 3).

Une méthode d'association est composée de deux éléments principaux : une stratégie de recherche et un critère d'association. Ces éléments seront développés au cours de cette section.

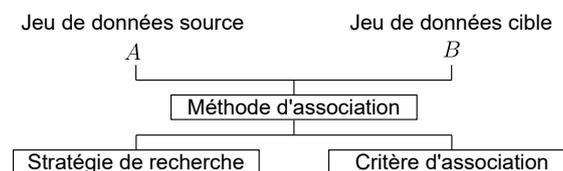


FIGURE 4.9 – Composition d'une méthode d'association.

#### 4.3.1.1 Stratégie de recherche

A partir d'une donnée  $a \in A$ , la stratégie de recherche représente la méthode utilisée pour étudier les données de  $B$  et identifier le ou les meilleurs candidats pour l'association, en fonction du critère d'association utilisé. Cette stratégie de recherche va en partie dépendre du moyen de représentation utilisé pour les données composant  $A$  et  $B$ . En fonction des moyens de représentation discutés au cours du (Chap 3), les nuages de points 3D et les octrees, deux stratégies sont exploitables.

**Stratégie de recherche exhaustive.** Cette première stratégie fonctionne sous l'hypothèse du « un contre tous ». Cela signifie que pour identifier le ou les meilleurs candidats de  $B$  pour l'association avec  $a$ , il faut dans un premier temps parcourir l'ensemble de  $B$  et évaluer la valeur du critère d'association pour chaque données  $b \in B$ . Cette stratégie est principalement destinée aux données non-organisées. Elle intervient dans le cadre du moyen de représentation des données par nuages de points 3D. Elle est plus consommatrice en terme de mémoire que la seconde stratégie discutée et peut difficilement être optimisée, compte tenu du nombre d'opérations à réaliser pour déterminer les meilleurs candidats à l'association.

**Stratégie de recherche organisée.** Cette seconde stratégie de recherche s'appuie sur des données structurées telles qu'en utilisant la représentation par les octrees. Comme déjà introduit et discuté au cours des chapitres (Chap 2 et Chap 3), la structuration des données permet de réaliser des traitements ciblés localement au sein des données. Cela permet d'optimiser les traitements et de pouvoir filtrer les données qui ne sont pas concernées par ces traitements, sans nécessiter le parcours du jeu de données  $B$  dans son intégralité.

### 4.3.1.2 Critère d'association

Différents critères peuvent être exploités dans le cadre des méthodes d'association de données. Ceux-ci peuvent être classés selon trois catégories distinctes représentées sur la (Fig 4.10).

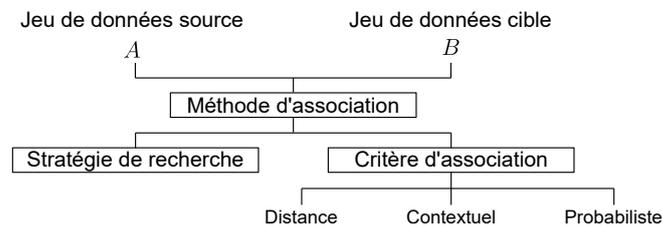


FIGURE 4.10 – Catégories de critères d'association.

**Remarque :** Pour conserver un esprit de synthèse, le choix a été fait de s'orienter sur certains critères spécifiques et les plus évidents vis-à-vis du cadre de ces travaux. Cependant, d'autres critères non présentés pourraient être exploités, tels qu'un critère d'association basé sur les normales à la surface des points [107] ou sur la distance de Manhattan [108]. Ils pourront faire l'objet d'études ciblées, dans des travaux ultérieurs, pour mesurer leurs apports vis-à-vis de la méthode d'association actuellement exploitée.

#### Critères d'association basés sur les distances.

**Correspondance directe :** la première approche nommée ici « correspondance directe » consiste à vérifier si pour une donnée  $a$  issue de  $A$ , on retrouve une donnée  $b$  appartenant au jeu de données  $B$  et dont les informations  $i = \{1 \text{ à } n\}$ , utilisées comme critères d'association, sont strictement identiques [31]. Cela se traduit par l'équation (Eq 4.2). Il s'agit d'un critère d'association « stricte » car il recherche une association parfaite entre les informations utilisées et ne donne pas la possibilité d'utilisation d'un seuil de tolérance.

$$\begin{aligned}
 i_{1_a} &= i_{1_b} \\
 i_{2_a} &= i_{2_b} \\
 &\dots \\
 i_{n_a} &= i_{n_b}
 \end{aligned}
 \tag{4.2}$$

Un exemple parlant est lorsque l'on cherche à associer des points 3D appartenant à deux nuages distincts  $A$  et  $B$ , en se basant sur la position des points. Les informations sur lesquelles se base l'association sont les coordonnées  $(X, Y, Z)$  des points. Pour que la correspondance directe soit respectée et que le point  $a \in A$  soit associé à un point  $b \in B$ , les coordonnées doivent être identiques comme le montre l'équation (Eq 4.3) et comme illustré (Fig 4.11).  $a$  peut être associé à un point  $b$  si :

$$\begin{aligned}
 X_a &= X_b \\
 Y_a &= Y_b \\
 Z_a &= Z_b
 \end{aligned}
 \tag{4.3}$$

**Distance euclidienne :** l'utilisation de la distance euclidienne peut s'exercer pour différentes applications [109]. Celle-ci représente la longueur de la ligne séparant deux points situés dans un espace euclidien. Elle s'appuie sur les coordonnées cartésiennes de ces points et peut-être mesurée dans un nombre de dimension

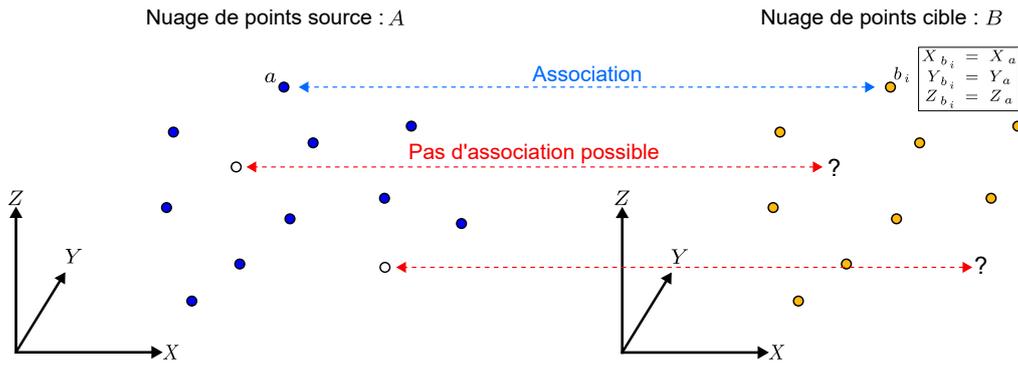


FIGURE 4.11 – Association par correspondance directe entre les données : L'objectif de l'approche est d'associer les points 3D du nuage source à ceux du nuages cible, ce dernier contenant les informations cherchée. Le transfert des informations est opéré quand des points sont positionnés exactement au même emplacement géographique dans chacun des nuages. Dans le cas inverse, le transfert n'a pas lieu et les points du nuage source (en blanc) ne sont pas associés.

infinie. L'équation (Eq 4.4) présente le calcul de distance entre deux points  $p$  et  $q$ , représentés dans un espace de  $n$  dimensions.

$$D_{(p,q)} = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \dots + (p_n - q_n)^2} \quad (4.4)$$

où  $D_{(p,q)}$  est la distance euclidienne séparant les points  $p$  et  $q$ .  $p_i$  et  $q_i$  correspondent respectivement aux coordonnées de  $p$  et  $q$  dans la dimension  $i$ .

Le distance euclidienne peut être utilisée comme critère d'association grâce à la définition d'une valeur maximale  $D_s$  jouant le rôle de seuil de tolérance. Pour reprendre l'exemple introduit pour le critère basé sur la correspondance directe : à partir d'un point  $a$ , les distances euclidiennes séparant  $a$  des points  $b_i$  appartenant au nuage cible  $B$  sont calculées. Les points  $b_i$  séparés de  $a$  par une distance euclidienne inférieure à  $D_s$  peuvent être associés à ce point. Avec des points 3D, l'utilisation de ce procédé revient à définir une sphère centré sur le point  $a$  et de rayon  $D_s$  (Fig 4.12).

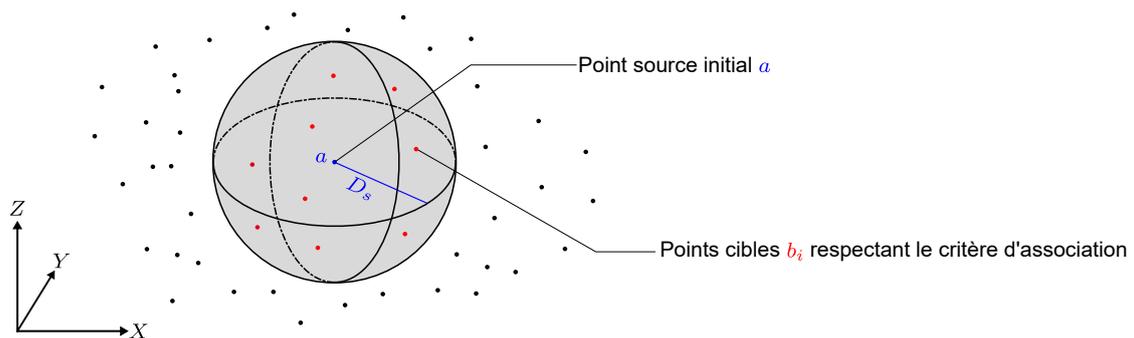


FIGURE 4.12 – Association de données basée sur la distance euclidienne : La sphère représente la zone de tolérance centrée sur le point  $a$  et dont le rayon est défini par  $D_{max}$ . Les points  $b_i$  représentés en rouge sont séparés de  $a$  par une distance euclidienne inférieure à  $D_{max}$  et correspondent aux points pouvant être associés à  $a$ .

L'ensemble des points  $b_i$  positionnés à l'intérieur de cette sphère satisfont la condition imposée par ce critère de distance euclidienne (Eq 4.5).

$$D_s > D_{(a,b_i)} = \sqrt{(X_a - X_{b_i})^2 + (Y_a - Y_{b_i})^2 + (Z_a - Z_{b_i})^2} \quad (4.5)$$

La paramétrisation de cette méthode peut cependant se montrer complexe dans la définition de  $D_s$ . Ce seuil doit être suffisamment grand pour permettre à des données de satisfaire le critère d'association et d'améliorer le taux d'association des données. D'un autre côté, ce seuil ne doit pas être trop important car, en réalisant des associations avec des données éloignées, cela impacterait la pertinence de l'association réalisée. Bien que cette approche offre une plus grande flexibilité par rapport à la correspondance directe, elle ajoute une complexité dans la définition du seuil de tolérance représenté par  $D_s$ . Dans le cadre des nuages de points, la densité de points aux différents endroits de la scène représentée, peut varier de façon importante à l'inverse du seuil défini qui est fixe. Il y a donc un compromis à trouver entre le taux d'association voulu et la tolérance de proximité entre les données à associer.

### Critères d'association contextuels.

**$k$  plus proches voisins :** ce critère d'association possède des points communs avec celui basé sur la distance euclidienne, dans la mesure où il s'appuie également sur des distances séparant une donnée source de  $A$ , des données cible de  $B$  [110]. Ce critère d'association est fixé par rapport à un paramètre  $k$ , fixé arbitrairement par l'utilisateur. Ce paramètre représente le nombre de candidats pris en compte pour l'association. Cela signifie qu'une donnée  $a$  pourra être associée à  $k$  données  $b_{i=\{1 \text{ à } k\}}$ .

Pour définir quelles sont ces données  $b_{i=\{1 \text{ à } k\}}$ , les distances euclidiennes calculées entre la donnée  $a$  et les données  $b_i$  sont classées par ordre croissant. Les données  $b_i$  correspondant aux  $k$  plus courtes distances euclidiennes mesurées, soit les  $k$  premières du classement, représentent les données associables à  $a$ . Une représentation dans le cadre de l'utilisation de ce critère avec des nuages de points 3D est proposée (Fig 4.13) pour différentes valeurs du paramètre  $k$ .

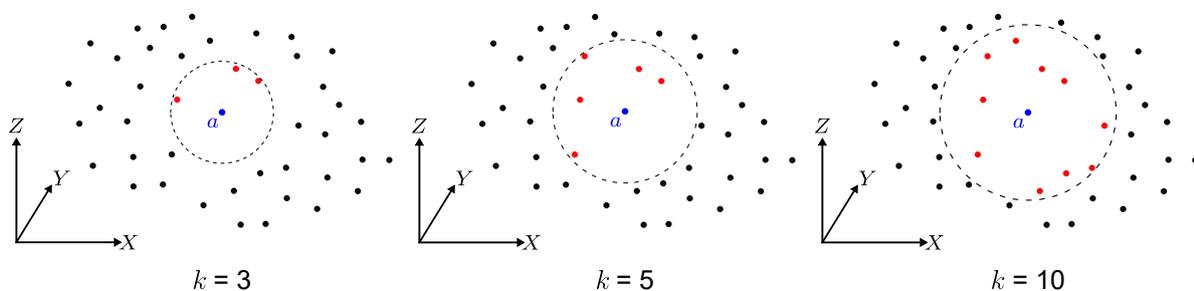


FIGURE 4.13 – Association de données par les  $k$  plus proches voisins : Représentations de la recherche des  $k$  plus proches voisins pour trois valeurs différentes de  $k$ . Les points rouges représentent ces  $k$  voisins  $b_{i=\{1 \text{ à } k\}}$ , appartenant au jeu de données cible  $B$  et pouvant être associés au point source  $a$ .

La plus grande complexité qui apparaît avec l'usage de ce critère d'association réside dans la définition du paramètre  $k$ . En effet, en fonction de la valeur de  $k$  fixée, le voisinage étudié peut être plus ou moins vaste (Fig 4.13). La prise en considération de plusieurs voisins peut être positive s'ils permettent de consolider une hypothèse recherchée avec l'association. Cependant, cela peut également amener des incertitudes lorsque certains des  $k$  voisins pris en compte possèdent des caractéristiques variables, ou lorsqu'ils sont trop éloignés géométriquement de la donnée source initiale.

De plus, comme évoqué dans le cadre de la distance euclidienne, les données éparses sur lesquelles nous nous appuyons peuvent intégrer des variations de densité importantes au sein d'un même nuage, comme l'illustre la (Fig 4.14). Pour une même valeur de  $k$  et pour un même nuage, les données voisines  $b_{i=\{1 \text{ à } k\}}$  considérées en fonction de la donnée initiale  $a$  peuvent être très différentes.

L'objectif recherché à travers l'association peut alors être biaisé par la prise en considération de voisins trop éloignés géométriquement de la donnée initiale et moins pertinents vis-à-vis de celle-ci.

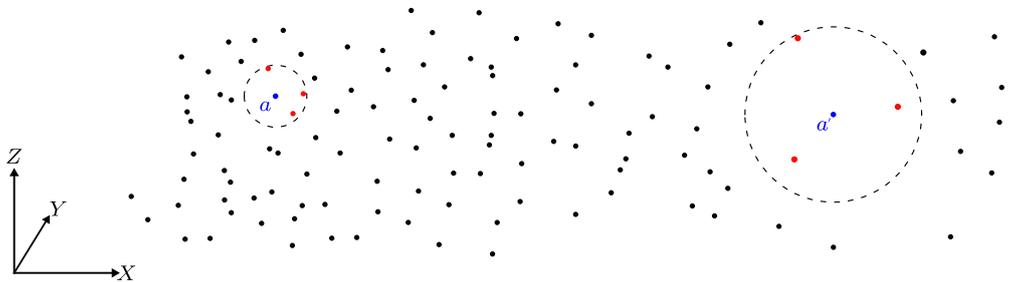


FIGURE 4.14 – Impact de la variation de densité pour l'association basée sur les  $k$  plus proches voisins : Association de données au sein d'un nuage de points 3D. Le point  $a$  se situe dans une zone du nuage où la densité de point est forte et les points  $b_{i=\{1 \text{ à } k\}}$  associés sont géométriquement proches  $a$ . À l'inverse,  $a'$  se situe dans une zone où la densité est plus faible. Les points  $b'_{i=\{1 \text{ à } k\}}$  associés sont plus éloignés de  $a'$  que les points  $b_i$  par rapport à  $a$ . Cela accroît le risque d'associer  $a'$  à des points dont les caractéristiques sont variables et non représentatives de ce point initial.

Pour contrer ce phénomène, une première possibilité est de fixer un seuil de tolérance : une distance euclidienne seuil  $D_s$ , au-delà de laquelle les voisins ne sont plus pris en compte, même si le nombre de données considérées associables est inférieur au  $k$  fixé. Cependant, cela rajoute une étape de contrôle vis-à-vis de ce seuil  $D_s$ . De plus, l'intérêt d'utiliser ce critère plutôt que la distance euclidienne est fortement réduit puisqu'il reviendrait à l'utiliser tout en fixant une limite supplémentaire :  $k$  limitant les données associables.

Lorsque ce critère d'association est utilisé pour un objectif bien précis, tel que l'annotation de données, une seconde possibilité permettant d'orienter le choix de  $k$  passe par la réalisation de tests. Ce critère est utilisé pour opérer des associations sur des données connues *a priori* et dont les annotations sont pré-définies. Après association, il est possible de contrôler que les annotations issues des associations sont cohérentes avec celles pré-définies. Ce test est réalisé à plusieurs reprises, en faisant varier la valeur de  $k$ . La valeur de  $k$  qui minimise le taux d'erreur mesuré vis-à-vis des annotations est conservé. Il faut cependant s'assurer que les jeux de données, autres que celui utilisé lors des tests, soient représentatifs de ce dernier, pour conserver la pertinence de  $k$ . Si trop de différences interviennent entre ces jeux (en termes de densité de points ou de leur répartition par exemple), la valeur optimale définie pour  $k$  peut alors perdre en pertinence.

**Critère basé sur la connexité :** la connexité ou « pixel connectivity » est issue du domaine de la vision par ordinateur, notamment avec le traitement d'image ou la reconnaissance d'image [111]. Initialement utilisée sur des images, cette méthode s'appuie sur la manière dont les pixels d'une image 2D sont liés à leurs voisins [112]. Depuis, ce principe a pu être adapté en 3D, avec l'utilisation des voxels qui composent les structures en octree [113, 114].

Elle s'appuie sur des données représentées de façon organisée et structurée telles que les cellules d'une grille de résolution fixe (par ex : grille 2D composée de pixels pour une image ou, grille 3D qui structure un octree). Lorsque l'on souhaite étudier une cellule<sup>1</sup> d'une grille en fonction de son voisinage, la connexité consiste à définir les «  $n$ -connexes » cellules voisines directes de la cellule étudiée. En fonction du type de voisinage que l'on souhaite prendre en considération pour l'association,  $n$  évolue (Fig 4.15).

1. La cellule correspond à l'élément type composant la grille. Pour une grille 2D d'image, chaque cellule de la grille correspond à un pixel. Pour une grille 3D dans le cadre d'un octree, les cellules correspondent aux voxels de l'octree.

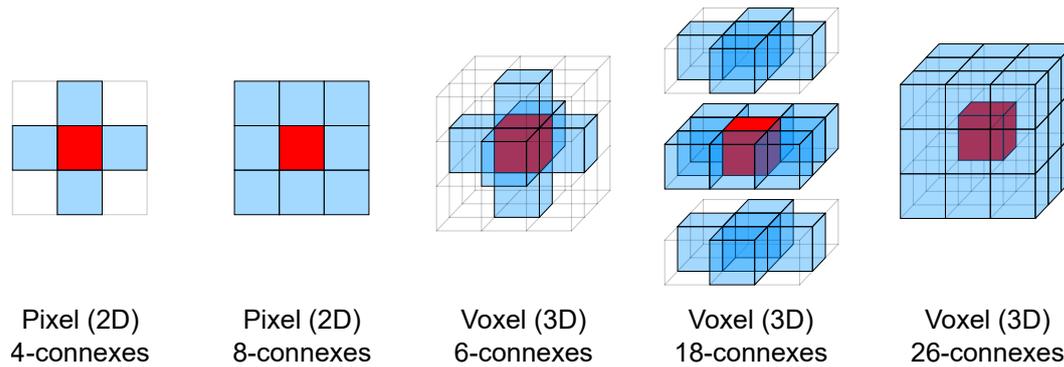


FIGURE 4.15 – Connexité des données : Représentation de la connexité avec les différents voisinages pouvant être pris en compte pour des pixels 2D ou des voxels 3D. La cellule rouge est celle pour laquelle le voisinage est analysé. Les cellules bleues correspondent aux voisins pris en considération lors de l'analyse. (Dans le cas des 18-connexes voisins, les données ont été représentées avec un espacement par souci de lisibilité.)

Initialement, la connexité n'est pas utilisée dans le cadre de l'association de données. Cependant, sous certaines hypothèses, il est possible de l'utiliser comme critère d'association de données entre deux jeux distincts  $A$  et  $B$ . Pour l'employer les données doivent être structurées sur la base d'une grille. Pour l'employer, sur deux ensembles de données distincts, les hypothèses suivantes doivent être respectées :

- Les grilles qui structurent respectivement les données de  $A$  et les données de  $B$  doivent être parfaitement alignées ;
- Ces deux grilles doivent être paramétrées par une résolution strictement identique.

Lorsque ces deux hypothèses sont respectées (Fig 4.16-a,b,c), il est alors possible pour une cellule  $a$  appartenant à  $A$ , de retrouver la cellule analogue  $a'$  au sein de  $B$ . Il est ensuite possible d'étudier le voisinage de  $a'$ .  $n$  représente alors le critère d'association. En fonction de valeur  $n$  fixée, les cellules  $b_{i=\{1 \text{ à } n\}}$  peuvent être associées à la cellule  $a$  initiale (Fig 4.16-d).

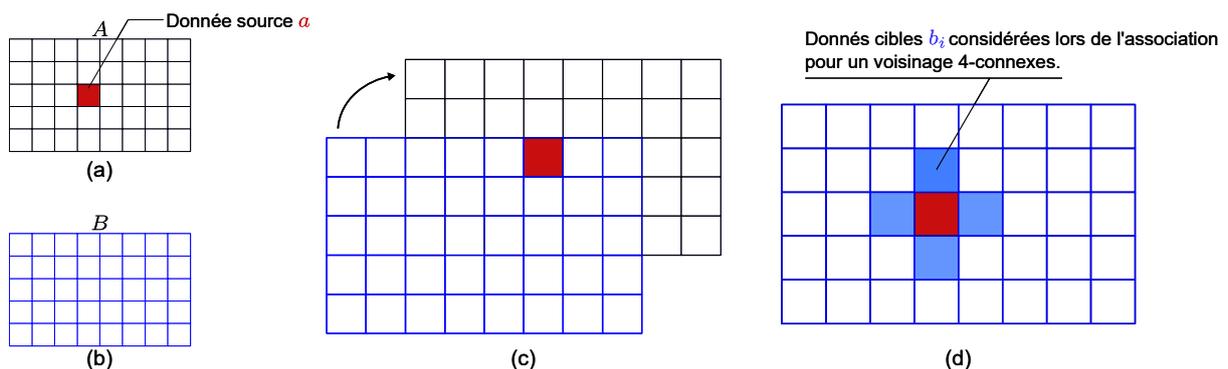


FIGURE 4.16 – Association de données en se basant sur la connexité : Exemple dans le cadre de données 2D, (a) représente la donnée source  $a$  (rouge) ainsi que la grille structurant les données de  $A$ ; (b) représente la grille structurant les données de  $B$ ; (c) illustre le fait que les grilles de  $A$  et  $B$  disposent d'une résolution identique et sont alignées; (d) Les cellules bleues correspondent aux données  $b_{i=\{1 \text{ à } n\}}$  (avec  $n = 4$ ) et pouvant être associées à la donnée  $a$ .

### Critères d'associations probabilistes.

**Distance de Mahalanobis :** introduite dans [115], cette distance permet de déterminer la similarité entre une donnée source et une série de données cibles. Elle se différencie de la distance euclidienne en prenant en considération la variance et la corrélation de la série de données cibles. Elle est donnée par l'équation (Eq 4.6).

$$D_M(a) = \sqrt{(a - m)^T \Sigma^{-1} (a - m)} \quad (4.6)$$

où  $a \in A$  représente la donnée cible et  $m$ , la moyenne des valeurs cibles. Dans le cadre des nuages de points,  $a$  représente le vecteur de coordonnées de ce point :  $a = [x_a, y_a, z_a]$ .  $m$  représente le vecteur des coordonnées  $[x_m, y_m, z_m]$  moyennes, sur l'ensemble des points cibles  $b_i \in B$ .

A la différence de la distance euclidienne, où chaque donnée cible est traitée indépendamment et avec le même poids, la distance de Mahalanobis pondère les données cibles en fonction de leur dispersion, ce qui permet de minimiser l'influence des données ayant une variance plus importante.

**Similarité cosinus :** une méthode destinée à mesurer la similarité entre deux données distinctes peut être utilisée dans le cadre de l'association, il s'agit de la similarité cosinus. Cette métrique est utilisée pour mesurer la similarité entre deux vecteurs de dimension  $n$  en déterminant le cosinus de l'angle situé entre ces deux vecteurs. Cette métrique se calcule grâce au produit scalaire et à la norme des vecteur comme indiqué sur l'équation (Eq 4.7).

$$\cos \theta = \frac{V_A \bullet V_B}{\|V_A\| \|V_B\|} \quad (4.7)$$

où  $\theta$  représente l'angle entre le vecteur  $V_A$  et le vecteur  $V_B$ . L'utilisation du cosinus permet de borner la valeur de la métrique dans l'intervalle  $[-1;1]$ . Plus la valeur est proche de 1 et plus les données sont similaires.

### 4.3.2 Caractéristiques de la méthode choisie

L'objectif de la méthode appliquée dans le cadre de cette thèse est d'opérer des associations entre données produites en conditions réelles (*ε domaine CRA*) et données générées synthétiquement (*ε domaine MAP*), dans l'objectif de transférer les informations sémantiques. Pour mettre en place cette méthode, trois éléments sont définis :

- La stratégie de recherche introduite (Sec 4.3.1), en fonction du moyen de représentation des données utilisé ;
- Le moyen de représentation choisi est adapté pour contenir les informations sémantiques contenues dans les scans de synthèse ;
- La façon dont les critères d'association sont utilisés dans le cadre du « Transfert d'informations sémantiques » entre les données des domaines *MAP* et *CRA*.

#### 4.3.2.1 Structuration des données

La discussion menée lors du (Chap 3) sur les moyens de représentations adaptés aux données LiDAR, ainsi que les avantages recensés avec les méthodes d'association (Sec 4.3.1), nous orientent vers des données

structurées sous forme d'octrees. Comme nous avons pu les introduire au cours des chapitres (Chap 2 et Chap 3), les octrees correspondent à une représentation de l'espace qui est volumétrique et multi-échelles. Les volumes qui composent l'espace et constituent l'octree sont des voxels, représentés numériquement par un nœud. Ce dernier est composé d'un premier attribut *key* qui identifie le voxel et le positionne dans l'arborescence. Un second attribut *value* est utilisé pour représenter la probabilité d'occupation du voxel.

L'utilisation des octrees, et la structuration des nuages de points 3D qu'ils impliquent, leur permettent d'être compatibles avec l'ensemble des critères d'association introduits. A l'inverse, les nuages de points 3D, non-organisés<sup>2</sup>, ne permettant pas d'appliquer certains critères tels que celui basé sur la connexité (Sec 4.3). Ce dernier nécessitant des données structurées sur la base d'une grille.

Enfin, l'utilisation de données structurées apporte d'autres atouts. Le premier est l'optimisation des traitements, grâce à une stratégie de recherche qui permet de cibler et d'étudier localement des données (Chap 2). Cette optimisation est particulièrement efficace lors des traitements tels que celui mis en place ici : l'association des données. Enfin, à partir de la résolution  $r_{max}$  définie, l'utilisation des octrees permet de travailler à différentes échelles de résolution, en fonction de la profondeur  $d$  ciblée (Fig 4.17). Cela ajoute une capacité d'adaptabilité supplémentaire par rapport aux nuages de points 3D.

Les scans présentés jusqu'ici et obtenus par acquisition, ou génération de données de synthèse, sont structurés afin d'utiliser uniquement les octrees dans la suite de cette thèse. On nommera ainsi  $o_{réel,i}$  un octree obtenu à partir d'un scan acquis en conditions réelles ( $scan_{réel,i}$ ). Les octrees créés à partir des scans générés de façon synthétique ( $scan_{synth,i}$ ) seront nommés  $o_{synth,i}$ .

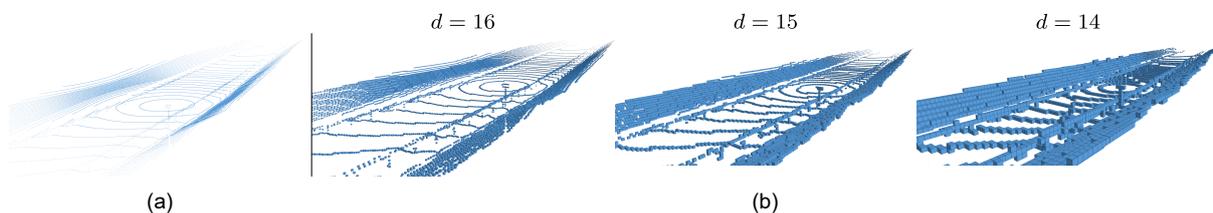


FIGURE 4.17 – Représentation de la structuration des données : (a) Scan  $scan_i$  représenté sous la forme d'un nuage de point 3D, avant structuration des données ; (b) Octree  $o_i$  obtenu à partir de  $scan_i$ , après avoir structuré les données et pour trois niveaux de profondeur différents correspondant respectivement à  $r_{16} = 10cm$ ,  $r_{15} = 20cm$  et  $r_{14} = 40cm$ .

#### 4.3.2.2 Enrichissement sémantique des octrees de synthèse

L'objectif de la méthode d'association est d'opérer le « Transfert d'informations sémantiques », entre les données de synthèse représentant le domaine *MAP*, vers les données issues de mesures en conditions réelles du domaine *CRA*, afin de finaliser leur référencement. Si l'utilisation des structures octrees nous permettent d'appliquer la méthode d'association souhaitée, il est également nécessaire que cette structure, et notamment les voxels qui la composent, puissent servir de support à l'information sémantique. L'atout que peuvent constituer les informations sémantiques dans la représentation des données a permis à différents travaux de la littérature de s'orienter sur cette problématique d'intégration de la sémantique dans les octrees [116, 117, 118].

2. Les nuages de points non-organisés (ou unorganized point cloud) correspondent à des données non structurées. Ces données se représentent par une liste de points suivant l'ordre des mesures réalisées par le capteur. Elles ne s'appuient sur aucune structure géométrique organisée, comme c'est le cas pour les nuages organisés qui se basent sur une structure du type matrice (ou image), et où les données sont divisées en lignes et en colonnes. Les traitements utilisant des opérations de proximité, sont bien moins efficace sur les nuages non-organisés que sur ceux organisés. L'utilisation de nuages organisés permet d'accélérer les calculs et de réduire les coûts de certains algorithmes.

De façon native, la librairie OctoMap [33], utilisée dans nos travaux, ne permet pas de stocker ces informations sémantiques dans les attributs des nœuds associés aux voxels. Pour introduire ces informations dans la structure octree, nous avons réalisé une première modification du code source d'OctoMap. Cette modification nous a permis d'intégrer deux nouveaux attributs dans la composition des nœuds propres à chaque voxel (Fig 4.18). Le premier attribut *label*, représente la valeur correspondante à une des classes sémantiques introduites dans le (Chap 3). Le second attribut *votes* correspond à un critère mis en place pour mesurer la pertinence de l'attribut *label*. Son utilisation est précisée dans la suite de cette section avec la description des deux méthodes introduites.

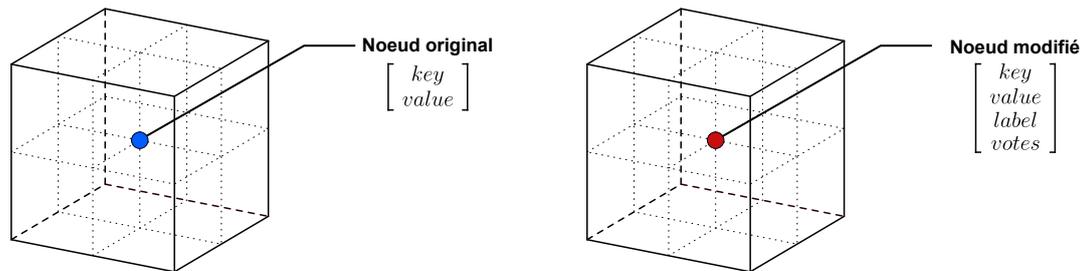


FIGURE 4.18 – Représentation de la structure des voxels modifiée : avec les attributs composant les nœuds originaux à OctoMap (à gauche) et les attributs composant les nœuds après nos modifications (à droite).

Les octrees permettent de décomposer l'espace de façon volumétrique grâce aux voxels. Dans le volume qu'un voxel représente, plusieurs points correspondant à différentes classes sémantiques peuvent être contenus (Fig 4.19).

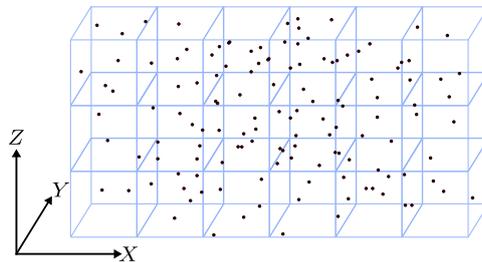


FIGURE 4.19 – Répartition des points en fonction des voxels : Représentation d'un nuage de points lors de sa structuration en octree. Les points qui composent ce nuage sont répartis dans les différents voxels (représentés en bleu).

Pour gérer l'attribution des classes sémantiques, à travers l'attribut *label* propre à chaque voxel, tout en tenant compte que plusieurs points associés à différentes classes peuvent cohabiter au sein d'un même voxel, deux traitements ont été mis en place :

- Le premier traitement permet d'associer une classe sémantique aux voxels les plus proches des données ponctuelles initiales (les points composant les scans). Ils s'agit des voxels situés au niveau le plus bas de l'arborescence et correspondant à la plus petite taille de voxel prise en compte, définie par la résolution  $r_{max}$  fixée (Chap 3) ;
- La seconde méthode est mise en place pour obtenir une transmission cohérente des informations sémantiques, à travers les différents niveaux de profondeur de l'arborescence octree. Cela permet de maintenir une cohérence sémantique entre les voxels parents et enfants, et dans l'octree de façon générale.

**Annotation des voxels au plus bas de l'arborescence.** Le premier algorithme, mis en place pour gérer l'attribution des classes sémantiques au niveau le plus bas de l'arborescence, est présenté (Alg 4.2). Il opère lors de la structuration des données vers un octree et s'appuie directement sur les informations sémantiques associées aux points traités. Pour un voxel  $V$  donné, cet algorithme se base sur un vote majoritaire pour définir sa classe sémantique finale. Chaque point situé dans le volume de l'espace représenté par  $V$  est annoté par un label, et représente une voix pour la classe sémantique correspondant à ce label.

Les nuages de points étant non-organisés, l'ordre des points qui les composent ne correspond pas à leur répartition géométrique dans l'espace. Lors de l'étape de structuration vers un octree, il est difficile de prédéfinir le nombre de points situés dans le volume représenté par un voxel  $V$  donné, ou le nombre de voix associées à une classe pour la définir comme majoritaire.

L'algorithme (Alg 4.2) mis en place permet alors de mettre à jour dynamiquement les attributs *label* et *votes* associés à  $V$ , en fonction de l'arrivée de nouveaux points contenus dans le volume que ce voxel  $V$  représente. Pour pallier cette complexité, des compteurs de votes associés à chaque classe sémantique prise en compte sont utilisés. Ils permettent de conserver, durant la période de structuration des données, les voix associées à chaque classe sémantique et de mettre à jour les attributs *label* et *votes*, à chaque fois qu'un nouveau point est assigné à  $V$ .

---

**Algorithm 4.2** Algorithme de mise à jour du label sémantique pour un voxel d'octree

---

**Initialisation :**

$V$  : voxel de l'octree

$p$  : nouveau point traité, situé dans le volume représenté par  $V$

$l_p$  : label sémantique associé au point  $p$

*label* : attribut de label sémantique du voxel  $V$

*votes* : attribut de  $V$  représentant le nombre de votes reçus pour *label*

*liste* : Tableau 2D dont la première colonne représente chaque label rencontré dans  $V$  et dont la seconde colonne représente le nombre de votes associés à chaque label

$t_{liste}$  : Nombre de lignes composant *liste*

**Entrée :** Point  $p$  contenu dans un voxel  $V$  de l'octree

**Sortie :** Valeurs des attributs *label* et *votes* du voxel  $V$  mises à jour

**if**  $V =$  nouveau voxel de l'octree **then**

    Instanciation de  $V$  et initialisation des valeurs de ses attributs :

$l_p$  est ajouté à *liste* et reçoit un vote.

*label* =  $l_p$

*votes* = 1

**else**

$V$  déjà existant dans l'octree.

**if**  $l_p$  correspond à un label déjà recensé dans *liste* **then**

        Ce label reçoit un vote de plus dans *liste*.

**else**

$l_p$  n'est pas recensé dans *liste*

$l_p$  est ajouté à *liste* et reçoit un vote.

**end if**

**for**  $i = 1$  à  $t_{liste}$  **do**

        Mise à jour des valeurs attributaires de  $V$  :

        Recherche du nombre maximal de votes  $v_{max}$  trouvé dans *liste*

        Récupération du label  $l_{max}$  correspondant à  $v_{max}$ .

**end for**

*label* =  $l_{max}$

*votes* =  $v_{max}$

**end if**

---

**Transmissions des labels dans les différents niveaux de profondeur de l'octree.** Le premier algorithme (Alg 4.2) est appliqué lors de la production de l'octree, sur les voxels situés au niveau le plus bas de l'arborescence et aux plus proche des données ponctuelles initialement utilisées avec les nuages de points 3D. Pour labelliser les voxels parents et conserver la cohérence sémantique aux différents niveaux de profondeur de l'arborescence, un second algorithme doit être mis en place.

On l'a vu lors de la présentation des structures en octree (Chap 3), un voxel parent est composé de huit voxels enfants. Le voxel parent  $V$  a pour valeur d'occupation  $value$ , la probabilité d'occupation maximale  $\mathcal{P}_{max}$  retrouvée parmi ses voxels enfants  $v_{i=\{1 \text{ à } 8\}}$ . Il peut être composé de voxels enfants occupés ou libres. Cependant, en se basant sur  $\mathcal{P}_{max}$ ,  $V$  est désigné comme occupé à partir du moment où l'un de ses voxels enfants  $v_i$  est occupé.

Pour avoir une transmission plus cohérente des informations sémantiques, l'algorithme (Alg 4.3) a été développé et introduit au code source d'OctoMap. Pour un voxel parent donné  $V$ , cet algorithme n'étudie que les voxels enfants  $v_i$  occupés et associés à une classe sémantique. L'ensemble des valeurs  $label$  rencontrées parmi les voxels  $v_i$  sont prises en compte, ainsi que les valeurs  $votes$ , associées à chaque valeur  $label$ . Ces valeurs sont ensuite étudiées dans leur globalité afin d'en retirer la valeur  $label$  majoritaire et de l'associer à  $V$ , comme le montre cet algorithme (Alg 4.3).

Grâce aux modifications de la librairie OctoMap, et à la mise en place de ces deux algorithmes, nous avons pu introduire l'information sémantique au sein des structures octree (Fig 4.20-a). Lors de l'étape de structuration permettant de passer d'un scan à un octree, ces traitements permettent de prendre en considération les labels associés aux différents points représentés par chaque voxel afin d'annoter ce dernier de façon cohérente. Cette cohérence est également maintenue aux différents niveaux de profondeur de l'octree grâce à la seconde méthode mise en place.

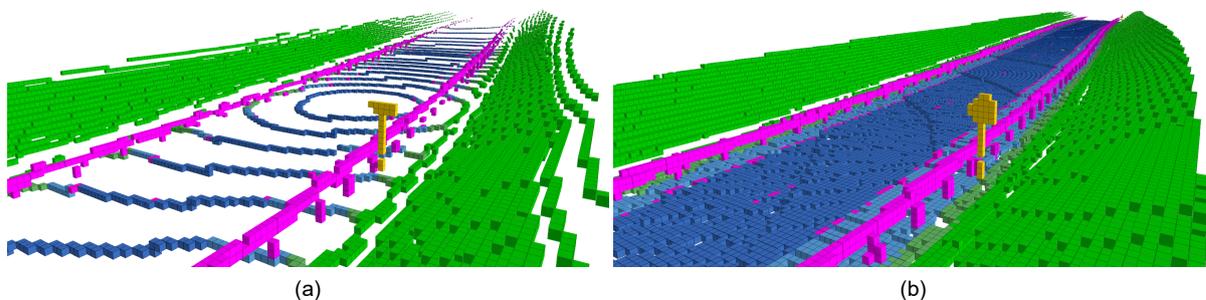


FIGURE 4.20 – Représentation des octrees annotés sémantiquement : (a) Octree  $o_{synth,i}$  correspondant à un scan  $scan_{synth,i}$  et dont les voxels sont annotés et colorisés en fonction de la classe sémantique ; (b) Extrait de la carte octree  $O_{synth}$  produite à partir des octrees  $o_{synth,i}$ .

Enfin, à partir de l'ensemble des octrees  $o_{synth,i}$ , il est possible de construire une carte octree  $O_{synth}$ , représentée (Fig 4.20-b) et qui couvre la totalité de trajectoire associée au jeu de données traité. Cette carte octree est créée en intégrant temporellement les données par la méthode décrite dans [33] et initialement présente dans la librairie OctoMap.

**Remarque :** Lorsqu'un octree  $o$  est créé à partir d'un nuage de points non annoté, mais qui est susceptible d'être annoté lors de traitements ultérieurs, les voxels qui composent  $o$  sont associés à une valeur  $label$  par défaut, ne correspondant à aucune classe définie (Chap 3). Une valeur  $votes$ , neutre ( $= 0$ ), est également associée à cette valeur  $label$ . Cette stratégie est exploitée avec les octrees  $o_{réel,i}$  produits à partir des scans  $scan_{réel,i}$  non annotés.

---

**Algorithm 4.3** Algorithme de transmission des labels sémantiques dans les différents niveaux de profondeur de l'arborescence octree.

---

**Initialisation :**

$tab$  : liste de deux colonnes, une pour les labels rencontrés, la seconde pour le nombre de votes correspondant

$size_{tab}$  : nombre de lignes de  $tab$

$correspondance$  : test booléen indiquant si la valeur cherchée est retrouvée dans  $tab$

$v_i$  : Voxels enfants du voxel  $V$

**Entrée :** Voxel  $V$

**Sortie :** Voxel parent labellisé en fonction de ses voxels enfants  $v_i$

**if**  $V$  est un voxel parent **then**

$V$  contient 8 voxels enfants  $v_i$ .

**for**  $i = 1$  à 8 **do**

**if**  $v_i =$  occupé **then**

$label_{v_i}$  = label stocké dans le voxel  $v_i$

$votes_{v_i}$  = nombre de votes associés au label de  $v_i$

**if**  $i = 1$  **then**

Premier voxel étudié et initialisation du tableau de valeurs

$tab$  stocke le label  $label_{v_i}$  et le nombre de votes correspondant  $votes_{v_i}$

**else**

bool  $correspondance =$  faux

**for**  $j = 1$  à  $size_{tab}$  **do**

**if**  $label_{v_i}$  retrouvé au sein de  $tab$  **then**

$correspondance =$  vrai

Le nombre de votes  $votes_{v_i}$  est cumulé au nombre de votes déjà associés à  $label_{v_i}$ .

**end if**

**end for**

**if**  $correspondance =$  faux **then**

Le label du voxel enfant  $label_{v_i}$  n'est pas encore renseigné dans  $tab$ .

$tab$  stocke le nouveau label  $label_{v_i}$  et le nombre de votes correspondant  $votes_{v_i}$ .

**end if**

**end if**

**end if**

**end for**

**for**  $k = 1$  à  $size_{tab}$  **do**

Recherche du nombre maximal parmi les votes cumulés pour chaque label.

Récupération du label  $L_{max}$  correspondant au nombre maximal de votes trouvé.

**end for**

Le voxel parent  $V$  est annoté avec  $L_{max}$ .

**end if**

---

### 4.3.2.3 Méthode d'association pour les octrees

Enfin, le « Transfert d'informations sémantiques » peut être opéré depuis les octrees  $o_{synth,i}$  représentant le domaine  $MAP$ , vers les octrees  $o_{réel,i}$  représentant le domaine  $CRA$ .

Au cours de leur production, les octrees  $o_{réel,i}$  et  $o_{synth,i}$  exploitent la même pose de capteur  $Pose_i$ . Cela permet deux choses :

1. Ces octrees conservent le géoréférencement des scans à partir desquels ils ont été produits. Par conséquent, les coordonnées des voxels qui composent ces octrees sont géoréférencés dans le même système de référence  $\mathcal{G}$ , défini au cours des chapitres précédents (Chap 2 et Chap 3) ;
2. Ces octrees sont basés sur des grilles 3D initialisées en fonction des coordonnées composant la pose  $Pose_i$  et dont l'origine et l'orientation sont identiques.

Si ces octrees sont paramétrés avec une résolution  $r_{max}$  identique, alors les grilles sur laquelle ils se basent sont parfaitement alignées. Cet ensemble de conditions permet par la suite d'appliquer la méthode d'association choisie entre les deux octrees.

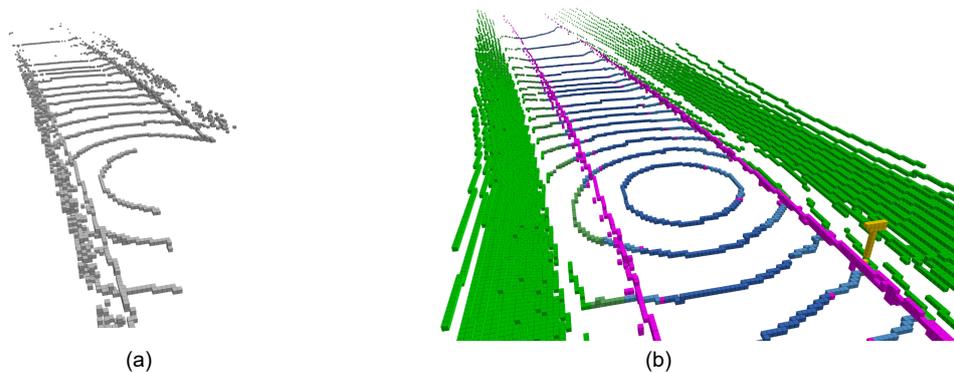


FIGURE 4.21 – Représentation des données à associer : (a) Octree  $o_{réel,i}$  généré à partir du scan  $scan_{réel,i}$  et neutre sémantiquement ; (b) Octree  $o_{synth,i}$  généré à partir du scan de synthèse  $scan_{synth,i}$  et annoté sémantiquement grâce aux informations sémantiques initialement contenues dans ces scans.

**Méthodologie de recherche pour l'association :** Pour opérer le transfert des informations sémantiques entre les voxels appartenant à ces deux types d'octrees, nous nous sommes orientés sur des critères d'association géométrique. La méthode d'association se base tout d'abord sur le critère de correspondance directe. Pour un voxel  $V_{o_{réel}}$  donné, appartenant à  $o_{réel,i}$ , on recherche le voxel analogue  $V'_{o_{synth}}$  au sein de  $o_{synth}$ . Deux possibilités peuvent apparaître :

- $V'_{o_{synth}}$  est occupé et est associé à une classe sémantique. Dans ce cas, sa valeur *label* est récupérée pour être transmise à  $V_{o_{réel}}$  et l'annoter ;
- $V'_{o_{synth}}$  n'est pas occupé. Dans ce cas, on fait appel à la connexité pour élargir la recherche en étudiant son voisinage.

On élargit le champ de recherche progressivement, en faisant évoluer la valeur  $n$  des  $n$ -connexes voisins pris en compte, jusqu'à retrouver des voxels occupés et en suivant les trois types de voisinage introduits (Sec 4.3.1). On commence par étudier le voisinage le plus réduit, si aucun des voisins n'est occupé, alors ce voisinage est élargi. Cette méthode est représentée (Fig 4.22).

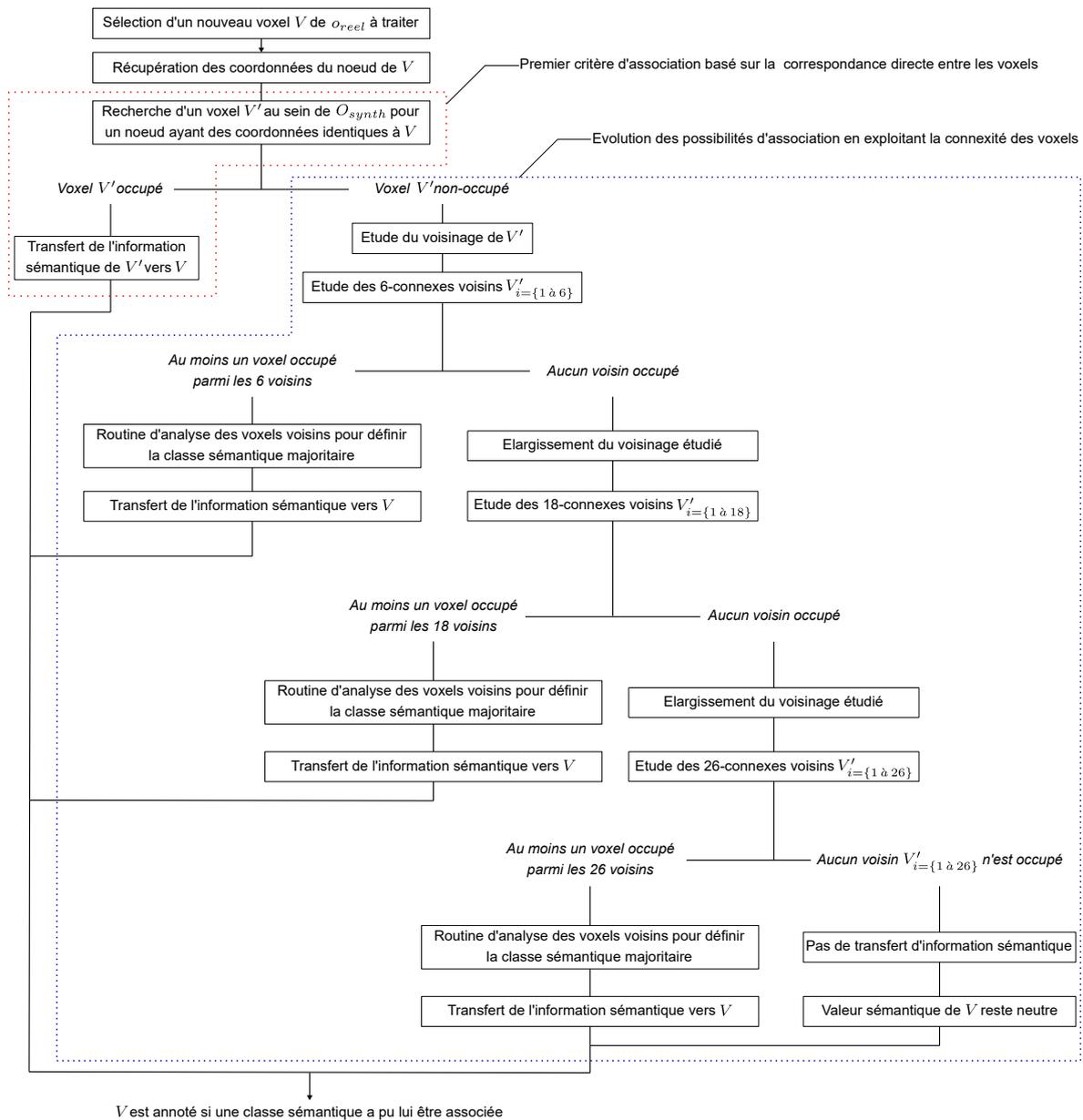


FIGURE 4.22 – Application de la méthode d'association définie : Étapes du transfert de l'information sémantique vers les voxels composant  $O_{réel}$ .

Elle est appliquée pour chacun des voxels occupés appartenant à  $o_{réel,i}$ . Si malgré l'étude des trois types de voisinage, aucun voxel  $V_{o_{synth}}$  n'est occupé,  $V_{o_{réel}}$  conserve sa valeur *label* initiale qui ne correspond à aucune des classes sémantiques prises en compte (Chap 3).

**Routine d'analyse lors de la prise en compte de plusieurs voisins.** Lorsque la connexité est appliquée, un voisinage allant de 6 à 26 voisins peut être étudié. Il est alors possible d'avoir plusieurs voxels voisins  $V_{o_{synth}}$  occupés et associés à différentes classes sémantiques. Pour départager ces voisins et récupérer la valeur *label* la plus pertinente, nous avons mis en place un algorithme (Alg 4.4). De façon cohérente avec les algorithmes mis en place jusqu'à maintenant, cet algorithme étudie par système de vote majoritaire la classe sémantique la plus représentée à travers l'attribut *label*.

---

**Algorithm 4.4** Transfert d'informations sémantiques lors d'association basée sur la connexité.

---

**Initialisation :**

$n$  : type de voisinage étudié avec les  $n$ -connexes voisins.

$V_{o_{synth},i}$  : voxel voisin  $i$  étudié.

$tab$  : tableau de deux colonnes, une pour les labels rencontrés, la seconde pour le nombre de votes correspondant.

$candidat$  : booléen permettant de définir si au moins un voisin étudié est occupé.

**Entrée :** Voxel  $V_{o_{réel}}$

**Sortie :** *label*

$candidat = faux$

**for**  $i=1$  à  $n$  **do**

**if**  $V_{o_{synth},i}$  = occupé **then**

$candidat = vrai$

$label_i$  = label associé au voisin

**if**  $label_i$  existe dans  $tab$  **then**

      1 vote est ajouté pour le  $label_i$  dans  $tab$

$votes_i += 1$

**else**

$label_i$  est ajouté dans  $tab$  et 1 vote lui est admis.

**end if**

**end if**

**end for**

**if**  $candidat = vrai$  **then**

  Au minimum un voxel voisin est occupé.

  On recherche le nombre de votes  $votes_{max}$  le plus élevé parmi les valeurs de  $tab$ .

  Le label  $L_{max}$  correspondant à  $votes_{max}$  est ensuite transféré au voxel traité.

$V_{o_{réel}}.label = L_{max}$ .

**else**

$candidat$  est resté sur sa valeur par défaut : *faux*

  Aucun voisin étudié n'est occupé.

$V_{o_{réel}}.label$  garde sa valeur initiale neutre : 0

**end if**

---

La classe sémantique la plus représentée parmi le voisinage étudié est récupérée pour être associée au voxel  $V_{o_{réel}}$  traité. Par cette stratégie, les annotations prédites à partir du domaine *MAP* sont transférées sur les données du domaine *CRA*.

Ce transfert d'informations sémantiques, représenté (Fig 4.23), permet d'annoter les données collectées en conditions réelles de façon optimisée et sans devoir faire appel à des méthodes d'association complexes, pouvant nécessiter une algorithmie poussée (par exemple : réseau de neurones pour classifier les données).

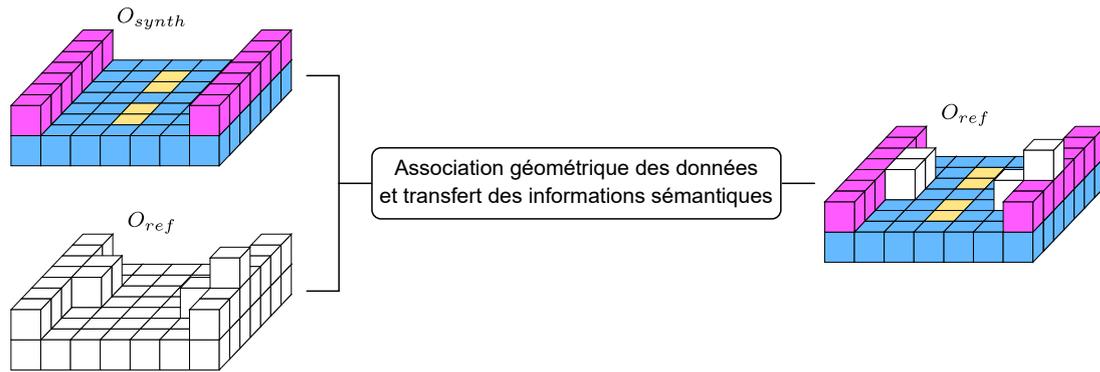


FIGURE 4.23 – Représentation du transfert d'informations sémantiques : Référencement opéré entre les octrees  $O_{ref}$  et  $O_{synth}$  pour finaliser l'adaptation de domaine et permettre le transfert des informations sémantiques. Les voxels pour lesquels le transfert n'a pas pu être opéré sont représentés en blanc.

Les octrees  $O_{réel,i}$  sont automatiquement annotés grâce à ces associations (Fig 4.24-a). Enfin, comme pour les octrees de synthèses, les octrees  $O_{réel,i}$  peuvent être cumulés pour former une carte octree, nommée  $O_{ref}$  (Fig 4.24-b), la plus complète possible. Les voxels  $V_{O_{ref}}$  qui composent cette carte octree  $O_{ref}$  conservent les informations sémantiques auxquelles ils sont associés et pourront servir de référence au cours de la dernière étape : l'« Évaluation de la solution de perception testée ».

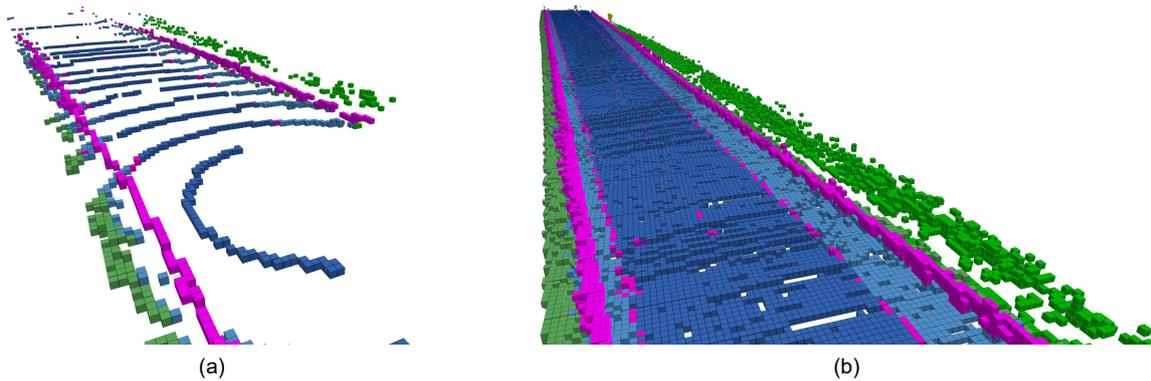


FIGURE 4.24 – Données réelles référencées automatiquement : Représentation des données issues du capteur de référence  $L_{ref}$  puis annotées automatiquement grâce à notre approche avec (a) Représentation d'un octree  $O_{réel,i}$  annoté ; (b) Extrait de la carte octree  $O_{ref}$  servant de référence au cours des évaluations de solutions de perception et basée sur l'intégration temporelle des octrees  $O_{réel,i}$ .

## 4.4 Validation de la méthodologie de référencement

### 4.4.1 Analyse de la méthode proposée

Les premières sections de ce chapitre (Sec 4.2 et 4.3) ont permis de décrire la méthode utilisée pour référencer les données de façon automatique. Elle se compose de deux étapes majeures. La première est la « Prédiction des annotations », centrée sur la génération de donnée de synthèse, par la simulation et en utilisant la carte sémantique 3D de l'environnement d'essai réel  $EE$  comme environnement simulé. Cette première étape permet d'extraire les informations sémantiques du premier domaine  $MAP$  et de les associer à des données de synthèse dont le moyen de représentation initial est identique à celui utilisé pour les données acquises par le capteur de référence  $L_{ref}$  : des nuages de points 3D. La seconde étape majeure représente l'adaptation de domaine avec le « Transfert d'informations sémantiques » depuis les données de synthèses associées au domaine  $MAP$ , vers les données acquises en conditions réelles (domaine  $CRA$ ).

Lors de la mise en place de cette seconde étape. Certains choix méthodologiques ont été opérés. Le premier concerne la structuration des données par l'utilisation des octrees et notamment de la librairie OctoMap. Ce moyen de représentation apporte plusieurs avantages tels que l'optimisation des traitements ou la gestion multi-échelles des données. De plus, en se basant sur une grille 3D probabiliste, ce moyen de représentation est adapté à la multi-modalité et nous permet de définir un moyen de représentation commun à toutes les données exploitées : de référence, de synthèse ou issues des solutions de perception testées. La comparaison des données acquises par  $L_{ref}$  et des données de synthèse permet d'observer certaines différences lorsqu'elles sont représentées sous la forme de nuages de points 3D. Elles sont dues à l'idéalisation des données de synthèse, ainsi qu'au bruit de mesure du aux conditions réelles et difficilement reproductible par simulation. Le découpage volumétrique de l'espace obtenu avec les octrees permet d'amortir ces variations entre les données en ne les représentant pas de façon ponctuelle. Enfin, les derniers choix méthodologiques réalisés portent sur les critères d'association sur lesquels les données de synthèse ont pu être associées aux données « réelles » et sur la base desquels le transfert des informations sémantiques a pu être opéré.

Différents points clés intervenant au cours de notre approche peuvent être analysés et évalués dans la suite ce chapitre. Les tests expérimentaux présentés au cours de cette section ont plusieurs objectifs. Le premier est de contrôler la bonne mise en place de l'approche, notamment du point de vue des modèles de capteur utilisés. Les tests réalisés permettent dans un second temps de cibler le fonctionnement de la méthode de référencement. Ils permettent de s'intéresser aux capacités de cette méthode ainsi qu'à sa robustesse.

#### 4.4.2 Analyse du modèle paramétrique du capteur de référence

Dans l'objectif d'utiliser un modèle de capteur synthétisé  $L_{model}$  qui se rapproche du capteur  $L_{ref}$ , utilisé lors des acquisitions, nous avons mis en place des premiers tests permettant d'étudier le modèle paramétrique du capteur  $L_{ref}$ , sur la base de données acquises en conditions réelles.

Avant la réalisation de ces tests, nous disposions déjà de deux modèles paramétriques correspondant au capteur LiDAR utilisé  $L_{ref}$ . Ils correspondent aux modèles paramétriques sphériques  $M_{L_{ref}}$  sur la base duquel les mesures sont réalisées initialement (Chap 2). Le premier modèle est celui fourni directement par le constructeur, nommé  $M_{L_{const}}$ . Le second,  $M_{L_{synth}}$ , correspond au modèle paramétrique inclus dans l'outil BlenSor et qui correspond au capteur  $L_{ref}$ .

En s'appuyant sur des données réelles acquises avec  $L_{ref}$  en conditions réelles, nous pouvons définir un nouveau modèle  $M_{L_{estim}}$ . Les paramètres qui composent ce modèle sont estimés en se basant directement sur les scans produits, sans tenir compte des modèles à disposition ( $M_{L_{const}}$  et  $M_{L_{synth}}$ ). La comparaison des paramètres propres à chaque modèle permet de vérifier la cohérence des trois modèles et de s'assurer que le capteur synthétisé  $L_{model}$  s'appuie sur un modèle paramétrique représentatif de celui du capteur réel  $L_{ref}$ . Les estimations faites sur les données de mesure réelles ont été réalisées pour trois vitesses de roulage de la plateforme : à l'arrêt ( $v = 0km/h$ ), à vitesse faible ( $v = 40km/h$ ) et à moyenne vitesse ( $v = 70km/h$ ). Cela permet également d'étudier par ces mêmes tests l'impact de la dynamique du véhicule sur les mesures.

##### 4.4.2.1 Méthode d'estimation des paramètres

Géométriquement, des points mesurés par un LiDAR rotatif peuvent être vus comme appartenant à un cône dont l'apex correspond au point d'émission du faisceau : l'émetteur. Chaque point mesuré par le faisceau va ainsi se trouver sur la surface du cône (Fig 4.25-a).

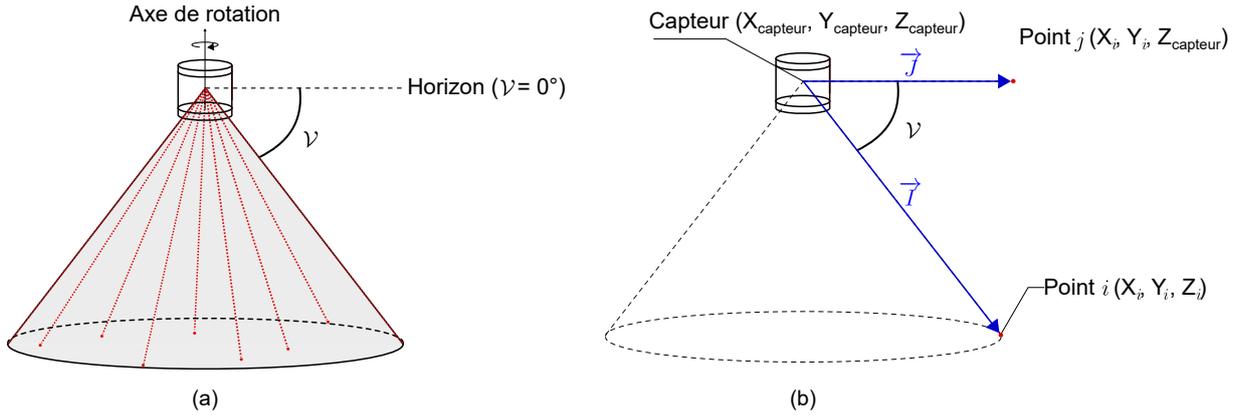


FIGURE 4.25 – Analogie entre un cône et la forme produite par un faisceau laser lors d'une rotation : (a) Représentation de la forme géométrique conique formée par les mesures d'un LiDAR 3D, pour un faisceau donné, et en fonction de son angle d'élévation ; (b) Représentation de l'angle d'élévation  $V$  estimé pour une nappe donnée du LiDAR, en fonction de la position du capteur d'un point mesuré appartenant à cette nappe.

**Estimation des angles d'élévation :** Pour une nappe *ring* donnée, l'angle d'élévation du faisceau qui la détermine peut être assimilé à l'angle séparant deux vecteurs. A partir d'un point  $i$  de la nappe *ring*, le premier vecteur  $\vec{I}$  est défini entre les coordonnées du centre du capteur (point d'émission du faisceau) et celles du point  $i$  (Eq 4.8).

$$\vec{I} = \begin{bmatrix} X_i - X_{\text{capteur}} \\ Y_i - Y_{\text{capteur}} \\ Z_i - Z_{\text{capteur}} \end{bmatrix} = \begin{bmatrix} dX_{i-\text{capteur}} \\ dY_{i-\text{capteur}} \\ dZ_{i-\text{capteur}} \end{bmatrix} \quad (4.8)$$

Le second vecteur  $\vec{J}$  est défini de la même manière que  $\vec{I}$  : entre le point d'émission et un point  $j$ . Ce point  $j$  a des coordonnées identiques à  $i$  dans les deux premières dimensions ( $X_j = X_i, Y_j = Y_i$ ). Pour la troisième dimension, la coordonnée du point  $Z_j$  correspond à la valeur d'altitude du capteur. Cela permet d'avoir une valeur  $dZ$  nulle pour le vecteur  $\vec{J}$ , soit un vecteur horizontal aligné par rapport au capteur (Fig 4.25-b). En étant sur l'horizon, il représente la valeur nulle de l'angle d'élévation. En calculant l'angle séparant les vecteurs  $\vec{I}$  et  $\vec{J}$ , on en déduit l'angle d'élévation du faisceau ayant permis de mesurer  $i$ . L'angle  $V$  entre ces deux vecteurs est obtenu par l'équation (Eq 4.9).

$$\cos V = \frac{\vec{I} \cdot \vec{J}}{\|\vec{I}\| \|\vec{J}\|} \quad (4.9)$$

soit

$$V = \arccos \left( \frac{\vec{I} \cdot \vec{J}}{\|\vec{I}\| \|\vec{J}\|} \right) \quad (4.10)$$

L'opération peut être répétée pour chaque point d'une même nappe afin de calculer une valeur moyenne de l'angle d'élévation  $V$ , ainsi que de mesurer l'écart-type  $\sigma_V$  des angles d'élévation sur l'ensemble de la nappe. Cette opération peut être menée sur chacune des nappes que propose le capteur.

**Estimation de la résolution azimutale :** Concernant les angles azimutaux, aucun angle n'est prédéfini dans les modèles  $M_{L_{const}}$  et  $M_{L_{synth}}$ . Concernant  $M_{L_{const}}$  la seule information à disposition est la résolution azimutale ( $r_{Az}$ ) qui varie en fonction des fréquences d'acquisitions exploitables (Tab 4.1). Le simulateur BlenSor propose à l'utilisateur de fixer la résolution azimutale pour le modèle  $M_{L_{synth}}$ , aucune valeur spécifique n'est définie.

| Fréquence $F$ (Hz)      | 5     | 10    | 15    | 20    |
|-------------------------|-------|-------|-------|-------|
| Résolution $r_{Az}$ (°) | 0.10° | 0.20° | 0.30° | 0.40° |

TABLE 4.1 – Résolution azimutale du Velodyne VLP-32C : Résolution annoncée en fonction de la fréquence de rotation  $F$  exploitée.

Comme pour les angles d'élévations,  $r_{Az}$  est ici calculée grâce aux angles séparant deux vecteurs  $\vec{I}$  et  $\vec{J}$ . Ces deux vecteurs sont respectivement définis entre les coordonnées du point d'émission et les coordonnées de  $i$  et  $j$ , deux points consécutifs d'une même nappe au sein du scan traité. Cette mesure est représentée dans la (Fig 4.26).

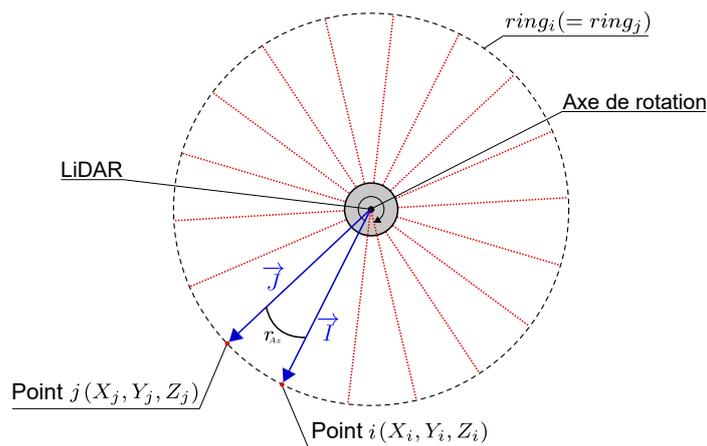


FIGURE 4.26 – Estimation de la résolution azimutale :  $r_{Az}$  est calculée en déterminant l'angle séparant les vecteurs  $\vec{I}$  et  $\vec{J}$ .

Pour chaque nappe, nous avons pu mesurer la valeur moyenne de  $r_{Az}$  en se basant sur ces mesures d'angles opérées entre les points consécutifs. Nous avons également pu mesurer l'écart-type  $\sigma_{Az}$  de la résolution azimutale sur chacune des nappes.

#### 4.4.2.2 Étude des angles d'élévation

Après avoir récupéré les valeurs indiquées pour les deux premiers modèles à disposition (Velodyne et BlenSor), nous avons pu appliquer ce test sur des données mesurées via le capteur  $L_{ref}$  équipant notre plateforme. Les résultats obtenus sont résumés dans le (Tab 4.2).

| ring | $\mathcal{V}_{ring}$ |                 |                 |                |                |
|------|----------------------|-----------------|-----------------|----------------|----------------|
|      | $M_{L_{const}}$      | $M_{L_{synth}}$ | $M_{L_{estim}}$ |                |                |
|      |                      |                 | $\nu = 0km/h$   | $\nu = 40km/h$ | $\nu = 70km/h$ |
| 1    | -25.000              | -25.000         | -25.000         | -25.000        | -25.000        |
| 2    | -15.639              | -15.639         | -15.639         | -15.639        | -15.639        |
| 3    | -11.310              | -11.310         | -11.310         | -11.310        | -11.310        |
| 4    | -8.843               | -8.843          | -8.843          | -8.843         | -8.843         |
| 5    | -7.254               | -7.254          | -7.254          | -7.254         | -7.254         |
| 6    | -6.148               | -6.148          | -6.148          | -6.148         | -6.148         |
| 7    | -5.333               | -5.333          | -5.333          | -5.333         | -5.333         |
| 8    | -4.667               | -4.667          | -4.667          | -4.667         | -4.667         |
| 9    | -4.000               | -4.000          | -4.000          | -4.000         | -4.000         |
| 10   | -3.667               | -3.667          | -3.667          | -3.667         | -3.667         |
| 11   | -3.333               | -3.333          | -3.333          | -3.333         | -3.333         |
| 12   | -3.000               | -3.000          | -3.000          | -3.000         | -3.000         |
| 13   | -2.667               | -2.667          | -2.667          | -2.667         | -2.667         |
| 14   | -2.333               | -2.333          | -2.333          | -2.333         | -2.333         |
| 15   | -2.000               | -2.000          | -2.000          | -2.000         | -2.000         |
| 16   | -1.667               | -1.667          | -1.667          | -1.667         | -1.667         |
| 17   | -1.333               | -1.333          | -1.333          | -1.333         | -1.333         |
| 16   | -1.000               | -1.000          | -1.000          | -1.000         | -1.000         |
| 19   | -0.667               | -0.667          | -0.667          | -0.667         | -0.667         |
| 20   | -0.333               | -0.333          | -0.333          | -0.333         | -0.333         |
| 21   | 0.000                | 0.000           | ∅               | ∅              | ∅              |
| 22   | 0.333                | 0.333           | 0.333           | 0.333          | 0.333          |
| 23   | 0.667                | 0.667           | 0.667           | 0.667          | 0.667          |
| 24   | 1.000                | 1.000           | 1.000           | 1.000          | 1.000          |
| 25   | 1.333                | 1.333           | 1.333           | 1.333          | 1.333          |
| 26   | 1.667                | 1.667           | ∅               | 1.667          | 1.667          |
| 27   | 2.333                | 2.333           | 2.333           | 2.333          | 2.333          |
| 28   | 3.333                | 3.333           | ∅               | 3.333          | ∅              |
| 29   | 4.667                | 4.667           | 4.667           | 4.667          | 4.667          |
| 30   | 7.000                | 7.000           | ∅               | 7.000          | 7.000          |
| 31   | 10.333               | 10.333          | ∅               | 10.333         | 10.333         |
| 32   | 15.000               | 15.000          | ∅               | 15.000         | 15.000         |

∅ : Pas de point mesuré pour cette nappe au sein de l'échantillon étudié.

TABLE 4.2 – Étude des angles d'élévation : Comparaison des angles d'élévation pour les différents modèles de capteur correspondant au Velodyne VLP-32C utilisé.

On peut dans un premier temps constater que pour les trois types de modèles ( $M_{L_{const}}$ ,  $M_{L_{synth}}$  et  $M_{L_{estim}}$ ) les différents angles d'élévation  $\mathcal{V}_{ring}$  sont parfaitement cohérents. De plus, l'analyse que nous avons pu faire sur les points appartenant à une même nappe nous ont permis de constater qu'aucun écart n'est mesuré ( $\sigma_{\mathcal{V}} = 0$ ). Tous ces points sont associés à un angle d'élévation  $\mathcal{V}_{ring}$  strictement identique. On en déduit donc que ces angles ne sont pas mesurés, mais imposés lors des mesures, en fonction de l'index *ring*.

#### 4.4.2.3 Étude la résolution azimutale

Concernant  $r_{Az}$ , aucune information n'est fournie par BlenSor. Son paramétrage est laissé libre à l'utilisateur, tout comme sa fréquence de rotation  $F$ . Du côté du constructeur, on a vu que la  $r_{Az}$  varie en fonction de  $F$ . Au sein des jeux de données à disposition, deux fréquences  $F$  différentes sont utilisées :  $20Hz$  pour l'acquisition à  $\nu = 40km/h$  et  $10Hz$  pour les acquisitions à  $\nu = 0km/h$  et  $\nu = 70km/h$ . Les résultats obtenus sont synthétisés dans le (Tab 4.3).

On peut noter que certaines variations sont observables, notamment sur les acquisitions à  $F = 10Hz$ . Le plus grand écart angulaire observé par rapport à la valeur vraie  $r_{Az}$  annoncée (Tab 4.1) est mesurée à  $0.023^\circ$  et représentée en rouge (Tab 4.3). Plus le point mesuré sera éloigné du capteur, plus cette erreur aura un impact important. Pour une mesure de distance de 200 m, portée maximale annoncée pour ce capteur, cet écart angulaire engendre une erreur de positionnement du point  $e = \pm 8cm$ .

| ring    | $r_{Az}$                 |                           |                           |
|---------|--------------------------|---------------------------|---------------------------|
|         | $\nu = 0km/h   F = 10Hz$ | $\nu = 40km/h   F = 20Hz$ | $\nu = 70km/h   F = 10Hz$ |
| 1       | 0.200°                   | 0.395°                    | 0.201°                    |
| 2       | 0.199°                   | 0.395°                    | 0.200°                    |
| 3       | 0.199°                   | 0.394°                    | 0.200°                    |
| 4       | 0.199°                   | 0.394°                    | 0.201°                    |
| 5       | 0.199°                   | 0.394°                    | 0.201°                    |
| 6       | 0.199°                   | 0.394°                    | 0.199°                    |
| 7       | 0.199°                   | 0.395°                    | 0.205°                    |
| 8       | 0.199°                   | 0.395°                    | 0.200°                    |
| 9       | 0.199°                   | 0.395°                    | 0.200°                    |
| 10      | 0.200°                   | 0.395°                    | 0.201°                    |
| 11      | 0.200°                   | 0.395°                    | 0.199°                    |
| 12      | 0.201°                   | 0.395°                    | 0.200°                    |
| 13      | 0.201°                   | 0.395°                    | 0.200°                    |
| 14      | 0.206°                   | 0.395°                    | 0.203°                    |
| 15      | 0.203°                   | 0.395°                    | 0.200°                    |
| 16      | 0.201°                   | 0.395°                    | 0.199°                    |
| 17      | 0.204°                   | 0.395°                    | 0.201°                    |
| 16      | 0.223°                   | 0.395°                    | 0.202°                    |
| 19      | 0.205°                   | 0.395°                    | 0.200°                    |
| 20      | 0.205°                   | 0.395°                    | 0.200°                    |
| 21      | 0.209°                   | 0.395°                    | 0.202°                    |
| 22      | 0.200°                   | 0.395°                    | 0.201°                    |
| 23      | 0.201°                   | 0.395°                    | 0.201°                    |
| 24      | 0.203°                   | 0.395°                    | 0.201°                    |
| 25      | 0.203°                   | 0.395°                    | 0.202°                    |
| 26      | ∅                        | 0.395°                    | 0.202°                    |
| 27      | ∅                        | 0.395°                    | 0.202°                    |
| 28      | ∅                        | 0.395°                    | ∅                         |
| 29      | 0.202°                   | 0.395°                    | 0.203°                    |
| 30      | ∅                        | 0.395°                    | 0.206°                    |
| 31      | ∅                        | 0.395°                    | 0.203°                    |
| 32      | ∅                        | 0.395°                    | 0.206°                    |
| EMQ     | 0.0053°                  | 0.0049°                   | 0.0023°                   |
| Moyenne | 0.2024°                  | 0.3950°                   | 0.2014°                   |

∅ : Pas de point mesuré pour cette nappe au sein de l'échantillon étudié.

TABLE 4.3 – Étude de la résolution azimutale : Comparaison des angles d'élévation pour les différents modèles de capteur correspondant au Velodyne VLP-32C utilisé.

Dans un souci d'optimisation des traitements, les octrees utilisés ne s'appuient que sur les points situés dans une sphère de rayon 100 m autour du capteur lors de leur création. Dans ces conditions, l'erreur  $e$  est réduite à  $e = \pm 4cm$ . Cet ordre de grandeur permet d'utiliser une résolution  $r_{max}$  fixée pour les octrees de 10 cm. En utilisant cette valeur pour  $r_{max}$ , cela permet d'englober  $e$  dans la représentation volumétrique de l'espace et d'avoir un impact négligeable sur la précision des données utilisées.

#### 4.4.2.4 Étude des mesures de distances

Enfin, concernant les mesures de distances, celles-ci sont directement liées à la précision de l'horloge utilisée, comme on a pu le voir avec le principe de mesure basé sur le temps de vol direct (Chap 2). Plus l'horloge sera précise, plus le temps  $\Delta t$ , séparant l'émission de la réception sera mesuré avec précision. Or, comme les distances mesurées dépendent directement de  $\Delta t$ , par l'équation (Eq 4.11), la précision de mesure de distance dépend directement de cette précision d'horloge.

$$D = \frac{1}{2} \cdot c \cdot \Delta t \quad (4.11)$$

Pour le LiDAR  $L_{ref}$  utilisé, cette précision d'horloge n'est pas annoncée par le fournisseur du capteur au sein de la fiche technique. En revanche, on retrouve la précision de mesure de distance, annoncée à  $\pm 3cm$ . Cette valeur a été déterminée par le constructeur via un « wall test » (Annexe D), dans des conditions ambiantes, pour l'ensemble des nappes laser.

### 4.4.3 Analyse du référencement

Au cours de ce chapitre nous avons présenté la méthode d'association utilisée pour référencer les données produites par le capteur de référence. La première analyse qui peut être réalisée sur cette méthode d'association concerne sa capacité à transférer les informations sémantiques contenues dans les données de synthèse du domaine *MAP* vers les données du domaine *CRA*.

#### 4.4.3.1 Capacité de transfert de l'information sémantique

Le premier test que nous pouvons réaliser sur la méthode de référencement présentée au cours de ce chapitre se concentre sur la capacité de transfert qu'il est possible d'atteindre entre les données de synthèse et les données acquises en conditions réelles. L'objectif de ce test est alors de mesurer le taux de données référencées automatiquement et qui pourront servir de référence lors des évaluations de différentes solutions de perception testées.

Pour réaliser cette opération, un échantillon composé de 300 scans  $scan_{réel,i}$  issus du capteur de référence  $L_{ref}$  sont prélevés du jeu de données utilisé. Ces scans sont ensuite filtrés avant d'être structurés et enfin, d'être annotés par notre méthode de référencement, présentée au cours de ce chapitre.

**Filtrage des points non-exploitable.** Dans un premier temps, une étape de filtrage est appliquée sur les scans échantillons. Cette étape de filtrage est définie en fonction de trois hypothèses :

1. L'altitude  $Z$  des points : le capteur de référence  $L_{ref}$  est positionné sur le toit de notre plateforme (Chap 3). Par conséquent, les points situés à une élévation supérieure à celle du capteur ne sont pas pertinents dans la compréhension de la scène ou dans le cadre de fonctions ADAS. Ils peuvent donc être isolés du reste des données;

2. Les points trop proches : pour la même raison énoncée en 1), les points situés à moins de 1 m du capteur correspondent à des points représentant le toit de la plateforme et n'apportent aucune information sur l'environnement. Ils sont donc filtrés ;
3. Le couloir d'étude : le troisième filtrage est appliqué en fonction de l'orientation du véhicule et plus particulièrement en fonction de son angle de lacet  $yaw$ . Cet angle permet de connaître la direction du véhicule. Dans l'objectif de se rapprocher du volume spatial représenté par la carte sémantique 3D et ainsi, de réduire davantage les écarts entre données de synthèse et données réelles, ce second filtrage intervient sous la forme de couloir. Il est représenté (Fig 4.27) et a été paramétré par une largeur de 6 m de part et d'autre du capteur  $L_{ref}$ . L'orientation de ce couloir est définie en fonction de l'angle  $yaw$  associé au capteur et aucun filtrage n'est opéré de façon longitudinale. Cela permet de conserver l'ensemble des voies situées entre les glissières de sécurité et donc de conserver l'ensemble des éléments pertinents de la scène vis-à-vis des fonctions ADAS.

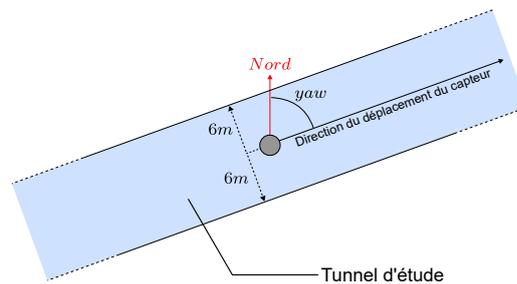


FIGURE 4.27 – Représentation du couloir d'étude : les points mesurés à l'extérieur de ce couloir (représentés en bleu) sont isolés et ne sont pas pris en compte au sein de nos travaux.

Le résultat de cette combinaison de filtres correspond à la définition d'un tunnel d'étude. Ce tunnel permet d'isoler les données non-pertinentes et ainsi, de conserver et cibler uniquement les données admises pertinentes vis-à-vis des fonctions ADAS. Le résultat de ce filtrage peut être constaté par les différences visibles (Fig 4.28-a et b).

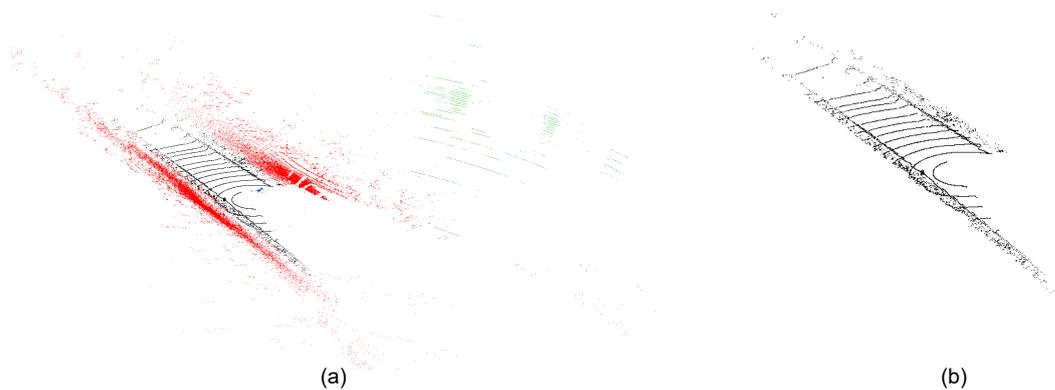


FIGURE 4.28 – Représentation du filtrage appliqué : (a) Scan produit par le capteur  $L_{ref}$  avant opération de filtrage. Les points représentés en rouge correspondent aux points filtrés en fonction du tunnel d'étude défini plus tôt. Les points verts correspondent à du bruit de mesure apparent par intermittence et qui est également filtré par le tunnel d'étude. Les points bleus correspondent aux points trop proches du capteur et qui correspondent à la carrosserie de la plateforme ; (b) Représentation du même scan après l'opération de filtrage. Les points restants sont tous pris en compte dans les différents traitements ultérieurs.

**Structuration des données.** Les scans filtrés  $scan_{réel,i}$  sont dans un second temps structurés sous la forme d'octrees  $o_{réel,i}$ . Au cours des expériences, les octrees ont été générés sur la base d'une résolution  $r_{max} = 10\text{ cm}$ . Le choix de cette résolution se justifie par le souhait de conserver l'ensemble des éléments

fixes de la scène. Or, la définition d'une résolution  $r_{max}$  plus faible (par ex : 20 cm ou 50 cm) peut amener à une perte d'informations, ciblée sur les objets de petites dimensions et liée à deux phénomènes :

- Une simplification de la scène amenant à une perte de certains détails ;
- Des voxels représentant des volumes plus importants, regroupant ainsi un plus grand nombre de points et amenant à la perte de certaines classes sémantiques moins représentées quantitativement.

Cela peut notamment être le cas avec des objets de petites dimensions, tels que les marquages au sol présents dans la carte sémantique 3D et dont la largeur correspond à 10 cm. Au cours de l'étape de structuration des données, avec l'utilisation d'une valeur  $r_{max}$  plus faible que 10 cm, les voxels représentant des points de marquages au sol englobent également des points associés à la route. Les marquages au sol sont entourés par la route et cette dernière classe sémantique peut être plus représentée et remporter la majorité des votes.

### Analyse du transfert d'informations sémantiques.

**Transfert entre des données de synthèse vers les données réelles :** une fois les données structurées, nous avons pu référencer chaque octree  $O_{réel,i}$  par association avec l'octree  $O_{synth}$ . Pour les 300 octrees référencés, nous avons orienté notre étude sur deux éléments.

Le premier est le taux de transfert des informations sémantiques de  $O_{synth}$  vers  $O_{réel,i}$ . Il s'agit ici de mesurer le pourcentage de données référencées. Pour ce faire, pour chaque  $O_{réel,i}$ , on calcule le nombre de voxels occupés. On mesure ensuite la quantité de voxels qui ont pu bénéficier de notre méthode de référencement et qui sont associés à une classe sémantique. Ce taux de référencement est mesuré, indépendamment, en fonction des différents niveaux de profondeurs  $d$  qui composent un octree, mais également de façon globale sur l'ensemble de l'octree étudié. Les résultats moyennés sur l'ensemble des 300 octrees  $O_{réel,i}$  composant l'échantillon sont présentés (Tab 4.4).

| Analyse du taux de transfert obtenu en fonction du niveau de profondeur dans l'octree |         |         |         |         |         |         |          |         |
|---|---------|---------|---------|---------|---------|---------|----------|---------|
| Niveau de profondeur ( $d$ )  | 16      | 15      | 14      | 13      | 12      | 11      | 10       | Global  |
| Résolution ( $r_d$ )  | 0.10 m  | 0.20 m  | 0.40 m  | 0.80 m  | 1.60 m  | 3.20 m  | 6.40 m   | /       |
| Taux de transfert   | 69.65 % | 78.06 % | 88.07 % | 98.51 % | 99.52 % | 99.84 % | 100.00 % | 78.58 % |

TABLE 4.4 – Taux de transfert en fonction du niveau de profondeur de l'octree : Chaque colonne représente un niveau de profondeur de l'octree (et une résolution) différent. La colonne « global » est quant à elle associée au taux de transfert moyen calculé sur les 16 niveaux de profondeur de l'octree.

**Remarque :** L'étude a été opérée dans le cas le plus défavorable, en mesurant le taux de transfert pour chaque octree  $O_{réel,i}$ , indépendamment les uns des autres. On l'a vu (Sec 4.3.2.3), les octrees référencés sont intégrés temporellement et utilisés sous la forme d'une carte octree  $O_{ref}$  lors des évaluations. Les taux de référencement mesurés sur  $O_{ref}$  auraient été avantageés par la redondance d'informations, obtenue sur plusieurs octrees  $O_{réel,i}$  consécutifs, qui aurait amené à des conditions plus favorables.

Le second élément étudié correspond aux types de voisinages utilisés au cours du référencement des octrees. Lors de cette étape, pour chaque association amenant à un transfert d'une information sémantique, le niveau de voisinage utilisé et permettant ce transfert est comptabilisé.

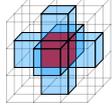
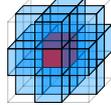
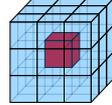
| Analyse du voisinage utilisé au cours du référencement |  |  |   |   |
|--|--|--|---|---|
| Type de voisinage                                      | Correspondance directe<br>$N_0$<br> | 6-connexes<br>$N_1$<br> | 18-connexes<br>$N_2$<br> | 26-connexes<br>$N_3$<br> |
| Taux d'exploitation moyen                              | 60.24 %  | 9.90 %   | 30.32 %   | 0.00 %  |

TABLE 4.5 – Analyse du voisinage utilisé

Les résultats obtenus sont présentés (Tab 4.5). Ces résultats permettent de voir que la majorité du transfert d'informations ne nécessite pas l'utilisation du voisinage et s'opère par correspondance directe. Ce résultat est obtenu grâce aux moyens mis en place et utilisés pour générer des données de synthèse représentativement proches des données réelles acquises. L'utilisation des structures octrees et des voxels permet également d'englober une partie des variations observables entre les données réelles et les données de synthèse, dues à l'idéalisation des données de synthèse et aux phénomènes extérieurs pouvant impacter les données acquises en conditions réelles.

On observe également que lorsqu'il n'y a pas de correspondance directe entre les voxels, l'élargissement du critère d'association avec le premier niveau de voisinage  $N_1$  (6-connexes) à un apport au référencement plus faible que le second niveau de voisinage  $N_2$  (18-connexes). Enfin, on observe que le dernier niveau de voisinage  $N_3$  (26-connexes) n'a aucun apport au cours du référencement. Cela signifie que si aucun voxel ne peut être associé lorsque l'association est élargie à  $N_2$ , l'utilisation de  $N_3$  ne permet pas d'améliorer le référencement en permettant une nouvelle association. Le voxel devant être associé conservera sa valeur neutre. Deux possibilités peuvent être envisagées : la première est de modifier  $N_3$  pour élargir davantage le voisinage pris en compte, au risque de perdre en pertinence lors des associations ; la seconde est de ne pas prendre en compte ce voisinage et de limiter l'association au voisinage  $N_2$ .

**Transfert entre les données de synthèse :** dans l'objectif de le compléter, nous avons réitéré ce test en modifiant les données utilisées. Nous avons alors remplacé les données réelles par des données de synthèse bruitées en utilisant les paramètres du simulateur BlenSor, introduits (Sec 4.2.2). En se basant uniquement sur des données de synthèse, cela nous permet d'étudier la méthode avec un aspect supplémentaire. Comme pour le test avec les données réelles, on peut dans un premier temps mesurer le taux de référencement et étudier les taux d'utilisation des différents voisinages au cours du référencement. En plus de cela, il est possible de vérifier l'exactitude du transfert réalisé, les données de synthèse étant annotées automatiquement à partir de la carte sémantique 3D utilisée au cours de la simulation. On connaît *a priori* la classe sémantique des données et il est possible de vérifier, après application du référencement, qu'on retrouve les classes originalement associées aux données.

Ce nouveau test a été réalisé pour quatre niveaux de bruit différents. Le premier est un niveau de bruit nul  $\sigma_0$  permettant de s'assurer que lorsque l'on cherche à associer les mêmes données, on retrouve logiquement un taux de référencement de 100 %, l'utilisation du premier niveau de voisinage à 100 % et qu'aucune erreur de classe sémantique n'est introduite par notre méthode de référencement. Les trois autres niveaux de bruits utilisés, impactant les distances mesurées par le capteur synthétisé (Sec 4.2.2), sont basés sur un modèle gaussien, centré sur 0 et avec des écarts-types respectifs :  $\sigma_{synth} = \pm 2cm$ ,  $\sigma_{synth} = \pm 10cm$  et  $\sigma_{synth} = \pm 20cm$ .

Les résultats obtenus, concernant le taux de transfert et l'étude du voisinage utilisé sont résumés dans le (Tab 4.6) et le (Tab 4.7). Comme attendu, lorsque le bruit de mesure est nul et que l'on réalise un transfert sur des données identiques, on retrouve un taux de transfert de 100.00 %, quelque soit la profondeur des voxels dans l'octree et en exploitant uniquement la correspondance directe  $N_0$ . Cela nous permet dans un premier temps de vérifier que notre méthode de référencement n'introduit pas d'erreurs de transfert au cours de son fonctionnement.

| Analyse du taux de transfert obtenu en fonction du niveau de profondeur dans l'octree |          |          |          |          |          |          |          |          |
|---|----------|----------|----------|----------|----------|----------|----------|----------|
| Niveau de profondeur ( $d$ )  | 16       | 15       | 14       | 13       | 12       | 11       | 10       | Global   |
| Résolution ( $r_d$ )  | 0.10 m   | 0.20 m   | 0.40 m   | 0.80 m   | 1.60 m   | 3.20 m   | 6.40 m   | /        |
| $\sigma_{synth} = 0cm$  | 100.00 % | 100.00 % | 100.00 % | 100.00 % | 100.00 % | 100.00 % | 100.00 % | 100.00 % |
| $\sigma_{synth} = \pm 2cm$  | 87.49 %  | 97.45 %  | 99.85 %  | 100.00 % | 100.00 % | 100.00 % | 100.00 % | 93.87 %  |
| $\sigma_{synth} = \pm 10cm$   | 85.44 %  | 96.60 %  | 99.58 %  | 100.00 % | 100.00 % | 100.00 % | 100.00 % | 92.31 %  |
| $\sigma_{synth} = \pm 20cm$   | 79.93 %  | 94.80 %  | 99.12 %  | 100.00 % | 100.00 % | 100.00 % | 100.00 % | 89.30 %  |

TABLE 4.6 – Taux de référencement en fonction des niveaux de bruit : Chaque colonne représente un niveau de profondeur de l'octree (et une résolution) différent. La colonne « global » est quant à elle associée au taux de transfert moyen calculé sur les 16 niveaux de profondeur de l'octree.

| Analyse du voisinage utilisé au cours du référencement en fonction du niveau de bruit |                              |                  |                   |                   |
|---|------------------------------|------------------|-------------------|-------------------|
| Type de voisinage   | Correspondance directe $N_0$ | 6-connexes $N_1$ | 18-connexes $N_2$ | 26-connexes $N_3$ |
| Taux d'exploitation moyen pour $\sigma_{synth} = 0cm$                                 | 100.00 %                     | 0.00 %           | 0.00 %            | 0.00 %            |
| Taux d'exploitation moyen pour $\sigma_{synth} = \pm 2cm$                             | 86.14 %                      | 6.61 %           | 7.58 %            | 0.00 %            |
| Taux d'exploitation moyen pour $\sigma_{synth} = \pm 10cm$                            | 80.26 %                      | 9.28 %           | 10.79 %           | 0.00 %            |
| Taux d'exploitation moyen pour $\sigma_{synth} = \pm 20cm$                            | 72.93 %                      | 13.32 %          | 14.08 %           | 0.00 %            |

TABLE 4.7 – Analyse du voisinage utilisé en fonction du bruit

En utilisant des données de synthèse bruitées (avec  $\sigma_{synth}$ ), on se rapproche des données acquises en conditions réelles. Ces deux tableaux nous permettent alors de confirmer, et de renforcer, les tendances observées avec le test analysant le transfert entre données de synthèse et données « réelles ». On constate que le taux de transfert croît rapidement dans les niveaux de profondeur de l'octree, jusqu'à atteindre 100 %. Concernant le voisinage utilisé, on observe bien que la majorité des transferts s'opère par correspondance directe  $N_0$ . Ensuite, bien que le phénomène soit moins marqué que dans le cadre du transfert vers des données réelles, on retrouve également que le voisinage  $N_2$  est plus utilisé que le voisinage  $N_1$  au cours des associations.

Enfin, comme annoncé au début de ce test, l'avantage de travailler avec des données de synthèse bruitées est que les classes sémantiques des voxels référencés sont pré-déterminées. Après transfert, on peut donc contrôler chaque voxel en comparant la classe sémantique issue du transfert à sa classe originale. Les résultats obtenus en fonction des différentes classes sémantiques sont résumés (Tab 4.8).

On constate alors que sur la plupart des classes, la méthode utilisée permet de transférer les bonnes classes sémantiques pour les 4 niveaux de bruits utilisés. On conserve un taux d'exactitude des classes supérieur à 85 %. On remarque cependant qu'une classe est particulièrement impactée par les variations, il s'agit de « line », correspondant aux marquages au sol. De par la petite taille de cet objet et sa position contextuelle, entourée par la classe « road », cette classe est plus sensible aux variations sur les données.

| Classes sémantiques | Pourcentage de données correctement référencées en fonction du niveau de bruit |                            |                             |                             |
|---------------------|--|----------------------------|-----------------------------|-----------------------------|
|                     | $\sigma_{synth} = 0cm$   | $\sigma_{synth} = \pm 2cm$ | $\sigma_{synth} = \pm 10cm$ | $\sigma_{synth} = \pm 20cm$ |
| road                | 100.00%  | 99.03%                     | 98.77%                      | 98.33%                      |
| border              | 100.00%  | 93.84%                     | 91.92%                      | 89.00%                      |
| line                | 100.00%  | 64.76%                     | 58.80%                      | 50.28%                      |
| barrier             | 100.00%  | 98.76%                     | 98.61%                      | 98.08%                      |
| sign                | 100.00%  | 89.49%                     | 87.32%                      | 85.11%                      |
| vegetation          | 100.00%  | 95.43%                     | 93.49%                      | 89.67%                      |
| embankment          | 100.00%  | 99.24%                     | 98.91%                      | 98.45%                      |

TABLE 4.8 – Analyse de l'exactitude des informations sémantiques transférées

#### 4.4.3.2 Impact de la précision de la localisation sur le référencement

Comme nous avons pu le voir au cours de ce chapitre, le système de localisation représente l'une des sources de données exploitées par la méthode de référencement automatique mise en place. Les données de localisation  $Pose_i$ , produites par ce système, sont utilisées pour deux objectifs : le premier est le géoréférencement des données de perception ; le second est l'utilisation de ces mêmes poses pour paramétrer le modèle de capteur simulé et le positionner au sein de la carte sémantique 3D, dans l'environnement simulé.

Ces conditions permettent de disposer de données réelles et de données de synthèse exprimées dans un même système de coordonnées. Grâce à cela, nous avons pu baser notre méthode de référencement sur des critères d'association géométriques simples à définir.

Dans le but d'étudier l'impact d'une perte de précision du système de localisation (représenté Fig 4.29) sur le référencement automatique des données, nous nous sommes appuyés sur le même échantillon de données. Différents niveaux de bruit ont ainsi été introduits sur les données de localisation, comme décrit dans la suite de cette section.

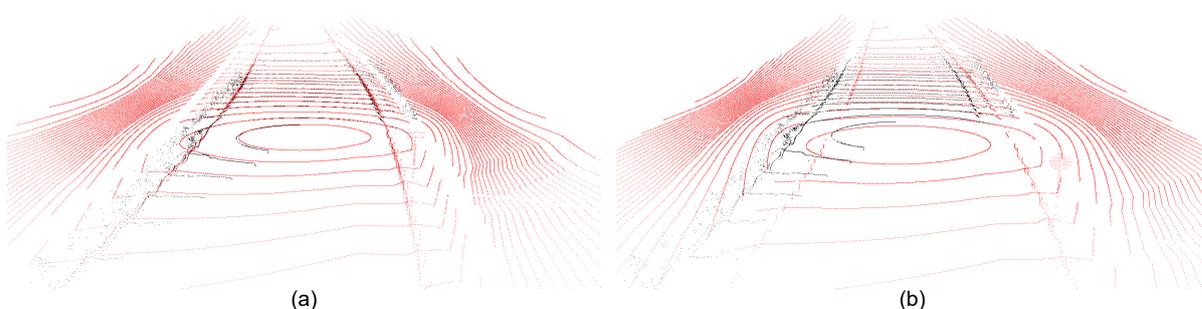


FIGURE 4.29 – Représentation des erreurs de localisation : L'impact des imprécisions du système de localisation peut être observé en comparant ces deux figures avec : (a) Le scan de synthèse  $scan_{synth,i}$  (représenté en rouge) et le scan réel  $scan_{réel,i}$  (en noir) superposés et avec des informations de localisation précises ; (b) les mêmes scans superposés mais cette fois-ci, avec des imprécisions sur les données de localisation utilisées au cours des traitements.

**Déroulement de l'expérience.** En se basant sur l'approche mise en place, nous avons introduit deux étapes supplémentaires nommées « Introduction du bruit de localisation » et « Correction du bruit » (Fig 4.30).

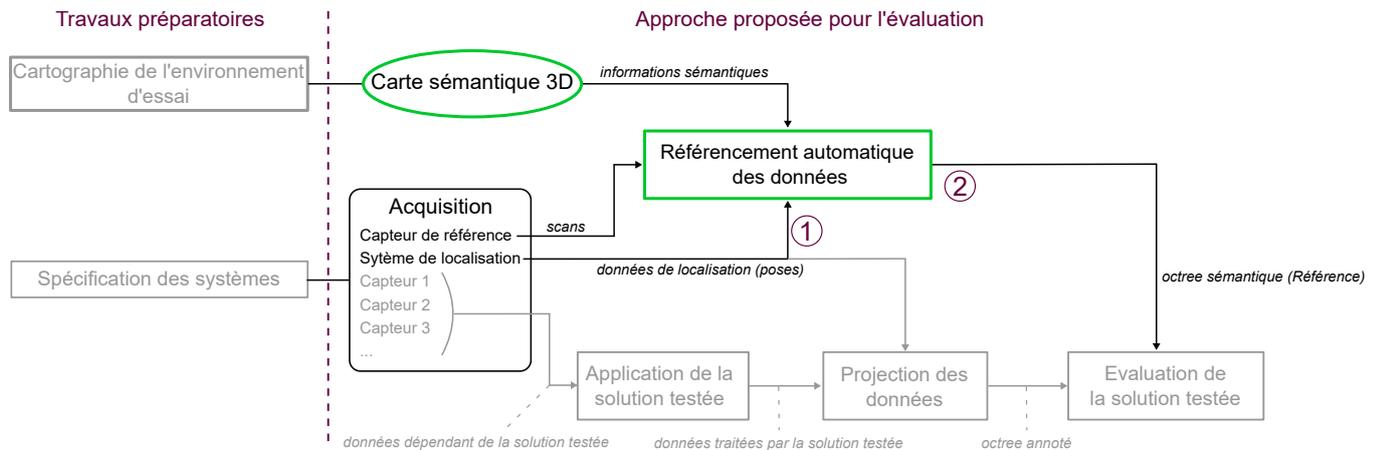


FIGURE 4.30 – Approche modifiée pour les expériences avec l'intégration de deux étapes : « Introduction du bruit de localisation » en 1 et « Correction du bruit » en 2. Celles-ci sont détaillées sous cette figure. Les étapes représentées en gris ne sont pas concernées par l'expérience.

La première étape « Introduction du bruit de localisation » consiste à introduire du bruit de mesure, noté  $\epsilon_{loc}$ , sur les poses du capteur de référence  $Pose_i$ . Cela permet d'impacter le géoréférencement des données produites par  $L_{ref}$ , mais également, la génération des données de synthèse, qui est impactée par ce même bruit. Pour bruitez ces données, nous nous sommes basés sur l'hypothèse qu'un bruit de localisation dans un environnement non obstrué et sans phénomène de multi-trajets correspond à un modèle gaussien [119].  $\epsilon_{loc}$  correspond ainsi à un bruit additif blanc gaussien, centré sur zéro.

Une fois  $\epsilon_{loc}$  introduit, les données sont traitées en suivant les étapes de l'approche. Dans un premier temps des données de synthèse sont générées. L'ensemble des données sont ensuite structurées sous forme d'octrees avant d'appliquer la méthode de référencement mise en place. Une fois ces étapes réalisées, on obtient un octree à partir des 300 scans et dont le référencement a été impacté par le bruit  $\epsilon_{loc}$ . On le nomme  $O_\epsilon$ .

Comme indiqué plus tôt dans ce chapitre, il n'est pas possible de connaître en amont du référencement, les classes sémantiques des voxels pour un octree basé sur des données de mesure. Dans l'objectif de mesurer le pourcentage de voxels correctement annotés malgré l'introduction de  $\epsilon_{loc}$ , un second octree  $O_{ref}$  est produit à partir de ce même échantillon, mais cette fois-ci, sans ajouter de bruit de localisation. Cet octree  $O_{ref}$  constitue alors notre référence et nous permet ainsi de mener des comparaisons selon les classes sémantiques associées aux voxels.

Enfin, l'étape « Correction du bruit » permet de corriger le bruit  $\epsilon_{loc}$  introduit et de repositionner l'octree  $O_\epsilon$  avec son géoréférencement original. Cette étape nous permet ainsi d'avoir les octrees  $O_\epsilon$  et  $O_{ref}$  parfaitement alignés et leurs voxels respectifs peuvent être comparés. Il est alors possible de déterminer le pourcentage de voxels  $V_{O_\epsilon}$  pour lesquels les valeurs de l'attribut *label* sont cohérentes avec les voxels  $V_{O_{ref}}$ . Un pourcentage peut être déterminé en fonction de chaque classe sémantique pour étudier la différence d'impact sur les différentes classes sémantiques prises en compte.

**Analyse des résultats.** Cette expérience a été réalisée pour différentes valeurs d'écart-types  $\sigma_{loc}$ , de plus en plus importantes. Cela nous a permis d'étudier la progression de la dégradation de référencement en fonction de l'écart-type  $\sigma_{loc}$  fixé. Les six valeurs utilisées sont les suivantes :  $\pm 10cm$ ,  $\pm 20cm$ ,  $\pm 50cm$ ,  $\pm 1.00m$ ,  $\pm 2.50m$ ,  $\pm 5.00m$ . Les résultats obtenus en fonction des différentes classes sémantiques introduites (Chap 3) sont résumés au sein du (Tab 4.9).

| Classes sémantique | Ecart-types $\sigma_{loc}$ |            |            |             |             |             |
|--------------------|----------------------------|------------|------------|-------------|-------------|-------------|
|                    | $\pm 10cm$                 | $\pm 20cm$ | $\pm 50cm$ | $\pm 1.00m$ | $\pm 2.50m$ | $\pm 5.00m$ |
| road               | 93.59%                     | 93.51%     | 92.25%     | 89.90%      | 64.64%      | 51.38%      |
| border             | 85.90%                     | 84.76%     | 79.77%     | 67.68%      | 31.45%      | 18.85%      |
| line               | 42.44%                     | 37.31%     | 21.80%     | 11.34%      | 4.51%       | 1.60%       |
| barrier            | 87.01%                     | 79.58%     | 67.66%     | 53.95%      | 23.23%      | 15.18%      |
| sign               | 48.84%                     | 37.28%     | 27.48%     | 23.21%      | ∅           | ∅           |
| vegetation         | 80.93%                     | 76.11%     | 62.56%     | 41.70%      | 17.67%      | 8.55%       |
| embankment         | 75.08%                     | 67.18%     | 46.27%     | 31.61%      | 13.10%      | 8.26%       |

TABLE 4.9 – Synthèse des résultats de référencement impacté par un bruit de localisation : pourcentages de voxels correctement annotés lors du référencement en fonction des différentes classes sémantiques et en fonction du niveau de bruit introduit.

Ces mêmes résultats sont représentés graphiquement (Fig 4.31).

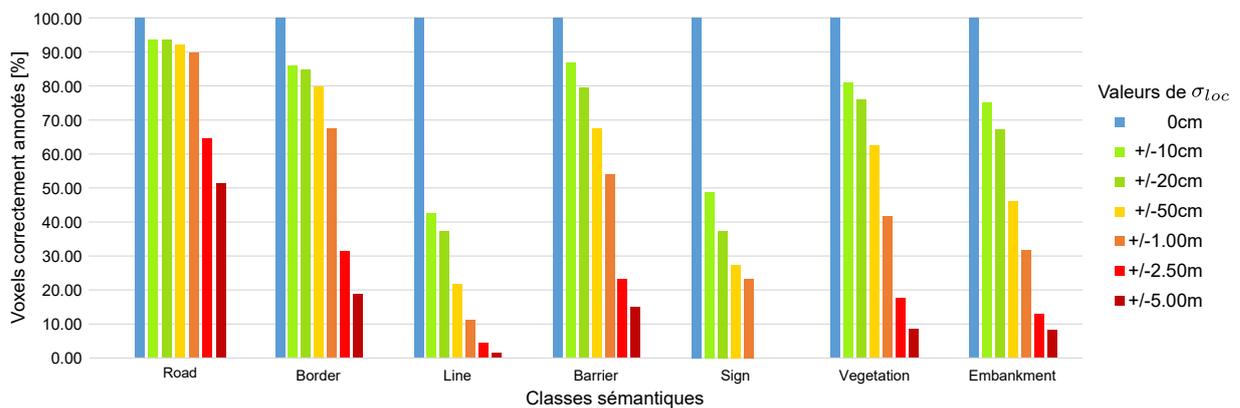


FIGURE 4.31 – Représentation graphique de l'impact du bruit de localisation sur le référencement : avec le pourcentage de voxels correctement annotés en fonction de la classe sémantique et du niveau de bruit introduit sur les données.

A travers ces résultats, plusieurs éléments peuvent être observés :

Le premier effet observé et attendu, est que plus l'ordre de grandeur des imprécisions sur les données de localisation augmente, moins le référencement des données obtenu est précis.

Le second élément correspond à la différence d'impact que peut avoir un même niveau de bruit, en fonction des classes sémantiques rencontrées. Pour étudier cet aspect dans sa globalité, on peut tout d'abord prendre en compte la répartition quantitative des classes, initialement observée au sein de l'octree  $O_{ref}$ . Cette répartition est présentée (Tab 4.10).

Sur la (Fig 4.31), on peut observer dans un premier temps que les classes les plus impactées correspondent aux classes « line », « embankment » et « sign ». Ces impacts importants se justifient dans un premier temps par la faible représentation de ces classes au sein de l'octree  $O_{ref}$  utilisé au cours de cette expérience. Ces classes sont moins représentées quantitativement et la moindre variation amène un impact plus important.

| Classes sémantiques | Représentation | Représentation [%] |
|---------------------|----------------|--------------------|
| road                | 416 478 voxels | 45.22              |
| border              | 246 835 voxels | 26.80              |
| line                | 19 055 voxels  | 2.07               |
| barrier             | 129 114 voxels | 14.02              |
| sign                | 64 voxels      | 0.01               |
| vegetation          | 81 834 voxels  | 8.89               |
| embankment          | 27 542 voxels  | 2.99               |

TABLE 4.10 – Répartition des classes dans l'échantillon d'étude : représentation de chaque classe mesurée en fonction du nombre de voxels et de l'équivalent en pourcentage.

On peut cependant, remarquer que la classe « embankment », représentée par un ordre de grandeur proche de la classe « line », semble moins impactée par les imprécisions de localisation que cette dernière. Cela se justifie par un second élément : la position des classes dans la scène. En effet, la classe « embankment » représente les fossés et talus, retrouvés sur les bords de la scène. Les voxels correspondant à cette classe sont regroupés sur ces bords de cette scène et sont en partie isolés. Ainsi, lorsque l'imprécision de localisation décale les voxels, en étudiant le voisinage on retombe généralement sur ce même label « embankment ». Si l'écart de position est trop important, ils ne sont pas référencés et ne sont pas comptabilisés.

A l'inverse, les objets des classes « line » et « sign » correspondent aux marquages au sol et panneaux de signalisation. Ces objets sont situés au centre de la scène et sont entourés par d'autres classes (par exemple : les marquages au sol sont entourés par des voxels de classe route « road »).

La méthode de référencement que nous proposons n'inclut pas d'étape d'alignement des données. Par conséquent, aucune étape ne permet de compenser le phénomène représenté (Fig 4.29). C'est pourquoi les voxels de  $O_\epsilon$  référencés avec un *label* cohérent avec ceux de  $O_{ref}$  sont majoritairement associés aux classes sémantiques les plus représentées dans l'octree.

Ces différentes observations permettent de soulever le fait que selon le type d'algorithme évalué, une importance particulière devra être accordée à la précision des mesures réalisées par le système de localisation. En fonction de cette précision, les données utilisées dans le processus d'évaluation pourront disposer d'un référencement précis et non biaisé, permettant d'évaluer des solutions de perception de façon pertinente. Cependant, elle met également en avant une perspective qui est : l'ajout d'une étape d'alignement des données. Cette étape permettra alors de compenser, dans une certaine mesure, les imprécisions des poses mesurées par le système de localisation et d'améliorer la pertinence des transferts réalisés entre les données de synthèse et les données de mesure.

**Remarque :** Au cours de cette expérience, nous nous sommes concentrés sur les erreurs de positionnement pouvant être commises par le système de localisation. Cette étude pourra être approfondie en étudiant le comportement de la méthode de référencement face à des imprécisions de plus faible niveau avec des écarts-types sur le positionnement du même ordre de grandeur que ceux observés lors des acquisitions. L'impact des imprécisions de mesures d'attitude pourront également être étudiées indépendamment des erreurs de positionnement, mais également, de façon cumulée.

## 4.5 Perspectives

Dans l'objectif d'améliorer les capacités et la robustesse de la méthode de référencement implémentée et testée grâce aux expériences de ce chapitre, plusieurs pistes d'étude peuvent être abordées. Ces premières pistes se répartissent sur deux aspects : les critères d'association utilisés et l'utilisation du géoréférencement dans ces associations.

### 4.5.1 Critères d'association

Le premier aspect correspond aux critères d'association utilisés. La méthode actuelle repose sur la correspondance directe, pouvant s'élargir sur trois niveaux de voisinages et permettant d'annoter  $\pm 70\%$  des données au niveau de profondeur le plus haut de l'octree. En observant les données référencées, on s'aperçoit de deux phénomènes :

- Plus la résolution  $r$  diminue, plus les données sont simplifiées, représentant la scène de façon grossière et plus le taux de transfert augmente jusqu'à atteindre 100 % ;
- Le second est que la majorité des données n'ayant pu être annotées, en utilisant des hautes résolutions, sont situées sur les bords de la scène et correspondent à de la végétation.

Ces phénomènes se justifient par le fait que la végétation, étant en perpétuelle évolution, notamment du point de vue des feuillages, ne représente pas un élément probant à cartographier. Elle n'est pas représentée fidèlement dans la carte sémantique 3D ce qui peut donc amener aux différences de perception observées entre les données réelles, s'appuyant sur la végétation existante et les données de synthèse.

Dans l'objectif d'améliorer davantage ce taux transfert, plusieurs possibilités peuvent être évoquées.

#### 4.5.1.1 Approche de recherche ascendante

Une première piste serait de modifier ou compléter la méthode de référencement actuelle en organisant la recherche d'association et en utilisant les différents niveaux d'échelle, ainsi que liens de parenté des structures en octree. Ainsi, en se basant sur le raisonnement introduit (Sec 4.3.2.3), lorsque les critères d'association ne sont pas respectés pour deux voxels,  $V_{o_{réel}}$  et  $V'_{o_{synth}}$ , situés à une même profondeur, il serait possible d'étudier le voxel parent de  $V'_{o_{synth}}$  et de récupérer son information sémantique s'il est occupé. S'il n'est pas occupé, il serait possible de remonter encore d'un niveau et cela de façon répétée jusqu'à rencontrer un voxel parent occupé. Ce raisonnement permettrait d'améliorer grandement le taux de transfert d'après les tests réalisés (Sec 4.4.3).

Cependant, plus le niveau de profondeur diminue plus la scène est simplifiée. L'information sémantique est alors associée à un voxel représentant un plus grand volume et est moins précise. Un critère de pertinence  $\mathbb{P}$  pourrait donc être mis en place pour accompagner cette méthode. Lorsque une association est réalisée à la profondeur de  $V_{o_{réel}}$ ,  $\mathbb{P}$  serait à sa valeur maximale. En revanche, s'il est nécessaire de remonter dans l'arborescence pour trouver un voxel  $V'_{o_{synth}}$  occupé, ce critère serait impacté négativement à chaque niveau traversé.

#### 4.5.1.2 Approche de recherche descendante

Dans le même esprit que l'approche ascendante, la perspective évoquée ici serait d'appliquer la méthode de référencement en parcourant l'octree  $O_{r\acute{e}el}$  de la profondeur la moins profonde  $d = 1$  (associée à la résolution la plus basse), jusqu'à descendre à la plus profondeur  $d = 16$  (associée à la haute résolution  $r_{max}$ ). En appliquant ce raisonnement sur l'ensemble des seize niveaux de profondeur des octrees, l'ensemble des voxels de  $O_{r\acute{e}el}$  seront annotés grâce aux taux de transfert obtenus dans les premiers niveaux de profondeur.

Pour les mêmes raisons que celles évoquées pour l'approche ascendante, un critère de pertinence  $\mathbb{P}$  peut être mis en place. Cependant à l'inverse de l'approche descendante celui-ci aurait une valeur correspondant à une faible pertinence. En descendant dans des niveaux de profondeurs de l'octree plus importants, les voxels  $V_{O_{r\acute{e}el}}$  qui sont mis à jour par association avec les voxels  $V'_{O_{synth}}$ , verrait leur critère  $\mathbb{P}$  évoluer. A chaque nouvelle association dans un niveau de profondeur supérieur au précédent, la valeur de  $\mathbb{P}$  se renforcerait jusqu'à atteindre sa valeur maximale en cas d'association au niveau de profondeur  $d = 16$ .

#### 4.5.1.3 Adaptation en fonction de la classe sémantique

Une autre perspective envisagée serait d'adapter les associations réalisées en fonction de la classe sémantique traitée. L'idée serait d'adapter la résolution en fonction de la nature de l'objet sémantique traité. Pour une classe « line », correspondant à un objet sémantique de petite dimension, le transfert serait opéré avec la plus haute résolution  $r_{max}$ . A l'inverse, pour des objets pouvant être représentés plus grossièrement tel que la végétation, l'association peut être faite en utilisant des voxels situés à une résolution inférieure à  $r_{max}$ .

Cette perspective nécessite de connaître *a priori* la classe sémantique potentiellement transférée lors d'une association. Les transferts seraient alors menés dans le sens contraire à ceux actuels. On partirait alors des voxels appartenant à  $O_{synth}$  pour rechercher des association avec les voxels des octrees  $O_{r\acute{e}el,i}$ . L'utilisation de cette perspective permettrait d'améliorer les capacités de transfert mesurés, notamment en augmentant les transferts liés à la végétation, qui comme on l'a vu plus tôt, restent le problème majeur actuel.

### 4.5.2 Utilisation du géoréférencement

Le second aspect est le géoréférencement sur lequel s'appuie le référencement pour associer géométriquement les données. Cependant, la pertinence du référencement mené avec la méthode actuelle dépend, en partie, de la précision des mesures du système de localisation et des poses utilisées  $Pose_i$ .

Dans l'objectif de diminuer cette dépendance et conserver un haut niveau de précision malgré les imprécisions pouvant impacter les mesures du système de localisation, une perspective serait d'introduire une étape d'alignement. On peut notamment penser à des approches retrouvées dans le SLAM, basées sur des LiDARs et qui utilisent des alignements entre différents scans. On retrouve ces méthodes dans [120, 121, 122, 123]. Certains travaux exploitent ces alignements en se basant directement sur les octrees comme dans [124, 125].

Cette étape aurait pour objectif d'aligner géométriquement les données de synthèse et les données réelles. Si des résultats probants sont obtenus en ajoutant un alignement des données, cette perspective permettrait à la fois de réduire la dépendance au système de localisation, mais également, de pouvoir agrandir les possibilités en utilisant des systèmes de localisation moins développés et moins coûteux que celui utilisé au cours de ces travaux de thèse.

## 4.6 Conclusion

Dans ce chapitre, nous avons détaillé la méthode de référencement permettant d'annoter automatiquement et sémantiquement les données acquises par le capteur de référence. Dans notre cas, le référencement passe par l'adaptation de domaine avec des associations réalisées entre des données de synthèse et des données obtenues en conditions réelles. Ces associations sont utilisées pour transférer les informations sémantiques depuis les données de synthèse et référencer les données « réelles ».

- On a pu voir que le référencement automatique passait par deux sous-étapes. Dans un premier temps, la « prédiction des annotations » permet de générer les données de synthèse à partir de la carte sémantique 3D, des mesures du système de localisation effectuées lors de l'acquisition en conditions réelles, ainsi que d'un modèle de capteur simulé. En utilisant un modèle de capteur connu et maîtrisé, correspondant au capteur de référence embarqué, les données de synthèse générées se rapprochent des données obtenues en conditions réelles. Cela permet d'optimiser les performances et le taux de transfert réalisés au cours de la seconde sous-étape. Ces données sont annotées automatiquement, au cours de leur génération, en exploitant les labels sémantiques des facettes qui composent la carte sémantique 3D ;
- Dans un second temps, le « Transfert d'informations sémantiques » est détaillé. Diverses méthodes d'association ont été introduites, avant de présenter la méthode utilisée au cours de nos travaux. Cette méthode passe par la structuration des données sous forme d'octree, permettant d'optimiser les traitements et de travailler à plusieurs niveaux d'échelle. Nous avons opéré des modifications sur la librairie OctoMap afin d'intégrer les informations sémantiques dans les voxels et d'avoir une transmissions des informations sémantiques cohérentes dans les différents niveaux d'échelle des octrees. Enfin, les associations effectuées entre les octrees basés sur les données de synthèse générées et ceux basés sur les données issues du capteur de référence, s'appuient sur des critères d'association géométriques et contextuels ;
- A l'issue de cette méthode, on obtient des données « réelles » annotées, pouvant servir de référence dans le cadre de l'évaluation de solutions de perception. Bien que cette première utilisation soit au cœur de ce sujet de thèse, d'autres utilisations peuvent être associées à ces données, comme dans le cadre du Machine Learning, où il y a un réel besoin de disposer de données annotées pour entraîner les algorithmes. Les données référencées par notre méthode pourraient alors servir de base pour un entraînement sur données réelles. De plus, dans certaines conditions (par exemple de nuit), l'utilisation d'une caméra de contexte peut être complexifiée par les conditions environnantes. L'interprétation humaine et l'annotation de données LiDAR peuvent ainsi être difficiles, voire impossibles. L'utilisation de notre approche peut également apporter une solution à ce type de scénarios en facilitant la création de la base de données référencées.

- Enfin, dans l'objectif de proposer une preuve de concept, plusieurs expériences ont été réalisées pour analyser les capacités de référencement de notre méthode, ainsi que sa robustesse face aux imprécisions de localisation. Des tests avec plusieurs niveaux de perturbations ont permis de mesurer les impacts sur le référencement et de faire ressortir les classes sémantiques plus sensibles à ces perturbations. Ces derniers tests ont notamment permis d'amener une des perspectives qui est d'intégrer une étape d'alignement des données. La méthode proposée repose sur une localisation précise. Ajouter cette étape d'alignement permettra de rendre la méthode plus robuste en permettant de compenser, dans une certaine mesure, les imprécisions de localisation pouvant apparaître.

Le travail décrit dans ce chapitre nous a permis de mettre en œuvre le concept proposé dans le (Chap 3), d'appliquer la méthode de référencement précédant l'évaluation de solutions. Dans le chapitre suivant, les dernières étapes constituant l'approche seront détaillées, avant de mener des expériences faisant appel à la globalité de l'approche de validation.

# Chapitre 5

## Évaluation de solutions de perception

### Sommaire

---

|       |  |     |
|-------|--|-----|
| 5.1   | Introduction . . . . .   | 113 |
| 5.2   | Processus d'évaluation . . . . .                               | 114 |
| 5.2.1 | Application de la solution testée . . . . .                    | 114 |
| 5.2.2 | Projection des données . . . . .                               | 114 |
| 5.2.3 | Évaluation de la solution testée . . . . .                     | 115 |
| 5.2.4 | Métriques d'évaluation quantitative . . . . .                  | 116 |
| 5.2.5 | Évaluation qualitative . . . . .                               | 117 |
| 5.3   | Évaluation de solutions de perception . . . . .                | 118 |
| 5.3.1 | Solutions de perception évaluées . . . . .                     | 118 |
| 5.3.2 | Évaluation des solutions de perception basées LiDAR . . . . .  | 121 |
| 5.3.3 | Évaluation de la solution de perception basée Caméra . . . . . | 127 |
| 5.4   | Perspectives . . . . .   | 136 |
| 5.4.1 | Évaluation de solutions connues . . . . .                      | 136 |
| 5.4.2 | Prise en compte des variations temporelles . . . . .           | 136 |
| 5.4.3 | Diversification des systèmes évalués . . . . .                 | 136 |
| 5.5   | Conclusion . . . . .   | 137 |

---

### 5.1 Introduction

Dans le (Chap 4), nous avons présenté les différentes étapes qui composent la méthode de référencement développée au cours de ce travail de thèse. Les données acquises en conditions réelles, grâce au capteur de référence  $L_{ref}$ , sont référencées automatiquement par cette méthode. Une fois le référencement effectué, ces données augmentées peuvent servir de référence pour évaluer une solution de perception donnée.

Ce cinquième chapitre s'intéresse à la dernière partie de notre approche qui est : l'évaluation des performances de solutions de perception tierces. Il débute par l'introduction des étapes permettant à une solution de perception testée, qu'on nommera  $T$ , d'être évaluée en fonction des données de référence produites par notre approche (Chap 4). Différents éléments intervenant au cours de ces étapes ou au cours du processus d'évaluation seront discutés, tels que la projection des données issues de  $T$  vers l'espace de représentation des données de référence ou les critères de performance mesurés. Dans la suite de ce chapitre, plusieurs solutions de perception seront évaluées et les résultats obtenus à partir des évaluations pourront être étudiés de façon détaillée.

## 5.2 Processus d'évaluation

En parallèle de la production des données de référence, réalisée par l'étape « Référencement automatique des données », deux étapes sont appliquées. Il s'agit de l'« Application de la solution testée » et de la « Projection des données » (Fig 5.1). Ces étapes ont pour objectif de conditionner les données issues d'une solution de perception testée  $T$ , pour la dernière étape de l'approche et qui correspond à l'évaluation. Une fois ces deux étapes appliquées, les données issues de  $T$  peuvent être utilisées au cours de la dernière étape : l'« Évaluation de la solution de perception testée, et permettre de mesurer les performances de cette solution.

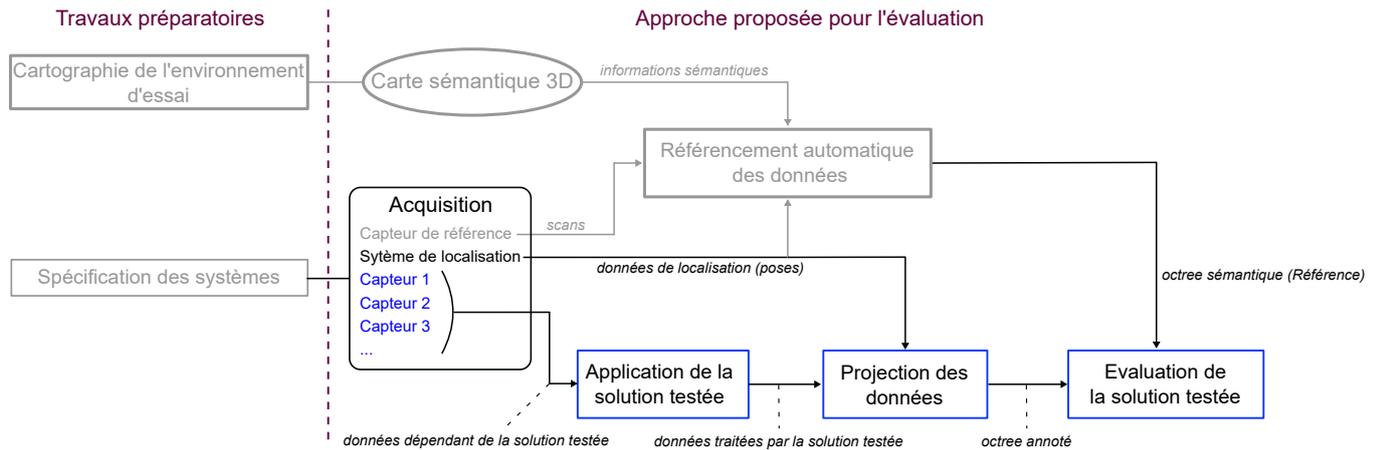


FIGURE 5.1 – Approche de validation des solutions de perception proposée : les étapes non concernées au cours de ce chapitre sont grisées.

### 5.2.1 Application de la solution testée

Comme introduite (Chap 3), l'étape « Application de solution testée » consiste à appliquer une solution de perception  $T$ , du type COTS (Commercial Off-The-Shelf) et que l'on souhaite évaluer dans des conditions identiques aux données acquises par  $L_{ref}$ . Pour ce faire, la couche algorithmique implémentée dans  $T$  est utilisée pour traiter les données produites par le capteur. Celles-ci sont acquises lors de l'étape « Acquisition de données ». Les données traitées et récupérées en sortie de l'étape « Application de solution testée » sont augmentées. En fonction du type de traitement associé à  $T$ , de nouvelles informations peuvent y être associées, telles que des annotations sémantiques. Plusieurs exemples de traitements, leurs rôles, ainsi que les données qu'ils produisent ont été introduits (Chap 2).

### 5.2.2 Projection des données

Les traitements réalisés sur les données produites par une solution testée  $T$  peuvent se baser sur des données exprimées dans le référentiel local du capteur associé à  $T$  (Chap 2), comme sur des données géoréférencées. Dans l'objectif d'utiliser un référentiel commun aux données de référence et aux données testées, une seconde étape de traitement est utilisée sur les données récupérées en sortie de  $T$ . Cette seconde étape : « Projection des données », est constituée de trois sous-étapes illustrées (Fig 5.2) et permet de répondre à deux objectifs. Le premier objectif est d'utiliser les paramètres de calibration extrinsèques ainsi que les mesures du système de localisation, introduits (Chap 2) et (Chap 3), pour calculer les transformations nécessaires et géoréférencer les données produites par  $T$  dans le même système de coordonnées de référence  $\mathcal{G}$ , également utilisé pour géoréférencer les données de référence issues de  $L_{ref}$ .

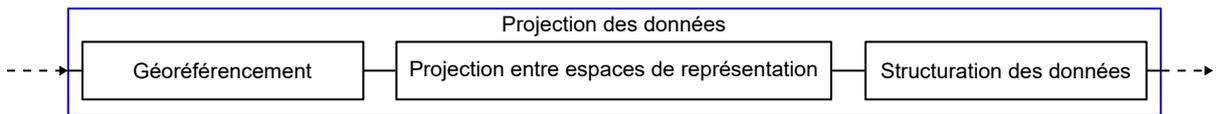


FIGURE 5.2 – Sous-étapes de la « Projection des données »

Le second objectif de cette étape est de projeter les données produites par d'autres modalités que le LiDAR, vers l'espace de représentation 3D, utilisé par nos données de référence et à l'inverse de ce qui est opéré dans les travaux [12]. Dans ces travaux, des données LiDAR sont également utilisées comme référence pour des solutions de perception basées sur une autre modalité : la vision. Ces données LiDAR sont alors projetées de la 3D vers la 2D, afin de correspondre à l'espace de représentation utilisé par les données de la solution testée  $T$ . L'un des rôles de notre approche est d'être suffisamment flexible pour évaluer des données associées à différentes modalités. La projection entre les espaces de représentation est alors opérée dans le sens inverse : ce sont les données produites par  $T$  qui sont projetées et adaptées à l'espace de représentation 3D utilisé par les données de référence.

Enfin, une dernière opération est appliquée pour homogénéiser les données évaluées avec les données de référence, en utilisant un moyen de représentation commun. Les données produites par  $T$  sont alors structurées en octree par le processus introduit (Chap 4). Pour chaque trame de données  $i$ , produite par  $T$  (scan, image, etc.), un octree  $O_{test,i}$  est créé.

Grâce à ces sous-étapes, les données issues de  $T$  utilisent un espace et un moyen de représentation identiques à celui des données de référence. Elles sont également géoréférencées dans le même système de coordonnées  $\mathcal{G}$ . Ces conditions permettent d'opérer la dernière étape de l'approche et qui correspond à l'évaluation de  $T$ .

### 5.2.3 Évaluation de la solution testée

Une fois les données issues de  $T$  pré-traitées par cette étape de projection, les octrees  $O_{test,i}$  qui en résultent peuvent être utilisés pour évaluer les performances de  $T$ . Au cours de l'étape « Évaluation de la solution testée », la comparaison des octrees est opérée entre les voxels homologues de  $O_{test,i}$  et  $O_{ref}$ . Les comparaisons voxel à voxel sont réalisées en se basant sur les coordonnées  $(X, Y, Z)$  de leurs nœuds respectifs et en fonction de leur niveau de profondeur  $d$  dans l'arborescence.

À l'issue de ces comparaisons, des premiers indicateurs peuvent être calculés. Ils peuvent soit correspondre aux critères de performance mesurés, soit être utilisés à posteriori, pour déterminer la valeur des métriques de performance souhaitées et détaillées (Sec 5.2.4). Pour être mesurés, ces indicateurs se basent sur l'annotation sémantique respective de chaque voxel. Un voxel  $V_{test} \in O_{test,i}$  sera alors considéré comme correct si la valeur de classe sémantique associée à son attribut  $label$  est cohérente avec celle de son voxel homologue  $V_{ref} \in O_{ref}$ .

Les solutions évaluées au cours de ce chapitre, introduites (Sec 5.3), ne sont pas multi-classes et ciblent une classe sémantique particulière. Dans ces conditions, lors des comparaisons entre les voxels, on retrouve quatre cas possibles qui correspondent à la composition des matrices de confusions : le vrai positif ( $VP$ ), le vrai négatif ( $VN$ ), le faux positif ( $FP$ ) et le faux négatif ( $FN$ ), représentés sur la (Fig 5.3).

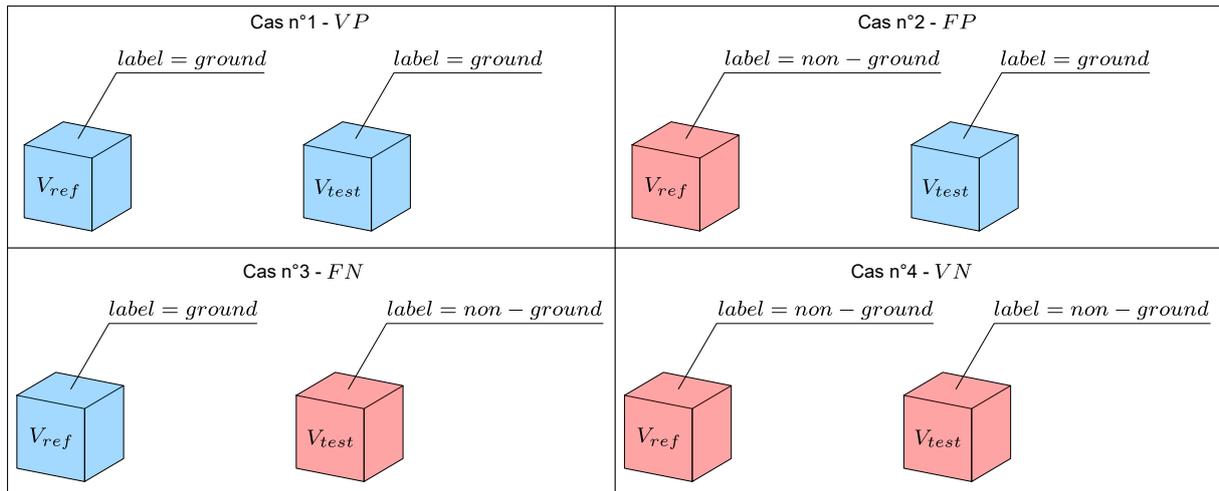


FIGURE 5.3 – Représentation des quatre cas de matrice de confusion lors des comparaisons entre voxels : l'exemple représenté ici, correspond à une solution dont l'algorithme opère une classification binaire des données. La classe sémantique ciblée par cette classification est le sol (« ground ») et les données annotées sont réparties en fonction des deux classes : ground et non-ground.

### 5.2.4 Métriques d'évaluation quantitative

Grâce aux indicateurs calculés à partir des comparaisons entre les voxels  $O_{ref}$  et  $O_{test,i}$ , des métriques d'évaluation connues peuvent être mesurées pour caractériser les performances des solutions testées  $T_i$ . Ces critères peuvent être adaptés en fonction du type d'algorithme qui compose la solution évaluée. Dans la littérature scientifique, on peut constater que plusieurs métriques sont utilisées pour évaluer les différents types d'algorithmes de manière la plus pertinente.

Pour les algorithmes de détection, deux cas peuvent être évoqués. Dans le premier cas, l'algorithme renvoie des informations limitées sur les détections (par exemple avec un algorithme de détection de lignes qui renvoie un polynôme pour chaque ligne détectée). Une métrique d'évaluation, adaptée à ce cas, est la "justesse de classe". On la retrouve dans les travaux [12, 126] et elle représente le taux de bonnes détections. Elle est notée  $CA^1$  et calculée par (Eq 5.1).

$$CA = \frac{N_{pred}}{N_{ref}} \quad (5.1)$$

où  $N_{pred}$  est le nombre de voxels correctement prédits (détectés ou classés) par la solution évaluée et  $N_{ref}$  est le nombre de voxels de référence pour cette même classe.

Dans le cas où l'algorithme de détection transmet des boîtes englobantes représentant les objets détectés, comme pour la partie détection d'objets du benchmark KITTI [4], les performances peuvent être mesurées à l'aide de la métrique  $IoU^2$  calculée par (Eq 5.2). Cette métrique est également la plus couramment utilisée pour évaluer les algorithmes de segmentation sémantique, basés sur des données 2D [14], comme sur des données 3D [67].

$$IoU = \frac{VP}{VP + FP + FN} \quad (5.2)$$

Enfin, pour les algorithmes de classification, la métrique privilégiée est le  $F_{score}$  comme dans [127, 128]. Il s'agit de la moyenne harmonique de deux autres métriques : la précision et le rappel.

1.  $CA$  pour Class Accuracy  
2. Intersection over Union

La précision de l'algorithme ( $PRE$ ) calculée par (Eq 5.3) correspond au ratio de prédictions vraies positives  $VP$  parmi tous les exemples étiquetés positivement. Le rappel ( $REC$ ) calculé par (Eq 5.3) représente le pourcentage de  $VP$  que l'algorithme a pu classer correctement.

$$REC = \frac{VP}{VP + FN} ; PRE = \frac{VP}{VP + FP} \quad (5.3)$$

$$F_{score} = 2 \cdot \frac{PRE \cdot REC}{PRE + REC} \quad (5.4)$$

Bien que ces métriques présentées soient les plus couramment utilisées. D'autres métriques peuvent également être calculées pour compléter l'évaluation des performances et analyser les algorithmes de manière plus détaillée. Par exemple, pour certains algorithmes évalués sur KITTI [4], on peut retrouver, parmi les métriques utilisées : la précision moyenne, le taux de faux positifs, ou encore, le taux de faux négatifs.

### 5.2.5 Évaluation qualitative

Les premiers indicateurs introduits précédemment et représentés (Fig 5.3) peuvent également être utilisés dans le cadre d'une évaluation qualitative. Cela passe par l'attribution d'une colorisation spécifique des données traitées, en fonction des différents indicateurs obtenus à l'issue des comparaisons. Cette évaluation qualitative permet d'analyser de façon ponctuelle une scène traitée par une solution  $T$  et d'émettre des hypothèses d'amélioration sur la base de ces résultats. On peut en voir une représentation (Fig 5.4), où une solution d'extraction du sol est évaluée qualitativement.

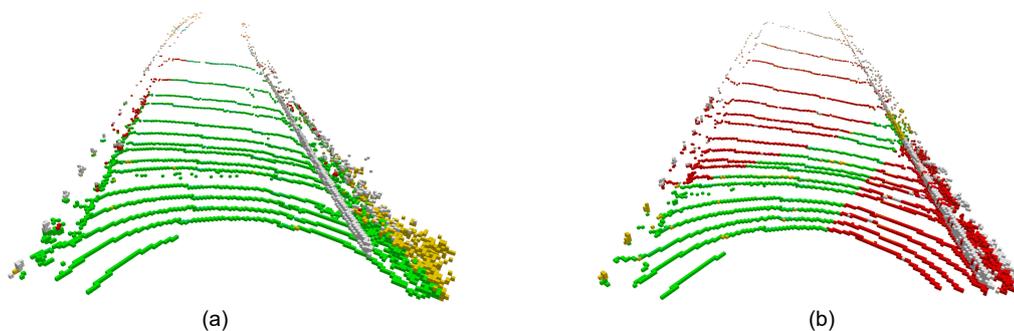


FIGURE 5.4 – Évaluation qualitative d'une solution  $T$  : Il s'agit d'une solution destinée à l'extraction du sol avec les indicateurs  $VP$  (en vert),  $FP$  (en jaune),  $FN$  (en rouge) et  $VN$  (en blanc). On peut alors comparer les résultats qualitatifs de  $T$  à deux moments du jeu de données : (a) Un moment où  $T$  a un haut niveau de performance ; (b) Un moment où  $T$  a un faible niveau de performance.

A titre d'exemple, après avoir réalisé l'évaluation quantitative, la valeur d'une métrique de performance peut être représentée sous la forme d'une courbe, ou en fonction de la trajectoire de la plateforme. Cela permet d'identifier temporellement les moments du jeu de données où les performances de la solution testée sont les plus élevées, ou à l'inverse les plus faibles. L'évaluation qualitative peut alors être utilisée sur ces moments spécifiques pour comprendre pourquoi on observe une hausse ou une baisse des performances, sur la courbe de performance associée. Une analyse du comportement d'une solution, en se basant uniquement sur la courbe d'un critère de performance, peut être insuffisante. Avoir un aperçu graphique du phénomène, tel qu'obtenu avec l'évaluation qualitative, peut alors simplifier la compréhension de ce comportement et permettre de cibler les améliorations à apporter en fonction des constats réalisés.

## 5.3 Évaluation de solutions de perception

Dans la suite de ce chapitre, différentes solutions de perception sont évaluées par notre approche. Elles sont dans un premier temps décrites afin d'introduire leurs spécificités et les types auxquels elles appartiennent. Dans un second temps, la méthode d'évaluation est appliquée sur ces solutions et les résultats obtenus sont analysés.

### 5.3.1 Solutions de perception évaluées

Pour expérimenter notre approche en conditions d'usage réelles, celle-ci a été utilisée pour évaluer trois solutions de perception différentes.

#### 5.3.1.1 Configuration de capteurs et jeu de données utilisé

Ces trois solutions testées, nommées  $T_1$ ,  $T_2$  et  $T_3$ , se basent sur les différents capteurs présents sur la plateforme équipée et introduite (Chap 3). Ils sont représentés (Fig 5.5). Chaque solution évaluée  $T_i$  utilise un capteur indépendant du capteur  $L_{ref}$ , afin d'éviter les biais d'évaluation pouvant apparaître lorsque le capteur utilisé comme capteur de référence est également celui utilisé par la solution de perception testée.

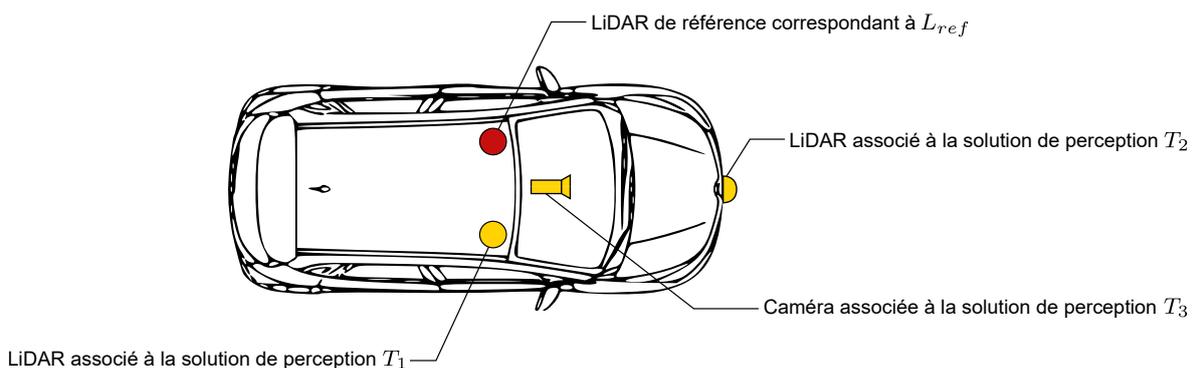


FIGURE 5.5 – Configuration sensorielle de la plateforme MELO : Représentation de la position des différents capteurs utilisés au cours des évaluations. Le capteur rouge représente  $L_{ref}$ . Les capteurs représentés en jaune correspondent à ceux utilisés par les différentes solutions de perception évaluées  $T_1$ ,  $T_2$  et  $T_3$ .

Parmi ces capteurs (Fig 5.5), on retrouve deux modalités. Une première modalité correspond aux données des LiDARs utilisées pour les solutions  $T_1$  et  $T_2$ . Le LiDAR associé à  $T_1$  est intégré à la même hauteur que le capteur  $L_{ref}$ , sur le toit du véhicule. Le second LiDAR, correspondant à  $T_2$ , est quant à lui intégré dans le pare-chocs avant de la plateforme. La seconde modalité se base sur la vision avec une caméra associée à la solution  $T_3$ , intégrée derrière le pare-brise de la plateforme.

Pour évaluer les différentes solutions de perception  $T_i$ , le jeu de données  $\mathcal{D}_2$  introduit (Chap 3) est de nouveau utilisé. L'acquisition des données globale qui constitue ce jeu de données correspond à un tracé de 2,3 km de longueur. Pour garder une indépendance vis-à-vis des tests opérés au cours du (Chap 4), l'échantillon de données utilisés pour ces premiers tests et représentant 600 m du parcours, est retiré des évaluations menées dans ce chapitre (Fig 5.6). Le reste de l'acquisition représente un parcours de 1,7 km et nous a permis de mener des évaluations sur 824 scans obtenus à partir du LiDAR associé à  $T_1$ , 829 scans pour le LiDAR associé à  $T_2$  et 2962 images produites par  $T_3$ .

Comme pour la méthode de référencement étudiée (Chap 4), la bonne évaluation des solutions de perception dépend en partie des données fournies par le système de localisation et du géoréférencement des voxels utilisés (Sec 5.2.3). Pour garantir des évaluations en utilisant des mesures de positionnement de bonne précision, les écarts-types de ces mesures  $\sigma_{pos}$ , fournis par le système de localisation, ont été étudiés sur l'ensemble de l'acquisition. Ces écarts-types sont représentés (Fig 5.6). Concernant le positionnement, on observe que ces écarts-types se situent dans l'intervalle  $\sigma_{pos} = [3.4 \text{ cm}; 5.2 \text{ cm}]$ .

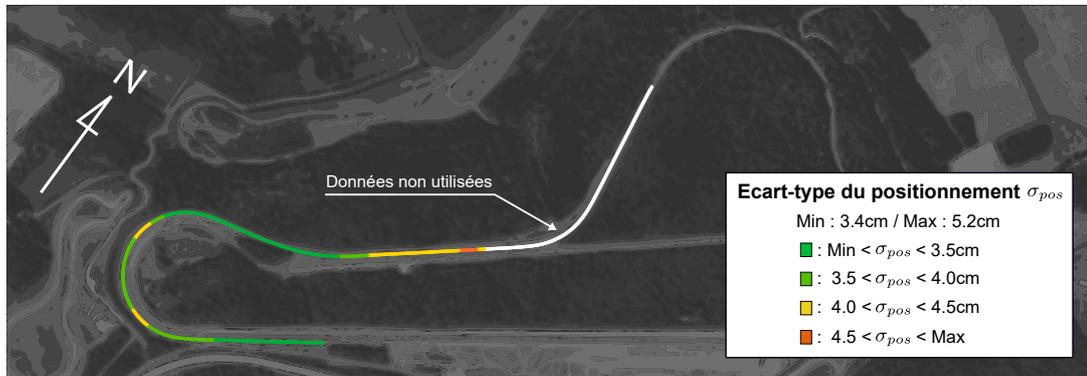


FIGURE 5.6 – Trajectoire suivie par la plateforme : Tracé de la trajectoire, suivie par la plateforme au cours de l'étape « Acquisition de données » et correspondant au jeu de données utilisé. Les couleurs sont utilisées pour représenter l'évolution de l'écart-type des mesures de positionnement au cours de cette acquisition. La partie blanche représente l'échantillon de données utilisé dans le (Chap 4) et n'est pas exploité au cours des évaluations réalisées dans ce chapitre.

### 5.3.1.2 Extraction du sol basée sur des données LiDAR

Les deux premières solutions de perception évaluées  $T_1$  et  $T_2$  utilisent donc les LiDARs représentés (Fig 5.5). Les algorithmes de traitement, auxquelles elles sont associées, correspondent respectivement aux travaux présentés dans [129] et [130]. Ces algorithmes traitent les scans et utilisent une classification binaire (Chap 2). Les points composant les scans traités sont séparés en deux classes sémantiques : sol/non-sol. Les données traitées par les algorithmes de ces solutions sont représentées (Fig 5.10). Ces nuages, récupérés en sorties de ces solutions, sont ensuite structurés sous forme d'octrees étiquetés et peuvent ensuite être comparés à l'octree de référence  $O_{ref}$ .

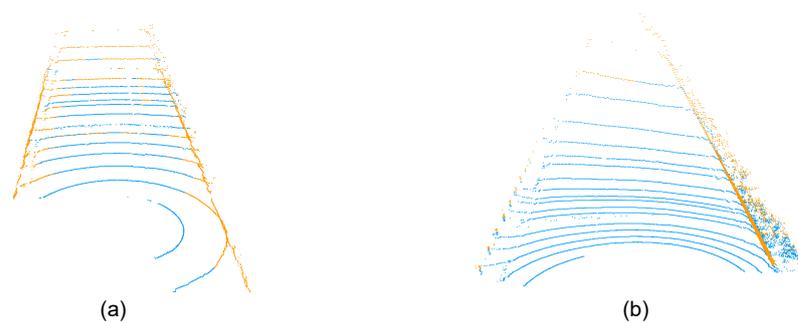


FIGURE 5.7 – Données obtenues dans le cadre de l'extraction du sol : Représentation de deux scans traités et obtenus en sortie des solutions  $T_1$  (a) et  $T_2$  (b). Les points bleus correspondent aux points classés comme « sol », les points oranges sont associés à la classe « non-sol ».

### 5.3.1.3 Détection des marquages basée caméra

La troisième solution de perception évaluée  $T_3$  correspond à une solution COTS se basant sur une caméra monoculaire dont la position est représentée (Fig 5.5). La solution  $T_3$  correspond au brevet [131] et intègre un algorithme de détection ciblé sur les lignes de marquages au sol.

Chaque image produite par la caméra est traitée de façon interne à la solution, sans possibilité d'accéder aux données brutes ou à la méthode de traitement. Les données récupérées en sortie correspondent à une liste de coefficients correspondants à plusieurs polynômes du troisième degré. Pour chaque ligne détectée dans l'image, les coefficients du polynôme s'adaptant le mieux à cette ligne sont ainsi donnés. A ces informations s'ajoutent la portée maximale ( $P_{max}$ ) sur laquelle ces polynômes peuvent être appliqués.

La solution  $T_3$  se base sur une modalité différente de celle du LiDAR  $L_{ref}$ , utilisé comme référence. Les données obtenues en sortie de cette solution sont exprimées en 2D, dans le plan  $XY$  (Fig 5.8). Comme indiqué dans la (Sec 5.2.2) l'étape "Projection des données" (Fig 5.1) de l'approche intégrée, pour cette solution, une projection des données vers une représentation 3D. Les lignes détectées sont d'abord représentées par des points 2D calculés dans le référentiel de la caméra associée à  $T_3$ . En utilisant les polynômes donnés en sorties de  $T_3$ , les points successifs d'une ligne représentés (Fig 5.8), sont calculés à l'aide de (Eq 5.5), pour des valeurs de  $x$  comprises dans l'intervalle  $[0.00; P_{max}]$  et avec un pas de 10 cm.

$$y = c_3 \cdot x^3 + c_2 \cdot x^2 + c_1 \cdot x + c_0 \quad (5.5)$$

où  $c_i$  représentent les coefficients du polynôme d'une ligne détectée dans l'image traitée et le couple  $(x, y)$ , les coordonnées 2D des points calculés en fonction des coefficients du polynôme.

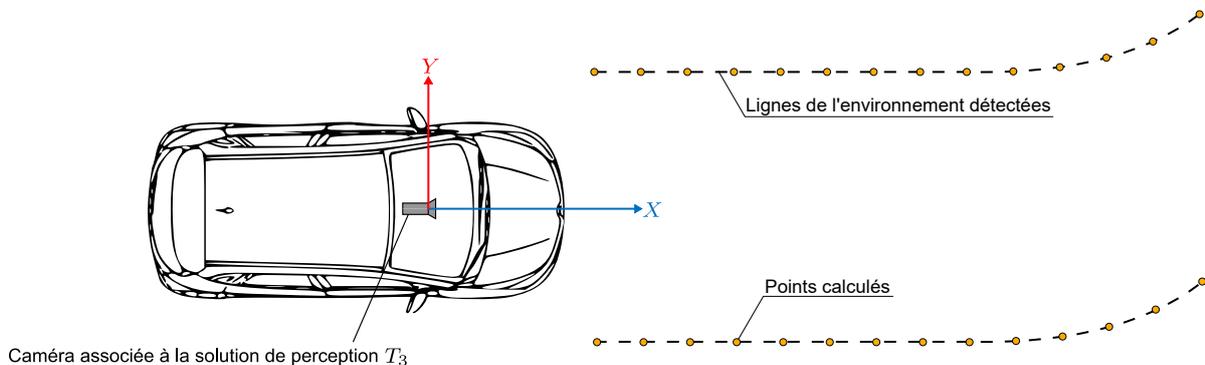


FIGURE 5.8 – Calcul des points associés aux lignes détectées : Représentation des axes de la caméra de la solution  $T_3$  et des points calculés (en jaune) à partir des coefficients polynomiaux associés aux différentes lignes détectées par cette solution.

En utilisant les paramètres de calibration extrinsèques de la caméra, définis lors de l'étape de calibration détaillée (Chap 3), les points calculés sont projetés en 3 dimensions, en utilisant notamment la hauteur de la caméra par rapport au sol ainsi que l'orientation de ses axes vis-à-vis du repère  $Ego$ . Cette projection est effectuée sous l'hypothèse d'un sol modélisable par un plan et est appliquée sur la portée  $P_{max}$  transmises.  $P_{max}$  ayant une valeur moyenne proche de 60 m (Fig 5.9), cette hypothèse est raisonnablement valable. La transformation rigide calculée lors de la calibration permet d'exprimer les coordonnées des points calculés dans le repère  $Ego$ . Ces données sont ensuite géoréférencées dans le référentiel  $\mathcal{G}$ , à l'aide des poses  $Pose_i$  associées à chaque image produite par  $T_3$ .

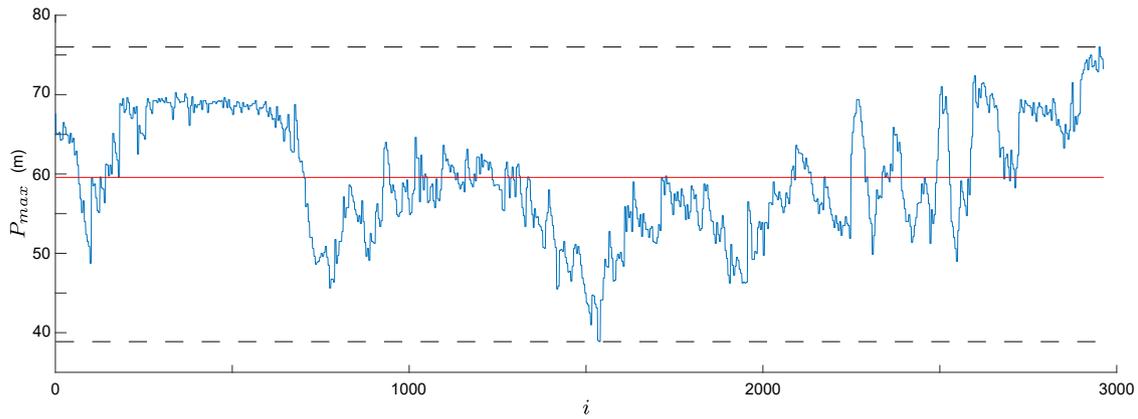


FIGURE 5.9 – Valeurs de  $P_{max}$  au cours du jeu de données : Courbe représentant les valeurs atteintes par  $P_{max}$  au cours de l'acquisition en fonction de  $i$  : l'indice de l'image traitée. Les valeurs de  $P_{max}$  sont comprises dans l'intervalle  $[min; max] = [38.87 \text{ m}; 76.00 \text{ m}]$  représenté par les tirets noirs. Sa valeur moyenne (en rouge) est de 59.57 m.

### 5.3.2 Évaluation des solutions de perception basées LiDAR

Les deux premières solutions  $T_1$  et  $T_2$ , introduites (Sec 5.3.1), reposent sur l'utilisation de LiDARs, associés à des algorithmes d'extraction du sol. Ces derniers peuvent être assimilés à des algorithmes de classification binaires, classant les points selon deux classes : sol et non-sol. Les performances de ces solutions peuvent être qualifiées en s'appuyant sur les métriques  $PRE$ ,  $REC$  et  $F_{score}$ . Les différentes étapes introduites au cours de ce chapitre et permettant l'évaluation sont représentées (Fig 5.10).

Les évaluations qualitatives représentées (Fig 5.10) permettent de voir différents moments de la séquence étudiée et différents niveaux de performance atteints par les solutions  $T_1$  et  $T_2$ . Elles permettent également de vérifier visuellement que l'approche mise en place évalue correctement et de façon cohérente ces solutions.

Ces deux solutions ont été évaluées sur quatre niveaux de profondeur d'octree : 0.10m, 0.20m, 0.40m et 0.80m. Les résolutions des niveaux supérieurs étant trop faibles, elles ne sont pas assez pertinentes pour évaluer les algorithmes. Les métriques calculées ont été moyennées sur les 824 scans traités par  $T_1$  et les 829 scans traités par  $T_2$  et sont résumées dans le (Tab 5.1).

| Résolution $r$ | $T_1$ [130] |         |             | $T_2$ [129] |         |             |
|----------------|-------------|---------|-------------|-------------|---------|-------------|
|                | $PRE$       | $REC$   | $F_{score}$ | $PRE$       | $REC$   | $F_{score}$ |
| 0.10m          | 78.23 %     | 54.53 % | 0.63        | 79.60 %     | 78.73 % | 0.78        |
| 0.20m          | 82.42 %     | 48.28 % | 0.59        | 93.60 %     | 72.64 % | 0.79        |
| 0.40m          | 96.47 %     | 37.67 % | 0.53        | 99.54 %     | 68.23 % | 0.78        |
| 0.80m          | 99.18 %     | 38.45 % | 0.54        | 99.98 %     | 65.75 % | 0.76        |

TABLE 5.1 – Résultats des solutions d'extraction du sol : en fonction des métriques  $PRE$ ,  $REC$  données en pourcentages et  $F_{score} [0; 1]$ . Les valeurs présentées correspondent aux moyennes de ces métriques, calculées sur l'ensemble de la séquence traitée.

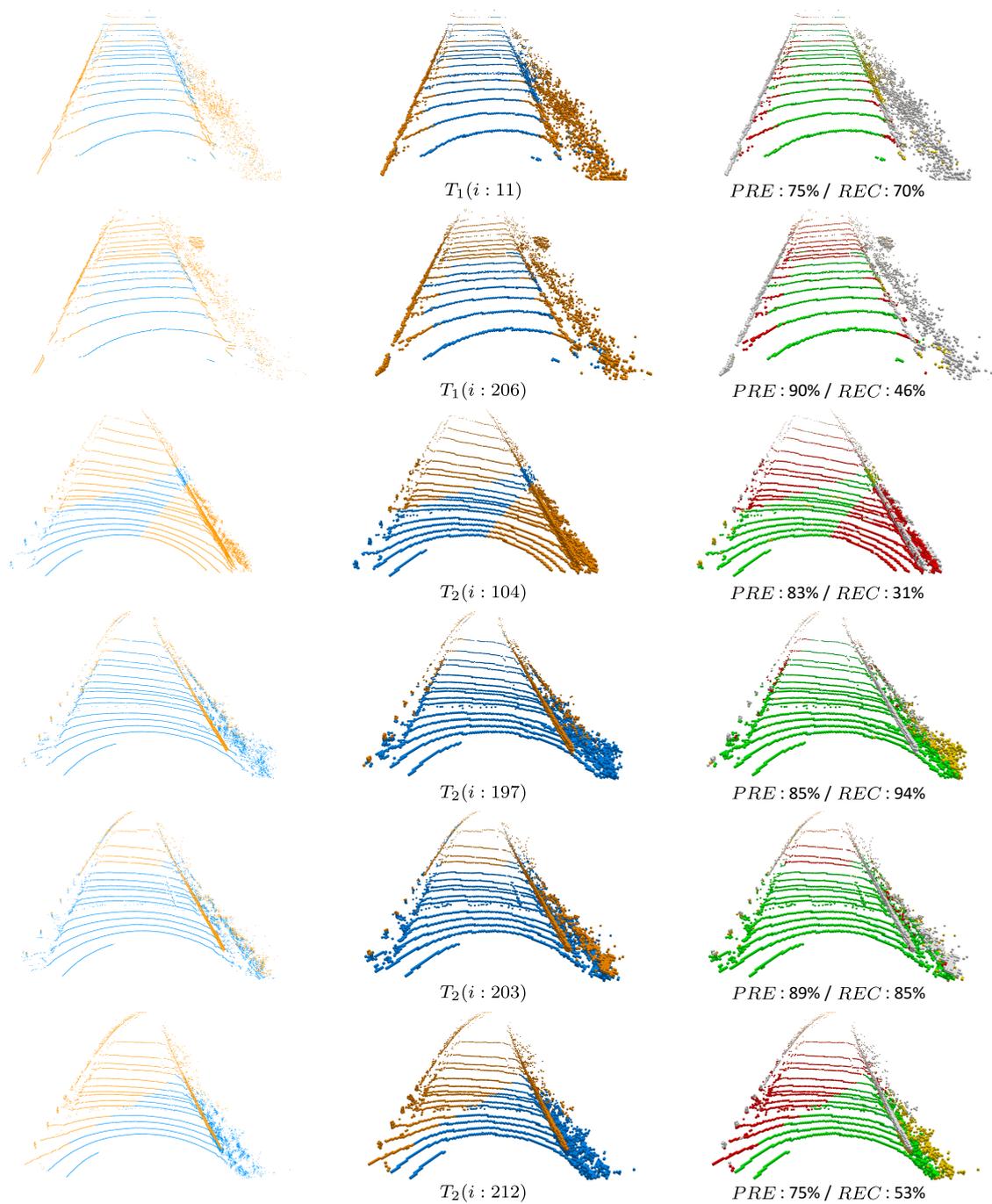


FIGURE 5.10 – Évaluation des performances des solutions d'extraction du sol : La colonne de gauche correspond aux scans traités récupérés en sortie des solutions  $T_1$  et  $T_2$ . Les points bleus ont été classés « sol », les points classés « non-sol » sont représentés en orange. La colonne du milieu représente ces mêmes scans après structuration sous forme d'octree  $o_{test,i}$ . Enfin, la dernière colonne représente les indicateurs obtenus au cours de la comparaison avec l'octree  $O_{ref}$ . Les voxels verts correspondent au cas  $VP$ , les rouges au cas  $FN$ , les jaunes au cas  $FP$  et les blancs au cas  $VN$ .

Nous avons pu voir dans le (Chap 3) et le (Chap 4) que l'utilisation des octrees nécessite de trouver un compromis entre résolution de l'octree et détails de la scène. Plus la résolution est importante, plus les détails de la scène seront conservés, mais moins l'optimisation des données sera efficace et moins le temps de traitement lors de la structuration des données sera rapide. Mener les évaluations à différentes échelles nous a permis d'analyser l'impact du choix de la résolution utilisée lors de l'évaluation d'une solution. Pour  $T_1$  et  $T_2$ , on peut voir (Tab 5.1) que cet impact sur le  $F_{score}$  est relativement faible. Ces solutions pourraient alors être compatible avec des évaluations à des résolutions inférieures à 10 cm, permettant un gain de temps sur le processus d'évaluation de ces solutions.

Si l'on souhaite mettre en comparaison les deux solutions, il est possible de s'appuyer sur les résultats synthétisés dans (Tab 5.1) dans un premier temps. Pour approfondir l'étude de ces solutions, une autre possibilité est de tracer l'évolution de leur métrique  $F_{score}$ , au cours de la séquence traitée et de comparer la courbe propre à chaque solution (Fig 5.11).

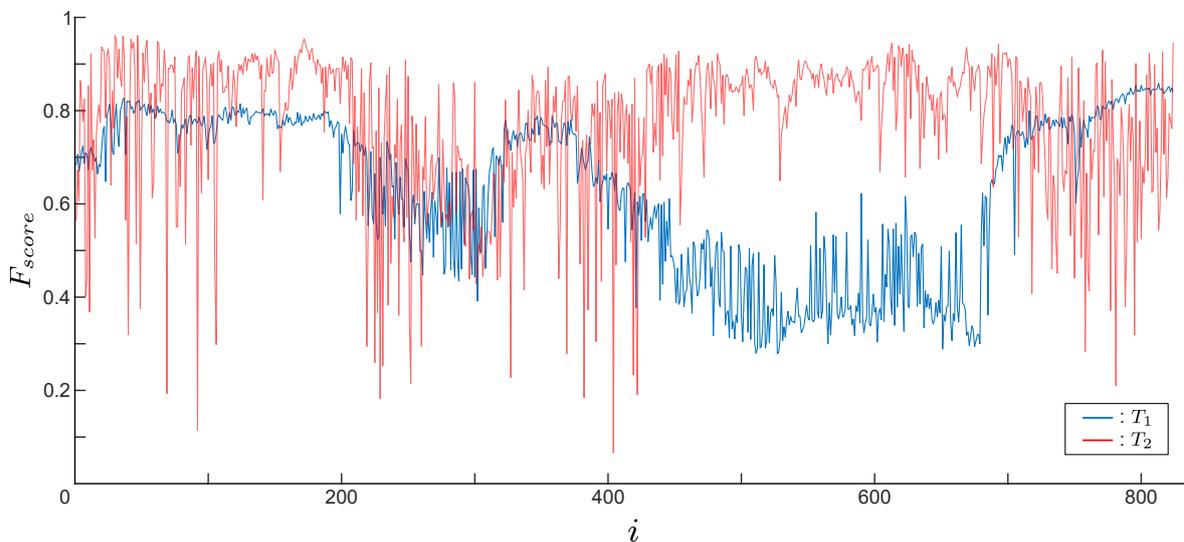


FIGURE 5.11 – Courbes du  $F_{score}$  au cours de la séquence étudiée pour  $r = 10\text{ cm}$  : le courbe bleue représente le  $F_{score}$  atteint par la solution  $T_1$  [130]. La courbe rouge correspond quant à elle à la solution  $T_2$  [129].

L'analyse de ces courbes permet de voir que la solution  $T_2$  a, de façon globale, de meilleures performances. Elle est cependant impactée par de plus grandes variations (identifiées par les pics rouges) sur de courts laps de temps. A l'inverse la solution  $T_1$ , dont les taux de performances globaux sont moins bons, a des variations moins marquées à court termes et semble plus stable. On note cependant qu'à deux moments de la séquence étudiée, correspondant approximativement aux intervalles  $i = [200 ; 350]$  et  $i = [400 ; 700]$ , les performances baissent de façon conséquente. Pour comprendre ce phénomène nous avons produit une autre représentation (Fig 5.12).

En représentant qualitativement la valeur du  $F_{score}$  en fonction de la trajectoire suivie par la plateforme, on obtient les graphiques représentés (Fig 5.12). Ils permettent de vérifier la cohérence entre les résultats présentés (Tab 5.1) et (Fig 5.11). Ils peuvent ainsi permettre de mieux comprendre le comportement des deux solutions étudiées en représentant de façon plus compréhensive l'évolution du  $F_{score}$  et en donnant la possibilité de le lier à des phénomènes physiques extérieurs.

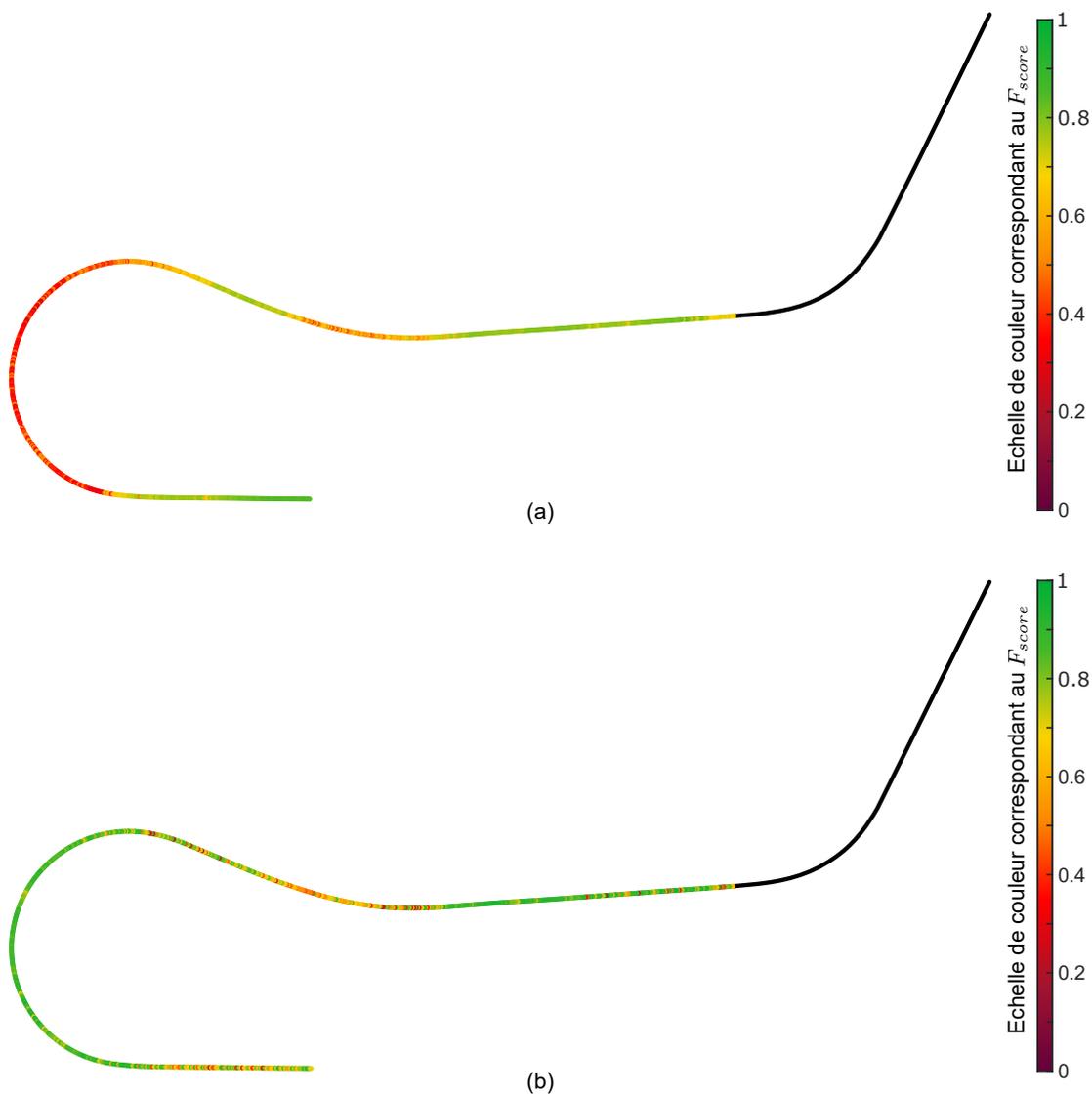


FIGURE 5.12 –  $F_{score}$  en fonction de la trajectoire d'acquisition : Représentation de la métrique  $F_{score}$  en fonction de la position de la solution sur la trajectoire ainsi que d'une échelle de couleur associée à la valeur de cette métrique. Les valeurs  $F_{score}$  sont représentées pour la résolution la plus fine prise en compte avec : (a) La solution  $T_1$  basée sur l'algorithme [130]; (b) la solution de perception  $T_2$  utilisant l'algorithme proposé dans [129]. Le début de courbe (en noir) représente les données non prises en compte dans l'étude.

Grâce aux courbes (Fig 5.11), nous avons pu identifier deux intervalles d'indice  $i$  pour lesquels les performances de la solution  $T_1$  diminuent. En observant la figure (Fig 5.12-a), il est possible d'identifier visuellement ces deux intervalles et de les associer aux moments où la trajectoire n'est plus linéaire, elle est perturbée par des virages. Une première hypothèse est que les perturbations liées aux virages provoquent des mouvements de caisse de la plateforme. Ceux-ci se propagent sur l'orientation du capteur associé à  $T_1$  et impactent les performances de l'algorithme de traitement associé. L'étude peut alors être ciblée sur ces perturbations.

En superposant sur les mêmes graphiques le courbe du  $F_{score}$  de  $T_1$  et celles des angles de roulis et de tangage, on peut confirmer l'hypothèse émise à partir de représentation (Fig 5.12). En effet, les courbes représentées (Fig 5.13) et (Fig 5.14) permettent de lier les variations de l'orientation du capteur avec les variations du  $F_{score}$ . Une des faiblesses de la solution  $T_1$  réside donc dans la sensibilité de son algorithme de traitement aux variations d'orientation du capteur.

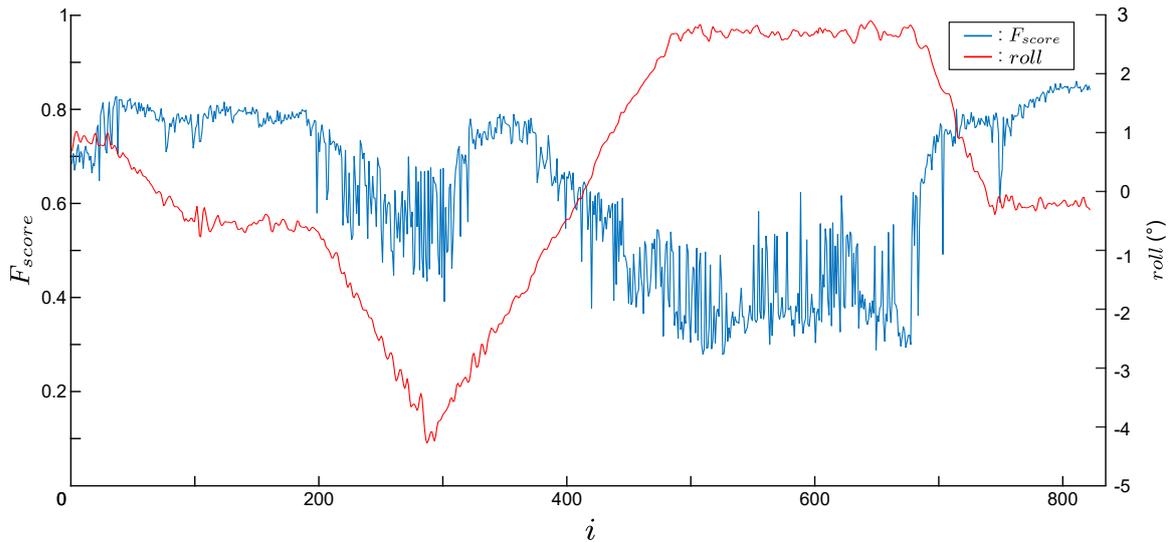


FIGURE 5.13 – Évolution du  $F_{score}$  et du roulis au cours de la séquence étudiée : Superposition de la courbe du  $F_{score}$  (en bleu) et de la courbe du roulis ( $roll$ ) (en rouge).

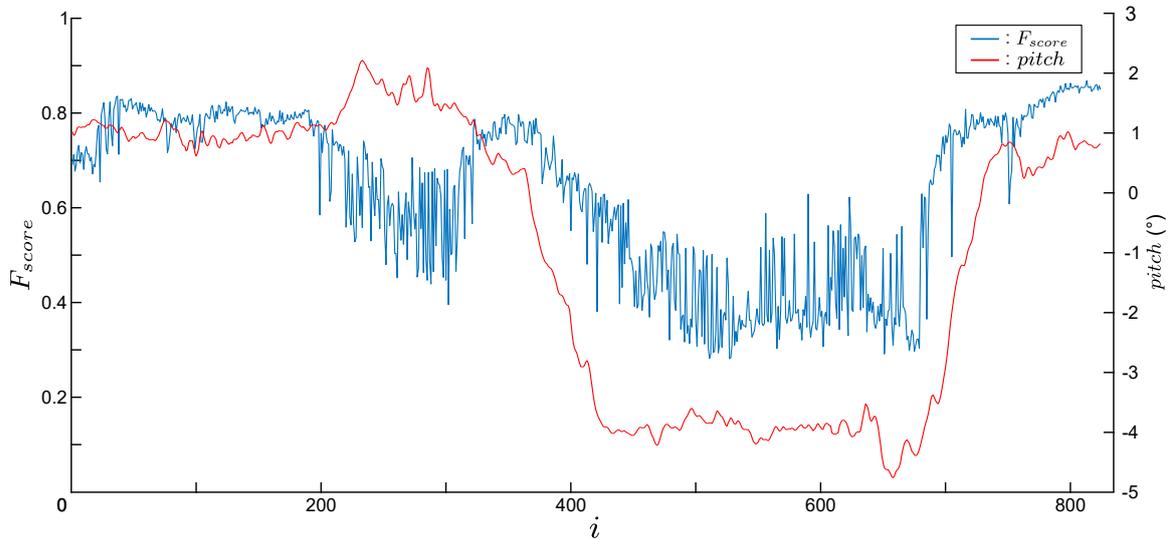


FIGURE 5.14 – Évolution du  $F_{score}$  et du tangage au cours de la séquence étudiée : Superposition de la courbe du  $F_{score}$  (en bleu) et de la courbe du tangage ( $pitch$ ) (en rouge).

Avec le graphique (Fig 5.12-b) on peut noter que la solution  $T_2$  n'est pas aussi impactée par ces variations d'orientation que  $T_1$ . Au-delà de l'algorithme de traitement en lui-même, cette différence de performance et de sensibilité entre les deux solutions peut être justifiée par un autre aspect. Il s'agit de la position des capteurs par rapport au sol. En effet, comme on peut le voir (Fig 5.15) et (Fig 5.16), plus un capteur sera positionné en hauteur par rapport au sol, plus les impacts du roulis et du tangage seront importants sur le capteur.

Le LiDAR associé à la solution  $T_1$ , est positionné sur le toit. Le LiDAR associé à  $T_2$  est quant à lui intégré dans le pare-chocs avant de la plateforme. Le capteur associé à  $T_2$  est donc beaucoup plus proche du sol que celui associé à  $T_1$ . Par conséquent, les effets du roulis et du tangage ont un impact beaucoup plus important sur la solution  $T_1$  que sur  $T_2$ . Ce phénomène permet ainsi de justifier et de comprendre la différence de performance entre les solutions et amène des pistes d'amélioration de la solution  $T_1$ , notamment du point de vue de la sensibilité par rapport à l'orientation du capteur.

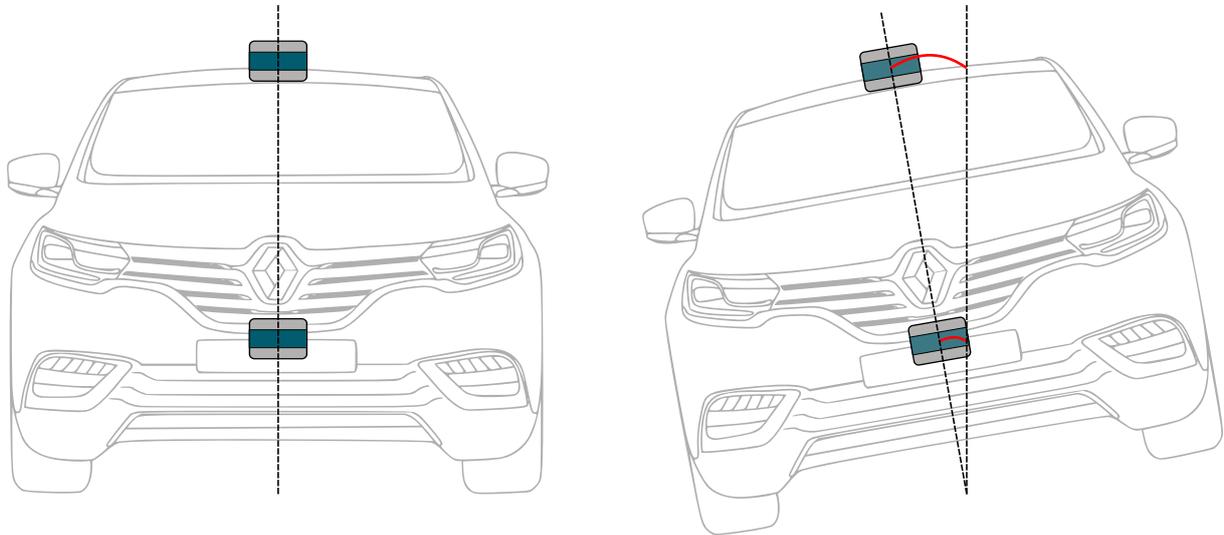


FIGURE 5.15 – Impact du roulis en fonction de la position du capteur : Représentation de l'impact d'un même angle de roulis pour deux capteurs positionnés à deux endroits différents de la plateforme avec : à gauche, une représentation où le roulis est nul ; à droite, une représentation où l'angle de roulis est de  $10^\circ$ . On voit que plus un capteurs est positionné haut, plus l'impact du roulis (en rouge) est important.

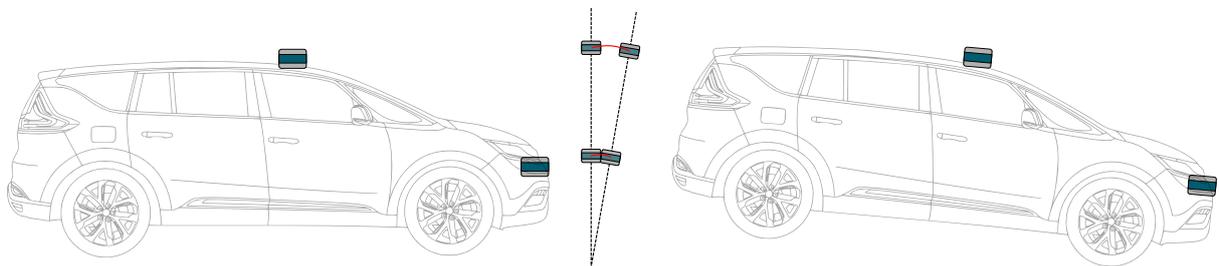


FIGURE 5.16 – Impact du tangage en fonction de la position du capteur : Représentation de l'impact d'un même angle de tangage pour deux capteurs positionnés à deux endroits différents de la plateforme avec : à gauche une situation où le tangage est nul ; à droite, une situation où le tangage est de  $10^\circ$  ; au centre, la projection des capteurs sur un même axe vertical, pour les deux situations et où l'impact du tangage est représenté en rouge.

### 5.3.3 Évaluation de la solution de perception basée Caméra

Comme nous l'avons introduit (Sec 5.3.1.3), la troisième solution de perception  $T_3$  se base sur la vision pour détecter les lignes des marquages au sol. Les données obtenues étant initialement des polynômes en 2 dimensions, l'évaluation de cette solution nécessite d'appliquer l'étape de projection pour obtenir des courbes composées de points en 3 dimensions et correspondant aux lignes détectées. Une dernière étape est de structurer ces points 3D en octrees pour pouvoir les comparer à l'octree de référence  $O_{ref}$ .

Une adaptation est cependant nécessaire au cours de l'étape d'évaluation pour mesurer correctement les performances de cette solution  $T_3$ . En effet, les polynômes récupérés en sortie de  $T_3$  permettent de générer des courbes 3D continues, représentant les marquages. Cependant, comme on peut le voir (Fig 5.17), l'environnement d'essai et les données de référence sont composés de marquages continus et discontinus. Pour ne pas biaiser les résultats de cette évaluation, la comparaison des données est faite dans le sens inverse : des voxels testés vers les voxels de référence. Pour évaluer cette solution, nous nous sommes basés sur la métrique  $CA$  (Sec 5.2.4) qui est adaptée à la représentation géométrique des données en sortie de  $T_3$ , ainsi qu'aux informations limitées qu'elle fournit. Pour chaque  $o_{test,i}$  testé le processus utilisé est le suivant :

1. Les octrees  $O_{ref}$  et  $o_{test,i}$  sont projetés dans le même espace 3D ;
2. Les voxels de référence ( $\epsilon O_{ref}$ ) sont filtrés en fonction de la portée  $P_{max}$  des lignes détectées par  $T_3$ . Le nombre de voxels  $V_{O_{ref}}$  situés dans cette zone filtrée sont comptabilisés et représentent la valeur  $N_{ref}$  dans le calcul de la métrique  $CA$  ;
3. Pour chacun des voxels de  $O_{ref}$  comptabilisé, on recherche au sein de l'octree  $o_{test,i}$ , le voxel  $V_{o_{test,i}}$  à la même position. Si ce voxel est occupé, il appartient bien à une ligne et est comptabilisé dans la valeur  $N_{pred}$ . A l'inverse, si le voxel est  $V_{o_{test,i}}$  n'est pas occupé, il n'est pas comptabilisé, ce qui ne pénalisera pas la valeur de  $CA$ .

Ce procédé nous permet de ne pas pénaliser la solution  $T_3$  en comptant les voxels associés aux interlignes comme des fausses détections. Des exemples de résultats de ce procédé avec différents échantillons sont représentés (Fig 5.17).

L'évaluation de  $T_3$  a été menée sur les mêmes résolutions d'octree que pour les solutions  $T_1$  et  $T_2$  : 10 cm, 20 cm, 40 cm et 80 cm. Les résultats moyens obtenus à partir des 2962 images traitées sont résumés dans (Tab 5.2).

| Résolution $r$ | $T_3[131]$ |
|----------------|------------|
|                | $CA$       |
| 0.10m          | 44.83 %    |
| 0.20m          | 62.35 %    |
| 0.40m          | 73.23 %    |
| 0.80m          | 77.56 %    |

TABLE 5.2 – Résultats mesurés pour la solution de perception  $T_3$  : Les valeurs de  $CA$  présentées correspondent aux moyennes de cette métrique, calculées sur l'ensemble de la séquence traitée.

Pour approfondir l'analyse de ces résultats, trois hypothèses sont discutées dans la suite de cette section et concernent différents aspects de cette solution avec : l'estimation de  $P_{max}$ , les limitations paramétriques internes à  $T_3$  et l'utilisation de l'hypothèse de sol plan.

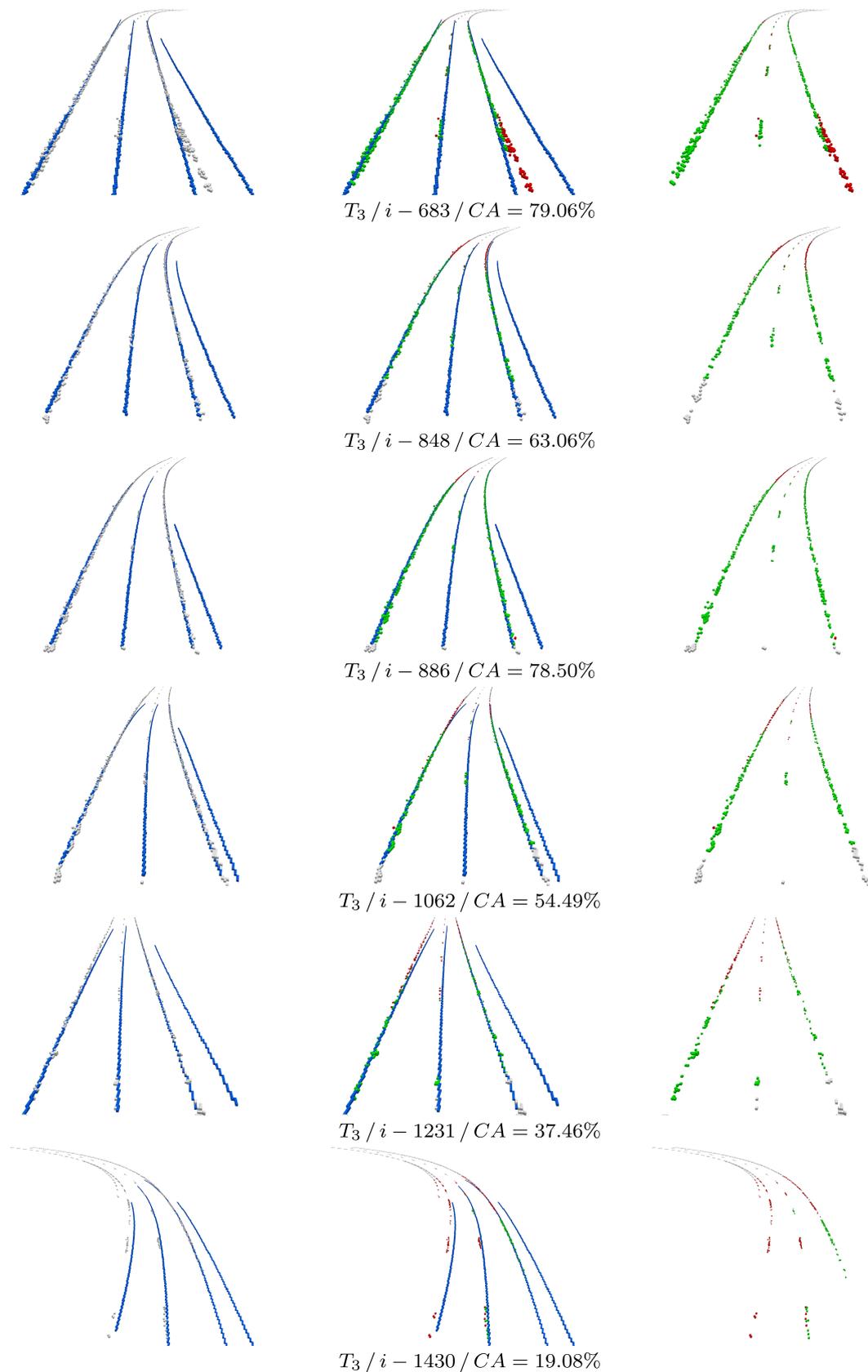


FIGURE 5.17 – Évaluation des performances de la solution de perception  $T_3$  : La colonne de gauche est la superposition des voxels correspondant aux lignes détectées par la solution  $T_3$  (en bleu) et des voxels de  $O_{ref}$  représentant les lignes (en gris). La colonne du centre correspond à la même superposition, mais les voxels  $O_{ref}$  sont colorisés en fonction des comparaisons effectuées. La somme des voxels colorisés (rouges et verts) correspond à la valeur  $N_{ref}$ , la somme des voxels verts correspond à  $N_{pred}$ . La colonne de droite reprend la colonne centrale en conservant uniquement les voxels de  $O_{ref}$  pour plus de lisibilité.

### 5.3.3.1 Estimation du paramètre $P_{max}$

Comme nous avons pu l'introduire (Sec 5.3.1.3), chaque trame de données fournie par la solution  $T_3$  correspond à un ensemble de polynômes correspondant aux lignes détectées ainsi qu'un paramètre  $P_{max}$  indiquant la portée sur laquelle les polynômes peuvent être appliqués. On peut voir (Fig 5.9) qu'au cours de la séquence étudiée,  $P_{max}$  varie selon l'intervalle [38.87m ; 76.00m]. En représentant ce paramètre en fonction de la trajectoire (Fig 5.18), on constate que les variations de ce paramètre sont liées à cette trajectoire. Ainsi, les valeurs de  $P_{max}$  sont plus importantes lorsque la plateforme évolue avec une trajectoire linéaire et diminuent lorsque la trajectoire entre dans les courbes des virages  $P_{max}$ .

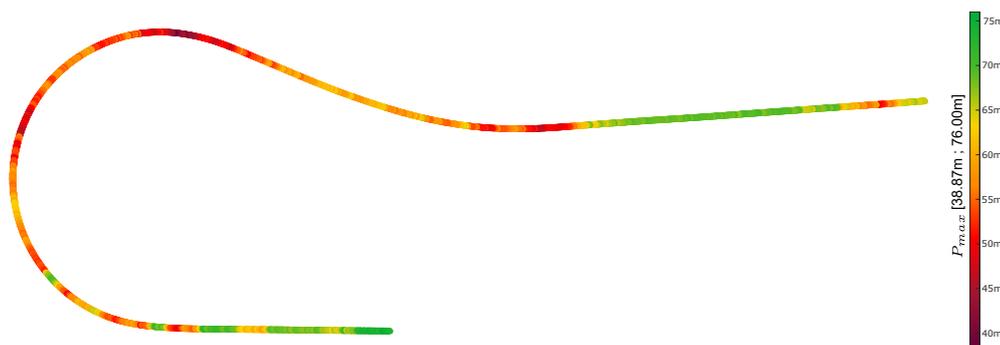


FIGURE 5.18 – Représentation du paramètre  $P_{max}$  en fonction de la trajectoire suivie par la caméra

En observant la figure (Fig 5.17), on constate également que la majorité des différents cas illustrés sont impactés par un phénomène similaire. Plus on s'éloigne de la position du capteur, moins les lignes sont détectées précisément. C'est pourquoi, sur cette figure, les lignes situées au bas des illustrations correspondent à celles les plus proches de la plateforme et sont généralement vertes. A l'inverse, les extrémités des lignes détectées représentées sur le haut des illustrations, sont généralement moins précises et représentées en rouge.

A partir de ces constats, la première hypothèse étudiée correspond à la surévaluation du paramètre  $P_{max}$  proposé par  $T_3$ . Pour appuyer cette hypothèse, nous avons réalisé de nouvelles évaluations de  $T_3$ , sur les mêmes données, mais en limitant la portée sur laquelle les polynômes sont appliqués. Nous avons donc évalué  $T_3$  en fonction d'un paramètre  $P_{test}$  limitant cette portée. Au cours de l'évaluation, lorsque  $T_3$  proposait une valeur de  $P_{max}$  supérieure à  $P_{test}$ , cette première valeur était ramenée à  $P_{test}$ . Grâce à ce procédé nous avons pu conserver l'ensemble des trames traitées, tout en limitant la portée des lignes détectées en fonction de  $P_{test}$ .

Ces nouvelles évaluations ont été réalisées pour les quatre valeurs de  $P_{test}$  suivantes : 15m, 30m, 45m et 60m. Les résultats moyens de ces évaluations sont synthétisés dans (Tab 5.3) et peuvent être comparés à ceux obtenus avec l'utilisation de  $P_{max}$ , sans limitation de la portée.

| Résolution $r$ | Portées maximales |                  |                  |                  |                               |
|----------------|-------------------|------------------|------------------|------------------|-------------------------------|
|                | $P_{test} = 15m$  | $P_{test} = 30m$ | $P_{test} = 45m$ | $P_{test} = 60m$ | $P_{max}$ (aucune limitation) |
| 0.10m          | 62.88 %           | 64.77 %          | 58.55 %          | 50.45 %          | 44.83 %                       |
| 0.20m          | 74.82 %           | 77.93 %          | 74.88 %          | 68.03 %          | 62.35 %                       |
| 0.40m          | 79.89 %           | 83.93 %          | 83.21 %          | 77.93 %          | 73.23 %                       |
| 0.80m          | 81.30 %           | 84.91 %          | 84.85 %          | 81.39 %          | 77.56 %                       |

TABLE 5.3 – Résultats de la solution de perception  $T_3$  pour différentes portées maximales

On peut ainsi voir qu'en limitant  $P_{max}$  à une valeur inférieure à celles proposées directement par la solution  $T_3$ , les performances moyennes sont améliorées. A l'inverse, l'utilisation de  $P_{max}$  sans limitation représente le cas le plus pessimiste. Ces premiers résultats peuvent déjà confirmer cette première hypothèse selon laquelle  $T_3$  surévalue la portée  $P_{max}$  sur laquelle les polynômes peuvent être appliqués. Pour aller plus en détails sur ces résultats, il est possible d'étudier les courbes de la métrique  $CA$  au cours de la séquence et en fonction du paramétrage utilisé ( $P_{test}$  ou  $P_{max}$ ). Ces courbes sont visibles (Fig 5.19) pour deux résolutions :  $r = 10cm$  et  $r = 20cm$ , qui représentent les deux résolutions les plus détaillées et fidèles à la scène.

A partir de ces courbes, il est possible de mieux interpréter les résultats synthétisés dans (Tab 5.3) et de compléter les constats déjà réalisés. Le premier est que de façon globale et sur l'ensemble de la séquence, plus on restreint la portée maximale utilisée pour les polynômes, plus les performances ont tendance à être élevées. Ce constat permet d'appuyer davantage notre hypothèse d'une estimation trop optimiste de  $P_{max}$  par  $T_3$ .

On peut cependant remarquer (Fig 5.19) que pour les trames de données situées approximativement dans l'intervalle spécifique  $i = [1700 ; 2200]$ , le phénomène inverse se produit : plus la portée maximale est faible, plus les performances diminuent. Ce second constat permet de comprendre pourquoi les résultats pour  $P_{test} = 15m$  sont légèrement inférieurs à ceux de  $P_{test} = 30m$ , ce qui n'était pas cohérent avec la tendance globale de l'hypothèse développée au cours de cette section. Pour mieux comprendre ce phénomène, de nouvelles études sont réalisées (Sec 5.3.3.2).

### 5.3.3.2 Limites paramétriques de la solution $T_3$

Dans le but d'interpréter le phénomène observé (Sec 5.3.3.1), nous avons analysé qualitativement les résultats en représentant graphiquement l'évolution de la métrique  $CA$  en fonction de la trajectoire suivie par la plateforme. Pour ce faire nous avons utilisé les valeurs de la métrique  $CA$  correspondant aux courbes 1-A, 1-B, 2-A et 2-B (Fig 5.19) et pour lesquelles le phénomène est le plus marqué. Cela nous permet d'étudier les représentations graphiques (Fig 5.20).

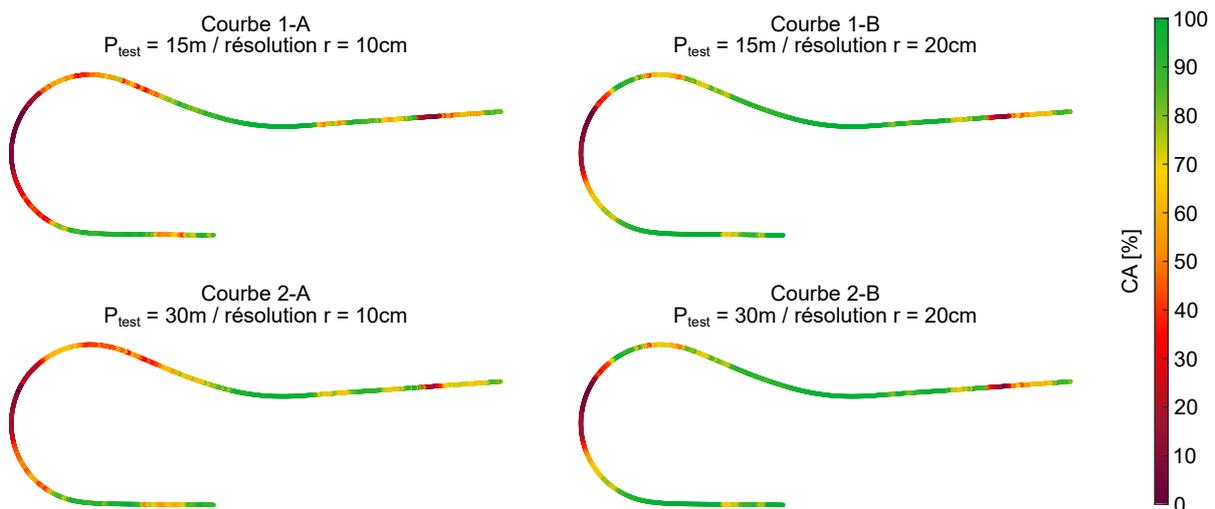


FIGURE 5.20 – Représentation de la métrique  $CA$  en fonction de la trajectoire suivie par la plateforme : Chaque représentation correspond à l'une des courbes présentées (Fig 5.19).

A partir de la (Fig 5.20), il est possible d'associer l'extrait de la trajectoire correspond à l'intervalle  $i = [1700 ; 2200]$  et pour lequel une chute des performances est visible (Fig 5.19) sur les courbes 1-A, 1-B, 2-A et 2-B. Il s'agit donc du centre du virage principal.

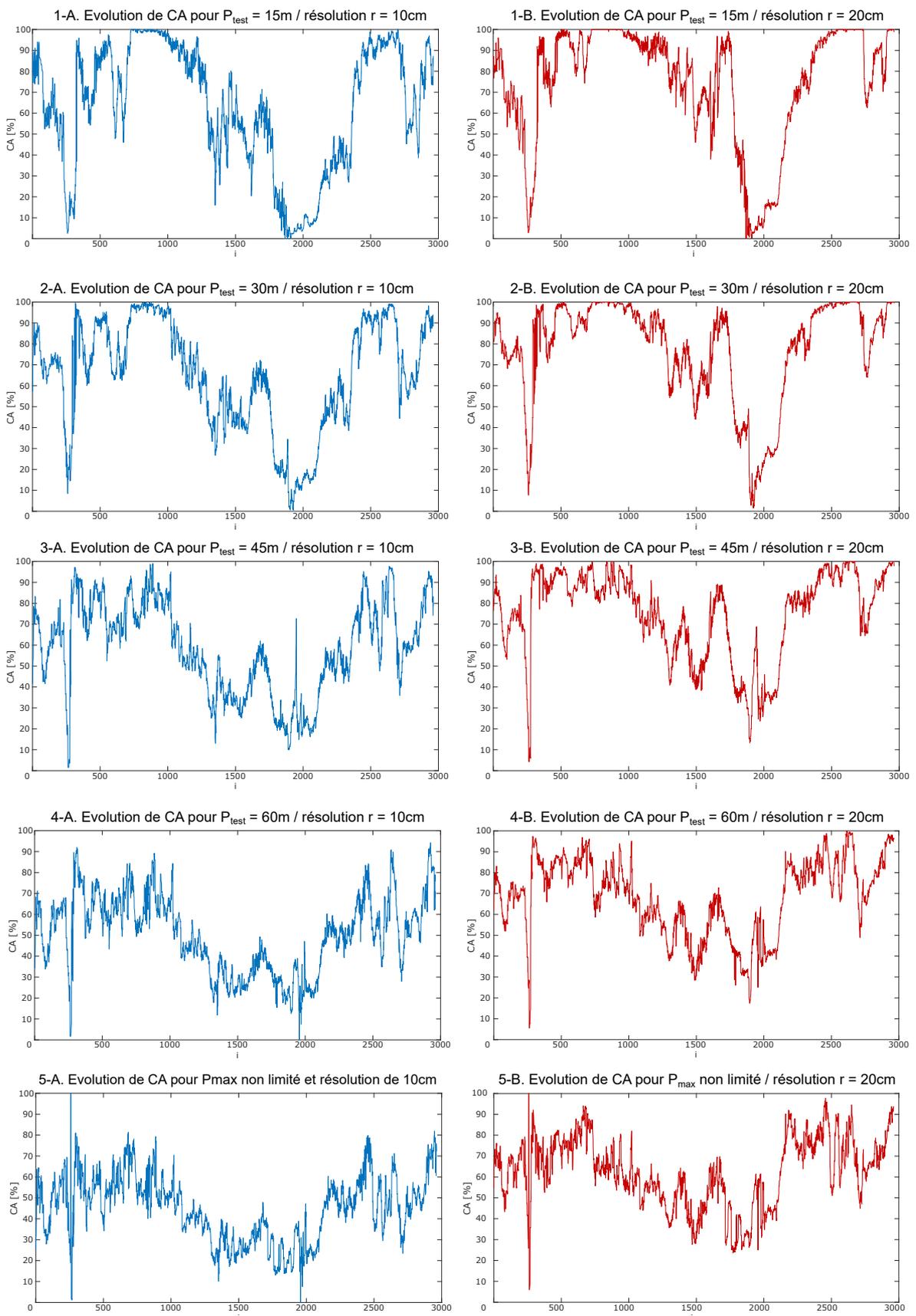


FIGURE 5.19 – Courbes de  $CA$  en fonction des paramétrages de portée maximale : Les courbes représentent l'évolution de la métrique  $CA$  au cours de la séquence. Chaque ligne correspond à une portée maximale limitée par  $P_{test}$  ou sans limite avec  $P_{max}$ . La colonne de gauche représente les courbes pour une résolution  $r$  de 10 cm, la colonne de droite correspond à une résolution  $r$  de 20 cm.

Compte tenu des observations réalisées (Sec 5.3.3.1), ainsi qu'au cours de cette section, une nouvelle hypothèse peut être émise. En effet, on sait que la solution  $T_3$  est réservée à une exploitation dans un environnement autoroutier, où les rayons de courbure sont généralement limités à des valeurs faibles. On peut donc supposer que les rayons de courbure exploitables par cette solution soient bornés par une valeur maximale.

Au cours de notre scénario, on voit (Fig 5.20) que les performances chutent au centre du virage principal, à l'instant où le rayon de courbure est le plus important. Il est donc envisageable que le rayon de courbure de ce virage dépasse la borne de valeur maximale implémentée au sein de  $T_3$ . Les coefficients des polynômes transmis par cette solution seraient limités par des valeurs maximales, empêchant à cette solution d'avoir des détections correctement alignées avec les lignes réelles. Ce phénomène impacterait majoritairement les lignes détectées à proximité du véhicule. A l'inverse, les marquages les plus éloignés, bénéficiant de la réduction progressive du rayon courbure, seraient donc moins affectés par ce phénomène.

Cette hypothèse permet de comprendre la raison pour laquelle, à ce passage spécifique du jeu de données, des performances plus faibles sont observées (Fig 5.19) lorsque la portée maximale prise en compte est plus courte. Cela permet également de justifier pourquoi les résultats observés (Tab 5.3) pour  $P_{test} = 15m$  sont légèrement inférieurs à ceux pour  $P_{test} = 30m$ . L'impact de cette limite sur l'intervalle  $i = [1700 ; 2200]$  serait suffisamment important pour que les résultats moyens sur l'ensemble de la séquence soient plus faibles qu'avec  $P_{test} = 30m$ . En connaissant le rayon de courbure maximal toléré par  $T_3$  et en le comparant au rayon de courbure du virage présent dans le scénario, il serait possible de confirmer cette hypothèse. Cependant, cette solution fonctionnant en boîte noire, cette information n'est pas donnée et il n'est pas possible de totalement valider cette hypothèse.

### 5.3.3.3 Hypothèse de sol modélisé par un plan et paramètres de calibration

Enfin, une dernière hypothèse que nous avons étudiée est l'impact de l'hypothèse de sol modélisable par un plan introduite (Sec 5.3.1.3). Elle est utilisée pour calculer les points 3D représentant les lignes, à partir des coefficients de polynômes transmis par  $T_3$  et des paramètres de calibration extrinsèques. Les imprécisions liées à cette hypothèse ne serait donc pas dépendantes de la solution en elle-même, mais de la méthode utilisée pour exploiter les données de cette solution.

Lorsque l'on superpose les courbes de la métrique  $CA$  pour les résolutions  $r = 10cm$  et  $r = 20cm$  comme dans (Fig 5.21), on peut voir que les différences de performance obtenues entre les deux résolutions ne sont pas homogènes pour toutes les portées maximales utilisées. En effet, lorsque  $P_{test} = 15m$  et  $P_{test} = 30m$ , on remarque que sur la globalité de la séquence, les différences de valeurs de  $CA$ , entre les deux résolutions, sont assez faibles. Au-delà de 30m, les écarts entre les deux courbes deviennent de plus en plus importants, avec notamment les courbes correspondant à la résolution  $r = 20cm$  affichant de bien meilleurs résultats que celles associées à la résolution  $r = 10cm$ .

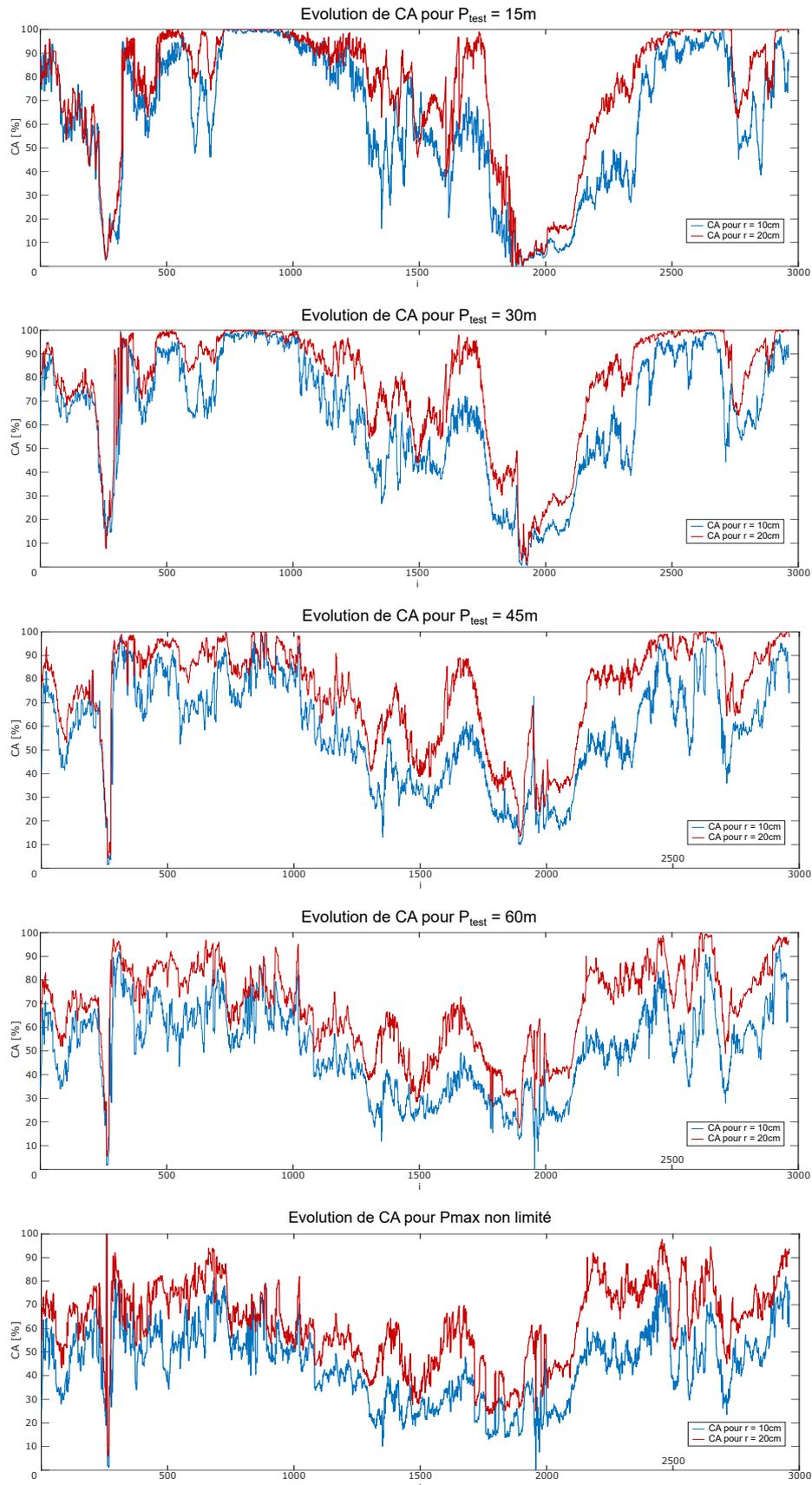


FIGURE 5.21 – Superposition des courbes de  $CA$  pour deux résolutions d'octree : Les courbes bleues représentent l'évolution de  $CA$  au cours de la séquence pour une résolution d'octree de  $r = 10\text{cm}$ . Les courbes rouges représentent quant à elles la même évolution de  $CA$  mais avec une résolution d'octree de  $r = 20\text{cm}$ .

La représentation de la métrique  $CA$  sous forme d'histogrammes de fréquence (Fig 5.22) permet de confirmer un changement de tendance entre l'utilisation d'une portée maximale inférieure à 30m et l'utilisation d'un portée maximale supérieure à 30m.

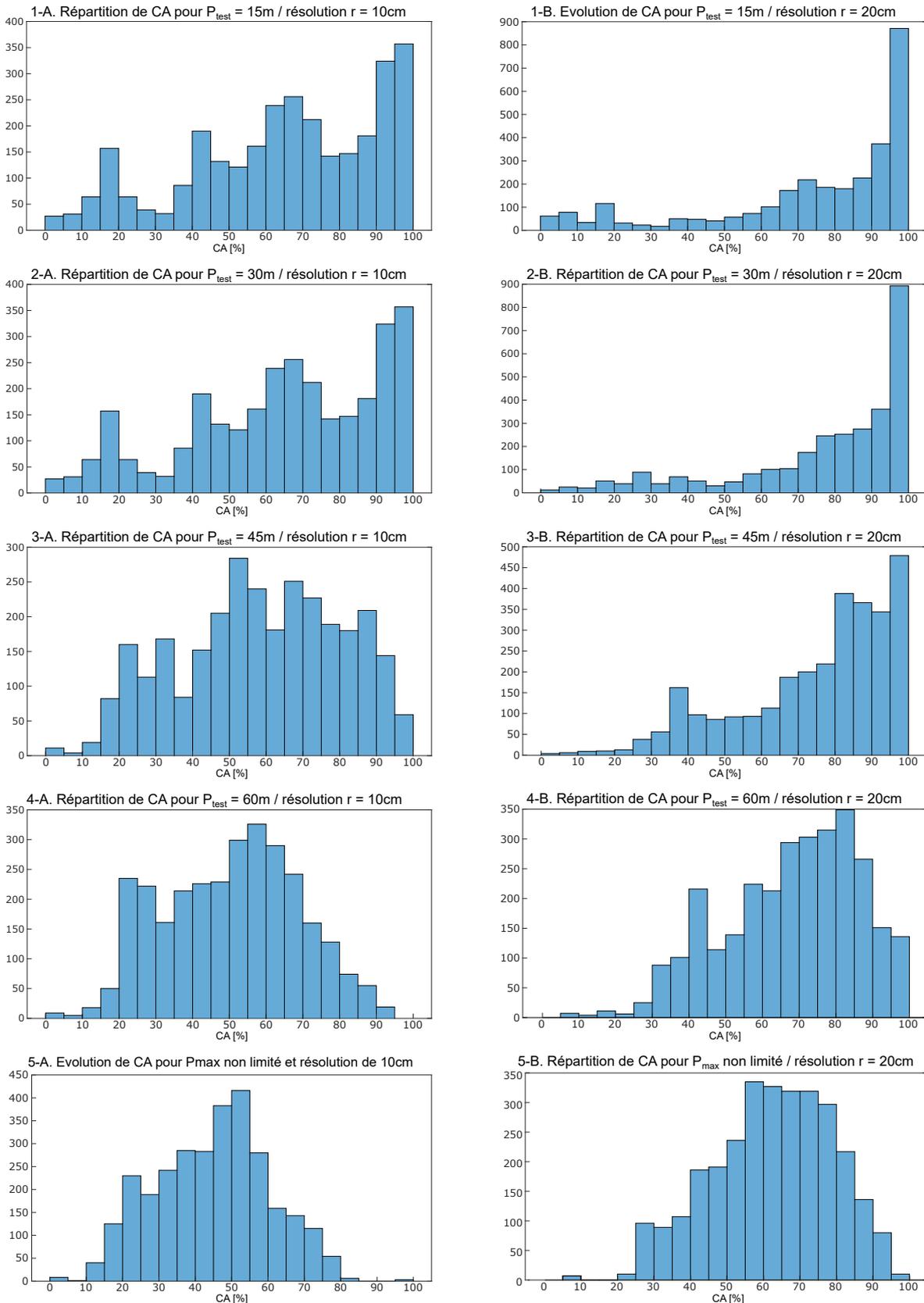


FIGURE 5.22 – Représentation de  $CA$  par histogrammes de fréquence

La raison pouvant expliquer ces constats est qu'en diminuant la résolution, on utilise des voxels de taille plus importante et qui permettent de compenser, dans une certaine mesure, les variations pouvant apparaître entre les lignes projetées en 3 dimensions à partir des données de  $T_3$  et les lignes de l'octree  $O_{ref}$ , déterminées directement en 3 dimensions, à partir des données LiDAR. Ces variations peuvent être issues de deux sources :

1. La première source est un défaut dans l'estimation des paramètres de calibration extrinsèques de la caméra. Cela peut amener à une erreur d'alignement angulaire et donc, un défaut d'orientation des lignes projetées. Plus la portée maximale utilisée est importante, plus l'impact de ce défaut d'alignement angulaire sera important sur la position 3D finale des voxels. L'utilisation de courtes portées contribue à limiter les effets de cette erreur d'alignement, ce qui expliquerait pourquoi dans (Fig 5.21), les écarts de performances entre les deux résolutions sont plus faibles sur les courbes avec  $P_{test} = 15m$  et  $P_{test} = 30m$ , par rapports à ceux observés entre les courbes correspondant à  $P_{test} = 45m$ ,  $P_{test} = 60m$  et  $P_{max}$  ;
2. La seconde source est l'utilisation de l'hypothèse d'un sol modélisé par un plan qui peut également amener à des erreurs d'alignement entre les lignes projetées à partir des données de  $T_3$  et les lignes issues des données LiDAR. En cas de variations de pente du sol, cette hypothèse amène à des erreurs car les lignes sont calculées sur la base d'une inclinaison régulière et ne varient pas en fonction de l'inclinaison réelle du sol. Si des variations de pentes sont présentes dans l'environnement, cette hypothèse n'est alors plus valable. Là encore, plus la portée maximale utilisée est importante, moins l'hypothèse d'un sol plan modélisable par un plan est valable et plus les risques d'erreurs sur la position 3D finale des voxels sont importants.

Ces deux sources peuvent être à l'origine des constats réalisés à partir des représentations (Fig 5.21) et (Fig 5.22). Elles permettent alors d'ouvrir des perspectives, notamment sur l'amélioration de l'utilisation des données de  $T_3$  et la projection des lignes en 3 dimensions. Inclure une prise en compte des variations de pentes du sol permettrait de corriger les erreurs liées à l'hypothèse de sol modélisable par un plan, d'améliorer les résultats de façon globale et de réduire les écarts de performances entre les différentes résolutions  $r$  utilisées.

Du point de vue de la méthode d'évaluation, une projection des octrees en 2D permettrait d'éviter des erreurs liées aux inclinaisons et amenant à des différences de coordonnées  $Z$  entre les voxels appartenant à  $O_{test,i}$  et ceux appartenant à  $O_{ref}$ . En revanche, l'évaluation de la solution serait alors limitée uniquement au positionnement des lignes selon les coordonnées  $(X, Y)$  des voxels.

## 5.4 Perspectives

A la suite de ces expériences, plusieurs perspectives peuvent être évoquées. Dans un premier temps, certaines perspectives peuvent être réalisées dans l'objectif de renforcer les preuves de concept apportées au cours de nos travaux. D'autres perspectives visent quant à elles, à intégrer de nouvelles informations pour élargir les possibilités de l'approche présentée.

### 5.4.1 Évaluation de solutions connues

Une première perspective à court terme serait d'appliquer notre approche sur d'autres solutions de perception, issues de la littérature et ayant déjà fait l'objet d'évaluation au sein d'outils de benchmarks connus tel que le KITTI Vision Benchmark. Pour une solution donnée, il serait alors possible de comparer les performances évaluées par notre approche, avec celles évaluées par ce type d'outils. Cela permettrait de positionner notre approche vis-à-vis de ces outils et de voir si notre approche de validation est plus optimiste, pessimiste ou du même ordre de tolérance que les approches utilisées par ces outils.

### 5.4.2 Prise en compte des variations temporelles

A plus long terme, d'autres perspectives peuvent également être évoquées. L'une d'entre elles est la prise en compte de l'aspect temporel, à la fois au cours du référencement et au cours des évaluations. Actuellement, l'approche repose principalement sur l'analyse spatiale et sémantique de la scène, via le géoréférencement et l'utilisation des poses des différents capteurs. Le référencement et les solutions évaluées sont principalement portés sur les objets composant la structure de l'environnement et qui sont fixes.

Introduire une analyse temporelle à cette première implémentation de l'approche permettrait de prendre en considération l'aspect dynamique de la scène. L'identification des objets en déplacement pourrait alors faire l'objet de traitements spécifiques. Ce premier apport permettrait d'élargir les possibilités du point de vue des solutions de perception évaluées. Des solutions de perception ciblées sur des objets en mouvement, comme pour le suivi de cibles par exemple, pourraient être prises en compte par notre approche. Il serait également possible d'intégrer dans les données de référence des objets dynamiques en déployant des scénarios maîtrisés dans l'environnement d'essai.

### 5.4.3 Diversification des systèmes évalués

Enfin, au cours de ces travaux, nous nous sommes concentrés sur des fonctions ADAS s'appuyant sur la vision et le LiDAR. Des solutions basées sur d'autres modalités pourront également être évaluées. On a vu qu'avec la prise en compte de l'aspect temporel, des fonctions ADAS ciblées sur des objets dynamiques pourront également être prises en compte. Cela permettra dans un premier temps d'utiliser l'approche avec des solutions plus hétérogènes.

Cependant, d'autres domaines pourraient être ciblés comme des systèmes de cartographie. On a pu voir que les données de référence pouvaient être utilisées avec l'intégration temporelle des octrees, amenant à une carte octree. Il est donc possible d'envisager l'évaluation de cartes issues d'algorithmes de SLAM, grâce aux cartes utilisées au sein de notre approche.

## 5.5 Conclusion

Ce chapitre 5 nous a permis de présenter le processus suivi pour évaluer une solution de perception. Différents critères permettant d'évaluer les performances de solutions testées ont été introduits. Enfin, au cours des expériences menées durant cette thèse, notre approche a pu être appliquée sur trois solutions de perception différentes. Les résultats de ces évaluations ont pu être détaillés et analysés. A l'issue de ce chapitre, plusieurs aspects peuvent être discutés :

- Un premier aspect est que ces évaluations ont permis de mettre en avant la flexibilité de notre approche en évaluant des solutions diversifiées avec : une première solution implémentée au cours de la thèse, une seconde solution exploitée telle que proposée, mais pour laquelle les données brutes sont accessibles et enfin, une troisième solution industrielle, très fermée et fonctionnant comme une boîte noire ;
- Un deuxième aspect est que ces évaluations ont permis de démontrer l'applicabilité de notre approche sur des solutions basées sur des modalités différentes. La principale modification à mettre en place entre différentes solutions réside dans l'utilisation d'une métrique permettant d'évaluer les performances d'une solution testée de façon adaptée ;
- Au-delà de ces capacités de flexibilité et d'adaptabilité, nous avons pu voir que notre approche permet de réaliser des analyses pouvant être synthétiques ou détaillées. Elles peuvent être réalisées sur la base de résultats quantitatifs et qualitatifs. L'utilisation conjointe de ces deux types de résultats facilite l'interprétation du comportement des solutions, au cours de la séquence étudiée. Grâce à ces interprétations, l'évolution dans le temps des métriques de performances utilisées est mieux comprise. Des passages spécifiques du jeu de données peuvent être cibler et permettent d'identifier les sources d'erreurs impactant les différentes solutions évaluées ;
- Enfin, en mettant en évidence les différentes sources d'erreurs pouvant impacter les résultats obtenus de ces solutions, l'utilisation de notre approche permet à son utilisateur de discuter avec les industriels fournissant ces solutions. Les analyses réalisées permettent d'argumenter sur les limitations identifiées. Il est ainsi possible de transmettre un retour détaillé sur les phénomènes observés, tout en s'appuyant sur des constats objectifs, réalisés à partir des différents graphiques produits. Notre approche et les analyses qu'elle permet de réaliser peuvent ainsi renseigner l'utilisateur sur les capacités réelles des solutions testées. Elles contribuent également à identifier de possibles perspectives d'améliorations pour ces solutions.



# Chapitre 6

## Conclusion générale

### Sommaire

---

|  |     |
|--|-----|
| 6.1 Synthèse des travaux réalisés à l'issue de cette thèse . . . . . | 139 |
| 6.1.1 Conception d'une méthode de référencement . . . . .            | 140 |
| 6.1.2 Évaluation de solutions de perception . . . . .                | 142 |
| 6.2 Perspectives . . . . .   | 143 |
| 6.2.1 Évolution du référencement . . . . .                           | 143 |
| 6.2.2 Déploiement de l'approche . . . . .                            | 144 |
| 6.3 Publications scientifiques . . . . .                             | 144 |

---

### 6.1 Synthèse des travaux réalisés à l'issue de cette thèse

Ces travaux de thèse ont été réalisés dans le cadre de la robotique mobile et plus particulièrement sur le sujet de la caractérisation des performances de solutions de perception, associées aux fonctions ADAS, dans le cadre du véhicule autonome. Ces solutions de perception peuvent être basées sur des principes physiques de mesure différents et renvoyer des données traitées sous différentes représentations géométriques. Enfin, ces solutions peuvent être maîtrisées ou à l'inverse, fonctionner en boîtes noires.

L'objectif de ces travaux de thèse était de proposer un nouvel outil méthodologique de validation, permettant de mesurer les performances de différentes solutions de perception rencontrées pour ce domaine scientifique. Pour répondre à cette problématique, les travaux se sont structurés en deux axes principaux : 1) la conception d'une méthode de référencement automatisée des données de perception et 2) la méthode d'évaluation de solutions de perception.

Le premier axe est orienté sur la mise en place d'une méthode innovante permettant le référencement de données. Les points ciblés au cours de sa conceptualisation étaient l'automatisation du référencement, l'évolutivité en termes de scénarios et l'adaptabilité du point de vue des configurations de capteurs. Cela permet d'avoir une méthode exploitable pour évaluer des solutions de perception hétérogènes. Pour y parvenir, la méthode de référencement que nous avons mise en place se base sur l'adaptation de domaine. Par cette méthode, il a été possible de référencer des données par transfert d'informations sémantiques et de les utiliser dans le cadre du second axe : l'évaluation de solutions perception.

Le second axe est centré sur l'évaluation de solutions de perception tierces. L'objectif est d'utiliser les données référencées via la méthode proposée dans le premier axe et de les exploiter comme référence pour mesurer les performances des solutions tierces testées. Plusieurs étapes viennent compléter l'approche. Une première étape de projection est employée pour homogénéiser l'espace de représentation des données issues des solutions testées avec celui des données de référence. Une seconde étape d'évaluation mesure les performances grâce à une méthode de comparaison géométrique des données. Cela nous a permis d'évaluer trois solutions de perception différentes et de tirer des constats à partir de ces résultats.

Le référencement de données obtenu et les évaluations menées au cours de ces travaux nous permettent d'avoir des preuves du concept. Nous avons cependant pu cibler certains points ouvrant à des perspectives pour améliorer d'avantage cette première approche développée et qui pourront faire l'objet de futurs travaux de recherche.

### 6.1.1 Conception d'une méthode de référencement

Les données de référence sont un des éléments fondamentaux dans le cadre de la validation ou de la caractérisation de solutions de perception. Le besoin de ces données est donc une problématique centrale de ce domaine. Pour conceptualiser notre approche globale de validation, un état de l'art centré sur les méthodes de référencement de données existantes nous a permis d'étudier leurs forces et limitations. Ces constats, nous ont permis de cibler plusieurs points au cours de la conceptualisation de notre méthode de référencement, tels que l'automatisation du référencement, l'adaptabilité des scénarios et l'évolutivité pour la configuration de capteurs utilisés. Cette nouvelle méthode proposée devait faciliter l'accès aux données de référence en répondant aux limites rencontrées parmi les méthodes actuelles et en réussissant à combiner leurs points forts.

Lors de la mise en place de notre méthode de référencement, nous avons cherché à répondre à ces objectifs en exploitant l'adaptation de domaine. Cette technique permet d'utiliser conjointement des données issues d'un premier domaine *MAP* : une carte sémantique 3D de l'environnement d'essai, avec des données d'un second domaine *CRA*, correspondant aux conditions réelles d'acquisition. Pour opérer le référencement, le rôle de l'adaptation de domaine est de transférer les informations sémantiques contenues dans la carte 3D, vers les données acquises par un capteur de référence, en conditions réelles.

Dans ce contexte, nous avons désigné un LiDAR comme capteur de référence qui, de par ses capacités de perception et sa précision, représente un bon candidat pour ce rôle. Deux phases composent alors la méthode de référencement. La première est la génération de données de synthèse annotées. Dans l'objectif d'optimiser le transfert, les informations sémantiques, initialement contenues dans la carte 3D, sont retranscrites à travers des données de synthèse. Elles sont issues de l'utilisation conjointe d'un simulateur, d'un modèle de capteur correspondant au capteur de référence et de la carte sémantique 3D, utilisée comme environnement de simulation. Cet ensemble permet d'obtenir des données de synthèse annotées sémantiquement et proches des données acquises en conditions réelles. L'utilisation des poses de capteur de référence, mesurées par le système de localisation lors de l'acquisition, permettent de générer un scan de synthèse équivalent à chaque scan acquis.

La seconde phase du référencement consiste à associer les données de synthèse générées avec les données acquises. Ces associations entre les données permettent le transfert des informations sémantiques et ainsi, l'annotation non-supervisée des données mesurées.

Afin d'optimiser ces traitements, d'utiliser des données sur différents niveaux d'échelles et de disposer de données représentées de façon favorable à l'utilisation de différentes modalités, nous avons opté pour une structuration des données en octree. Les modifications, que nous avons apporté sur une librairie existante utilisant ces structures, nous ont permis de stocker des informations sémantiques.

Enfin, l'utilisation d'un couplage GNSS-RTK/IMU, nous a permis d'obtenir un géoréférencement précis. Nous avons pu tirer profit de cet aspect pour orienter les associations entre données de synthèse et données « réelles » sur des critères géométriques évolutifs, en partant d'une correspondance directe et en ouvrant le périmètre de recherche en se basant sur la connexité et différents niveaux de voisinage. L'objectif est de pouvoir réaliser des associations ciblées localement, tout en laissant une marge d'erreur pour englober les variations observables entre des données de synthèse idéalisées et des données acquises en conditions réelles et impactées par les phénomènes extérieurs, non reproductible par simulation.

Afin d'évaluer le potentiel de cette première implémentation de la méthode de référencement, plusieurs expériences ont été réalisées sur des jeux de données acquis, en conditions réelles, avec la plateforme MELO. Les objectifs de ces premières expériences étaient de mesurer les capacités de transfert obtenues par la méthode introduite, mais également, d'étudier son comportement face à des perturbations sur les poses de capteurs. Ces poses, mesurées par le système de localisation, sont clés dans le fonctionnement de la méthode car elles sont utilisées pour la génération des données de synthèse et pour le géoréférencement les données de perception. Après avoir réalisé un premier contrôle sur la cohérence des modèles de capteurs utilisés, les expériences se sont donc réparties en deux types de tests. Ils ont été appliqués sur un échantillon prélevé du jeu de données le plus complet à disposition et représentant environ 30% des données acquises. Le reste des données ont été exploitées au cours des évaluations menées dans le second axe des travaux.

Les premiers tests étaient ciblés sur les capacités de transfert de la méthode présentée. Nous avons appliqué la méthode de référencement sur les données qui composent l'échantillon de test. Au cours de ce référencement, le taux de voxel recevant des informations sémantiques a été mesuré par rapport au nombre de voxels composant chaque octree référencé. Ces mesures ont été réalisées distinctement pour chaque niveau d'échelle des octrees, en partant d'un niveau le plus bas ayant une résolution de 10cm. Un premier constat était que les taux n'étaient pas homogènes sur les différents niveaux d'échelles mais progressif. Plus le niveau d'échelle augmente, plus la résolution diminue et un voxel englobe plus de données, ce qui amène rapidement à un taux de voxels annotés à 100%. Avec la résolution maximale étudiée, 10cm, on a pu voir que le taux de transfert approchait déjà les 70% des voxels, qui est déjà un résultat intéressant.

Au cours de ce premier test, nous avons également pu nous intéresser aux critères d'associations utilisés au cours du transfert. Là encore, une tendance s'est dégagée avec la majorité des transferts opérés en fonction de la correspondance directe et, à l'inverse, un troisième niveau de voisinage n'apportant pas de nouvelle association.

Dans l'objectif de valider ces tendances observées, nous avons réalisés les mêmes tests sur des données de synthèse, perturbées par différents niveaux de bruit dans l'objectif de se rapprocher des données obtenues en conditions réelles. L'avantage de réaliser ces mêmes tests avec des données de synthèse est qu'il est possible de connaître *a priori* les informations sémantiques auxquelles les voxels sont associés. En plus de confirmer les tendances observées sur les données réelles référencées, nous avons pu comparer les informations sémantiques initiales à celle obtenues à l'issue du référencement. En fonction des différentes classes sémantiques prises en compte au cours de ces travaux, nous avons pu voir que certaines classes pouvaient être plus impactées par les variations de par leurs caractéristiques physiques vis-à-vis de la scène ou de par leur représentation.

La seconde partie des tests était orientée sur la robustesse de la méthode face aux imprécisions pouvant impacter les données de localisation utilisées dans différentes étapes du processus. Nous avons donc référencé les données qui composent l'échantillon de données test avec du bruit de mesure, puis comparé ces données au même échantillon référencé sans bruit ajouté pour mesurer le taux de voxels associés aux bonnes classes sémantiques. Différents niveaux de bruit croissants ont été utilisés, allant d'un écart-type de +/-10cm, jusqu'à +/-5m, ce qui représente l'ordre de précision des mesures de position basées sur un GNSS, sans corrections RTK. En fonction des classes sémantiques, on a pu voir qu'elles n'étaient pas impactées de façon homogène, ce qui rejoint le constat réalisé au cours des premiers tests.

### 6.1.2 Évaluation de solutions de perception

Le second axe est centré sur l'utilisation des données référencées automatiquement par notre approche, dans le cadre de l'évaluation de différentes solutions de perception. Notre approche étant destinée à pouvoir évaluer des solutions hétérogènes, plusieurs étapes sont nécessaires et constituent le processus d'évaluation.

Une première étape est de récupérer les données traitées en sortie de la solution de perception testée pour les géoréférencer dans le même référentiel que les données de référence. Cela permet d'avoir une cohérence spatiale entre les données évaluées et les données de référence. Pour les solutions de perception correspondant à d'autres modalités que celle des LiDARs 3D, une étape de projection est nécessaire. Cette étape vise à projeter les données évaluées dans le même espace de représentation que celui utilisé par les données de référence. Les données évaluées sont à leur tour structurées sous forme d'octrees, facilitant davantage leur mise en comparaison avec les données de référence. Enfin, la dernière étape est la mise en comparaison de ces données, permettant de calculer des métriques adaptées aux solutions de perception testées et permettant de d'évaluer leur niveau de performance. A l'issue de ces comparaisons, des analyses peuvent être réalisées sur la base de résultats quantitatifs ou qualitatifs.

Pour mettre en pratique l'approche développée, nous avons évalué trois solutions de perceptions distinctes, chacune rattachée à un capteur de perception propre, indépendant du capteur de référence, ainsi qu'à un algorithme de traitement des données spécifique. Deux modalités ont pu être évaluées avec deux des solutions basées sur des données laser : une première solution avec un LiDAR positionné sur le toit de la plateforme, la seconde avec un LiDAR positionné dans le pare-chocs avant de la plateforme. La troisième solution se base quant à elle sur la vision avec une caméra intégrée derrière le pare-brise. L'utilisation de notre approche nous a permis d'obtenir des résultats synthétiques que nous avons pu approfondir par la suite.

Les analyses approfondies, basées sur l'utilisation de différents types de graphiques, ont permis d'en apprendre davantage sur les solutions testées. Malgré qu'une des solutions évaluées corresponde à une solution industrielle fermée et utilisée comme une boîte noire, notre approche a permis de l'analyser de façon détaillée. L'utilisation conjointe des résultats représentés sous différentes formes a permis d'interpréter plus facilement le comportement des solutions au cours de la séquence étudiée. Les différents facteurs impactant les performances des solutions de perception évaluées ont pu être identifiés. L'approche proposée permet de renseigner l'utilisateur sur les performances réelles d'une solution testée, de comprendre les limitations de cette solution, mais également de disposer d'analyses pouvant être retournées vers l'industriel fournissant cette solution. En identifiant les faiblesses et limitations des solutions testées, notre approche peut également servir de source de nouvelles perspectives d'améliorations de ces solutions.

Les différents résultats obtenus et les constats réalisés à partir des différentes expériences représentent de réelles preuves de concept de notre approche. Ils sont encourageants pour la poursuite de ces travaux dans la direction de l'approche proposée et amènent à de nouvelles perspectives qui sont discutées dans la suite de cette conclusion.

## 6.2 Perspectives

A l'issue de ces travaux, des perspectives peuvent être discutées sur les deux axes ciblés. Les premières perspectives concernent donc de potentielles améliorations vis-à-vis de la méthode de référencement des données. Les autres perspectives concernent le déploiement de l'approche de validation sur de nouvelles solutions de perception destinées aux ADAS et dans d'autres domaines.

### 6.2.1 Évolution du référencement

Du point de vue du référencement, une première évolution pouvant être étudiée porte sur la stratégie d'association utilisée pour transférer les informations sémantiques depuis les données de synthèse, vers les données mesurées. Plusieurs stratégies ont été énoncées et représentent des perspectives qui pourraient être étudiées dans l'objectif d'améliorer le référencement. Ces pistes cherchent à tirer des avantages de l'architecture des structures en octree, ainsi que de leurs différents niveaux d'échelles pour améliorer les taux de référencement.

Une deuxième piste évoquée repose sur une adaptation du niveau d'échelle utilisé au cours du transfert, en fonction de la classe sémantique ciblée. Certaines classes sémantiques (par exemple, la végétation hors des zones de déplacement de la plateforme) ne requièrent pas la même sensibilité que d'autres classes plus sensibles (telles que les marquages au sol). En prenant en compte cet aspect, il serait possible d'utiliser une résolution de voxel adapté à la classe sémantique représentée pour améliorer ce transfert d'informations sémantiques.

Une troisième piste discutée concerne le rôle du géoréférencement dans la méthode de référence actuelle. En basant les associations de données sur une analyse spatiale, nous avons pu observer une dépendance à la précision des poses de capteur. Celles-ci sont calculées à partir des mesures du système de localisation et la qualité du référencement dépend indirectement de la précision de ces mesures. Dans l'objectif de réduire cette dépendance, une première piste d'étude serait d'apporter une étape d'alignement entre les données de synthèse et les données mesurées. Cette étape aurait pour rôle de corriger les éventuelles erreurs de positionnement ou d'orientation produites par le système de localisation. Indirectement, cela permettrait de réduire l'importance du rôle de ce système au sein de l'approche et d'ouvrir les possibilités d'utilisation de systèmes moins coûteux que celui utilisé au cours de ces travaux.

Dans l'objectif d'apporter de nouvelles informations au référencement, une autre piste d'étude serait d'intégrer l'analyse temporelle des données. Comme discuté, cela permettrait de prendre en considération la dynamique de la scène. Cela permettra d'identifier les objets en mouvement, de créer des scénarios dans lesquels ces objets pourront être référencés et ainsi offrir davantage de possibilités vis-à-vis des solutions de perception évaluées.

Enfin, au cours de discussions plus récentes, une nouvelle perspective peut venir compléter celles déjà évoquées au cours de ces travaux. Il s'agirait d'apporter de nouvelles modifications à la structure octree utilisée durant ces travaux. Ces modifications auraient pour but de pouvoir utiliser des voxels multi-classes.

Les voxels représentant un certain volume de l'espace en fonction de la résolution fixée, plusieurs points proches dans l'espace et associés à différentes classes sémantiques pourraient être regroupés dans un même voxel. La possibilité de pouvoir attribuer différentes classes sémantiques à un voxel permettrait d'éviter de faire un choix, comme il est fait actuellement et d'éviter la perte d'informations.

### 6.2.2 Déploiement de l'approche

Du point de vue de l'application de l'approche dans le cadre des évaluations de solutions de perception, la perspective de prise en compte de l'aspect temporel, déjà évoquée, aura également un rôle à jouer. Celle-ci permettra d'évaluer un spectre de solutions plus large, notamment avec des solutions ciblées sur des objets en mouvement. L'évaluation de solutions dans des scénarios contenant des objets dynamiques renforcera l'adaptabilité de notre approche.

Dans l'objectif de démontrer la flexibilité de l'approche, l'évaluation de nouvelles solutions représente une perspective pouvant être mise en place à court terme. L'évaluation de solutions très hétérogènes, reposant sur des modalités de perception différentes de celles utilisées au cours de ces travaux, ou sur des algorithmes de traitement différents permettront de renforcer cet aspect d'adaptabilité. Toujours dans cet optique, l'évaluation de solutions destinées à d'autres domaines que celui des fonctions ADAS est également une piste qui pourrait être étudiée et renforcer davantage les atouts associés à notre approche.

Enfin, une autre perspective réalisable à court terme serait d'évaluer des solutions de perception trouvées dans la littérature et ayant déjà fait l'objet d'évaluation au sein des publications ou via des outils de benchmark populaires. En utilisant les mêmes métriques d'évaluation, il serait possible d'appliquer notre approche sur ces solutions et de la positionner vis-à-vis des évaluations déjà menées sur ces solutions. Cela permettrait d'apporter une nouvelle preuve de concept pour cette première implémentation de notre approche et encouragerait à lui apporter de nouvelles améliorations au cours de futurs travaux de recherche.

## 6.3 Publications scientifiques

- R. Defraiteur, S. A. Rodríguez F., M. Mittet, R. Reynaud, and N. El Zoghby. *A Date Referenced Framework for Performance Assessment of Perception Functions. An Application for Autonomous Vehicles and Advanced Driving Assistance Systems*. ISPRS Mobile Mapping Technology (MMT), May 2019, Shenzhen, China.
- R. Defraiteur, S. A. Rodríguez F., M. Mittet, R. Reynaud, and N. El Zoghby. *Towards a Reference Data Generation Framework for Performance Assessment of Perception Systems*. IEEE Intelligent Transportation Systems Conference (ITSC), Oct 2019, Auckland, New Zealand.

## **Annexe A**

# **Documentation de la centrale Ixblue Atlans-C**

# Atlans

A FOG-based high grade INS coupled with the latest GNSS technology offering a fully-integrated solution for innovative navigation and geo-referencing applications.



Atlans is a high performance all-in-one Inertial Navigation System (INS) for both land and airborne applications. Atlans benefits from smart coupling technique between iXblue Fiber-Optic Gyroscope (FOG) and data from the integrated latest generation Septentrio Global Navigation Satellite System (GNSS) receiver. In the most challenging applications, Atlans is specially designed to provide continuous and accurate 3D position, speed and orientation information.

## FEATURES & BENEFITS

- Simplified Integration with its all-in-one housing and single GNSS antenna setup.
- Continuous 200Hz output of orientation and position information even in GNSS-denied environments.
- INS/GNSS SIGIL<sup>1</sup> coupling
- Permanent quality data thanks to the associated iXblue post processing software APPS.
- 24/7 worldwide technical assistance
- 5 years warranty
- Free ITAR Component

## APPLICATIONS

- Asset management
- Land mobile mapping
- Airborne mobile mapping
- Image capture
- LIDAR mapping
- GIS data collection
- Pavement condition survey
- Tunnel mapping
- Railroad and road survey
- Underground survey
- Vehicle control and guidance
- Autonomous vehicles
- Ground Truth

## TECHNICAL SPECIFICATIONS

### PERFORMANCE | IMU<sup>(2)</sup>

|                      |       |
|----------------------|-------|
| Drift (deg/hr)       | 0.1   |
| Noise (deg/sqrt(hr)) | 0.005 |

(1) SIGIL: Septentrio iXblue GNSS Inertial link

(2) Typical RMS performance

## PERFORMANCE | LAND APPLICATIONS

### With GNSS<sup>(3)</sup>

| Correction type               | SPS Natural | SBAS  | DGNSS | PPP** | RTK***           | PPK****          |
|-------------------------------|-------------|-------|-------|-------|------------------|------------------|
| Position Horizontal (X,Y) (m) | 1.200       | 0.600 | 0.030 | 0.060 | 0.006+ 0.500 ppm | 0.006+ 0.500 ppm |
| Position Vertical (Z) (m)     | 1.900       | 0.800 | 0.050 | 0.090 | 0.01 + 1ppm      | 0.01 + 1ppm      |
| Heading <sup>(4)</sup> (deg)  | 0.080       | 0.050 | 0.030 | 0.025 | 0.020            | 0.015            |
| Roll & Pitch (deg)            | 0.020       | 0.015 | 0.010 | 0.008 | 0.008            | 0.005            |

### During GNSS outage<sup>(3)</sup>

| Outage duration              | SBAS/PPP 1000sec       | RTK*** 60sec | PPK**** 60sec |
|------------------------------|------------------------|--------------|---------------|
| Horizontal (X,Y) (m)         | 0.3% TD <sup>(5)</sup> | 0.350        | 0.150         |
| Vertical (Z) (m)             | 0.2% TD <sup>(5)</sup> | 0.300        | 0.100         |
| Heading <sup>(4)</sup> (deg) | 0.080                  | 0.025        | 0.020         |
| Roll & Pitch (deg)           | 0.020                  | 0.010        | 0.008         |

## PERFORMANCE | AIR APPLICATIONS

### With GNSS<sup>(3)</sup>

| Outage duration               | SPS/Natural | SBAS  | DGNSS | PPK**** |
|-------------------------------|-------------|-------|-------|---------|
| Position Horizontal (X,Y) (m) | 1.2         | 0.600 | 0.300 | 0.020   |
| Position Vertical (Z) (m)     | 1.9         | 0.800 | 0.500 | 0.050   |
| Heading <sup>(4)</sup> (deg)  | 0.080       | 0.050 | 0.030 | 0.015   |
| Roll & Pitch (deg)            | 0.020       | 0.015 | 0.010 | 0.005   |

## Characteristics

|                        |   |
|------------------------|---|
| Weight                 | 2.6 kg  |
| Material               | Aluminium   |
| Size                   | 160 mm x 160 mm x 113 mm  |
| Power                  | < 22 W, 12 to 33 VDC  |
| Operating temperature  | -20°C to 55°C   |
| Storage temperature    | -40°C to 80°C   |
| Logging capacity       | 48 hours (INS and GNSS data)  |
| MTBF                   | Environmental 100,000 hours   |
| Standard               | IP 66   |
| GNSS supported signals | GPS (L1, L2, L3, L5), GLONASS (L1, L2, L3), GALILEO* (E1, E5a, E5b, AltBOC, E6), BEIDOU* (B1, B2, B3), SBAS (EGNOS, WAAS, GAGAN, MSAS, SDCM) (L1, L5), IRNSS (L5)*, QZSS (L1, L2, L3, L5, L6) |
| PPP Support            | TerraStar and VERIPOS services <sup>(6)</sup>   |
| RTCM Support           | RTCM v2.2, 2.3, 3.0, 3.1  |

\*Optional features

<sup>(3)</sup> Typical RMS Performance

<sup>(4)</sup> Secant latitude = 1 / cosine latitude

<sup>(5)</sup> TD = Travelled Distance (CEP50), with DMI

<sup>(6)</sup> Requires service subscription

\*\* PPP: Precise Point positioning

\*\*\* RTK: Real-Time Kinematic, up to 40km from base stations

\*\*\*\* PPK: Post processing Kinematic using Advanced Post-Processing Software (smart coupling of INS with GNSS in forward/backward)/

## INTERFACES

|  |   |
|--|---|
| Output refreshing rate                       | 0.1 Hz to 200 Hz                                  |
| Latency                                      | < 3ms   |
| Time tagging                                 | PPS output  |
| Ethernet                                     | UDP / TCP Client / TCP server                     |
| Serial RS232 or RS422                        | 2 inputs / 2 outputs + 2 dedicated                |
| Input / Output formats                       | Industry standards: NMEA0183, ASCII, BINARY       |
| Pulses                                       | 3 in/2 out  |
| Options & accessories                        | DMI (odometer)<br>APPS (post processing software) |
| GNSS interface embedded                      |   |
| Serial or Ethernet link to the embedded GNSS |   |

contact@ixblue.com | www.ixblue.com

EMEA +33 1 30 08 88 88 | Americas +1 781 937 8800 | APAC +65 6747 4912

**ixblue**

## Annexe B

# Transformation rigide

Une transformation rigide correspond à la transformation permettant de projeter des données appartenant à un référentiel euclidien  $A$  vers un référentiel euclidien  $B$  (Fig B.1).

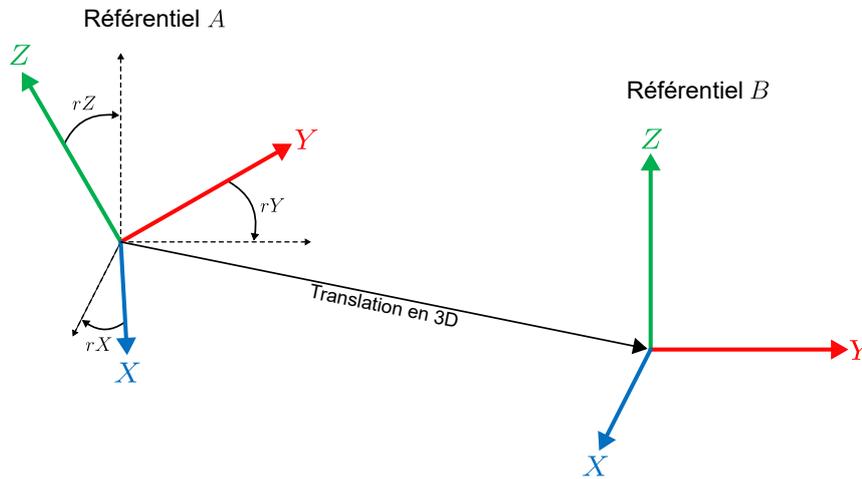


FIGURE B.1 – Représentation d'une transformation rigide.

Dans le cadre d'un point 3D, cette transformation correspond à l'équation (Eq B.1).

$$p^B = R * p^A + T \quad (\text{B.1})$$

où  $p^A$  et  $p^B$  représentent respectivement le même point dans le référentiel  $A$  puis  $B$ .  $T$  est le vecteur de translation entre les origines des deux référentiels et se compose de trois valeurs  $T = [dX_{A \rightarrow B}, dY_{A \rightarrow B}, dZ_{A \rightarrow B}]$  : les distances euclidiennes séparant les deux origines de  $A$  et  $B$ , en fonction des axes  $(X, Y, Z)$ . Enfin,  $R$  représente la matrice de rotation permettant d'aligner les données en fonction des axes qui définissent le référentiel  $B$ . Elle est calculée à partir des angles de rotations  $rX$ ,  $rY$  et  $rZ$  séparant les axes des référentiels  $A$  et  $B$  et qui sont notés respectivement  $\gamma$ ,  $\beta$  et  $\alpha$ . Cette matrice est donnée par (Eq B.2).

$$R_{(\gamma, \beta, \alpha)} = \begin{pmatrix} \cos \alpha \cos \beta & \cos \alpha \sin \beta \sin \gamma - \sin \alpha \cos \gamma & \cos \alpha \sin \beta \cos \gamma + \sin \alpha \sin \gamma \\ \sin \alpha \cos \beta & \sin \alpha \sin \beta \sin \gamma + \cos \alpha \cos \gamma & \sin \alpha \sin \beta \cos \gamma - \cos \alpha \sin \gamma \\ -\sin \beta & \cos \beta \sin \gamma & \cos \beta \cos \gamma \end{pmatrix} \quad (\text{B.2})$$

Au cours de ce relevé, des points ont également été mesurés sur le sol, tout autour de la plateforme. Ces points ont permis d'estimer le plan moyen représentant le sol. A partir de la redondance des points mesurés sur les instruments embarqués, nous avons également pu définir l'inclinaison de chaque instrument par rapport au plan moyen du sol ainsi que leur hauteur respective par rapport au sol. Ces calculs ont ainsi permis d'estimer l'orientation de chaque instrument par rapport au référentiel *Ego*.

Grâce à ce procédé, nous avons pu estimer les paramètres de calibration extrinsèques ainsi que les transformations à appliquer pour exprimer entre le référentiel local, propre à chaque instrument et le référentiel *Ego*. Cela nous a permis de réunir l'ensemble des données produites sous un seul et même référentiel avant de les géoréférencées dans le système de référence  $\mathcal{G}$  utilisé.

**Annexe C**

**Documentation Riegl**

# Technical Data Mobile Laser Scanning System *RIEGL* VMX<sup>®</sup>-250

Laser Product Classification

Class 1 Laser Product according to IEC60825-1:2007

The following clause applies for instruments delivered into the United States:  
Complies with 21 CFR 1040.10 and 1040.11 except for deviations pursuant  
to Laser Notice No. 50, dated June 24, 2007.

CLASS 1  
LASER PRODUCT

## 2 x VQ-250 Measurement Performance

| Effective Measurement Rate <sup>1)</sup>   | 100 kHz                                    | 200 kHz        | 300 kHz        | 400 kHz        | 600 kHz       |
|--|--|----------------|----------------|----------------|---------------|
| Max. Measurement Range <sup>2)</sup><br>natural targets $\rho \geq 10\%$<br>natural targets $\rho \geq 80\%$ | 180 m<br>500 m                             | 130 m<br>380 m | 110 m<br>340 m | 100 m<br>300 m | 75 m<br>200 m |
| Max. Number of Targets per Pulse   | practically unlimited (details on request) |                |                |                |               |

Minimum Range

1.5 m

Accuracy <sup>3) 5)</sup>

10 mm

Precision <sup>4) 5)</sup>

5 mm

Max. Effective Measurement Rate <sup>1)</sup>

600 000 meas./sec (2 x 300 000 meas./sec)

Line Scan Speed (selectable)

up to 200 lines/sec (2 x 100 lines/sec)

1) Rounded values, selectable by measurement program.  
2) The following conditions are assumed: target larger than  
the footprint of the laser beam, perpendicular angle of  
incidence, visibility 23 km, average ambient brightness.

3) Accuracy is the degree of conformity of a measured quantity to its actual (true) value.  
4) Precision, also called reproducibility or repeatability, is the degree  
to which further measurements show the same result.  
5) One sigma @ 50 m range under *RIEGL* test conditions.

## IMU/GNSS Performance <sup>6)</sup>

Position (absolute)

typ. 20 - 50 mm

Position (relative) <sup>7)</sup>

typ. 10 mm

Roll & Pitch

0.005°

Heading

0.015°

6) One sigma values, no GNSS outages, with DMI option,  
post-processed using base station data.

7) With a control point spacing < 100 m.

## Physical Data

VMX-250-MH Measuring Head

*including GNSS antenna and protection caps*

*Main Dimensions (L x W x H)*

737 x 456 x 485 mm

*Weight (approx.)*

43 kg

VMX-250 Protective Cover

620 x 747 x 364 mm

3 kg

VMX-250-CU Control Unit

560 x 455 x 265 mm

26 kg

VMX-250-RM Roof Mount

778 x 515 x approx. 120 mm

13 kg

*including mounting brackets*

VMX-250-MC Main Cable

3 m (standard length)

5 kg

VMX-250-CS6 Camera System

607 x 1038 x 208 mm <sup>8)</sup>

19 kg <sup>8)</sup>

8) Typical configuration with 4 cameras.

## Electrical Data / Interfaces

Power Supply Input Voltage

11 - 15 V DC

Power Consumption

typ. 330 W (max. 680 W) <sup>8)</sup>

Interfaces

LAN, 10/100/1000 MBit/sec

USB 2.0

DVI

SYNC OUT (synchronization output NMEA+PPS)

NAV RS232 (COM of IMU/GNSS system for RTK, SBAS)

interfaces for additional sensor devices (scanner, cameras, etc.)

removable hard disks for project data transfer

## Environmental Data

Temperature Range

VMX-250-MH Measuring Head

-10°C to +40°C (operation) / -20°C to +50°C (storage)

VMX-250-CU Control Unit

0°C to +40°C (operation) / -20°C to +50°C (storage)

VMX-250-CS6 Camera System

-10°C to +40°C (operation) / -20°C to +50°C (storage)

Humidity

max. 80% non condensing @ +31°C

Protection Class

VMX-250-MH Measuring Head

IP64, dust and splash-proof

VMX-250-CU Control Unit

IP64 (closed lid), IP20 (open lid)

VMX-250-CS6 Camera System

IP65, dust and water jet-proof



*RIEGL Laser Measurement Systems GmbH, 3580 Horn, Austria*  
Tel.: +43-2982-4211, Fax: +43-2982-4210, E-mail: office@riegl.co.at

*RIEGL USA Inc., Orlando, Florida 32819, USA*  
Tel.: +1-407-248-9927, Fax: +1-407-248-2636, E-mail: info@rieglusa.com

*RIEGL Japan Ltd., Tokyo 1640013, Japan*  
Tel.: +81-3-3382-7340, Fax: +81-3-3382-5843, E-mail: info@riegl-japan.co.jp

www.riegl.com

## Annexe D

# Description du « wall test »

Le « wall test » est une méthode pouvant permettre de déterminer la précision des mesures de distances produites par un capteur donné. Cette méthode d'étalonnage qui consiste à positionner le capteur testé sur une base, face un mur (vertical) et où la distance horizontale  $D_{ref}$  base-mur est parfaitement connue (Fig D.1).

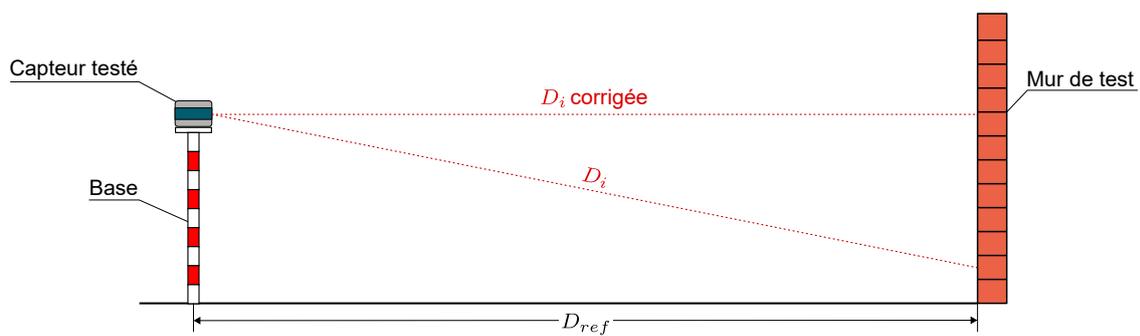


FIGURE D.1 – Représentation du « wall test »

Les distances mesurées  $D_i$  par le capteur testé peuvent alors être comparées à la distance  $D_{ref}$ . Après avoir corrigé les distances  $D_i$ , en fonction de l'angle d'élévation de la nappe auxquelles elles appartiennent, pour les ramener à l'horizontal, il est possible de comparer ces distances mesurées à la distance  $D_{ref}$ .

En répétant ce procédé sur un grand nombre de mesures, il est possible d'estimer en fonction de chaque nappe, l'ordre de précision des mesures produites par le capteur ainsi que ses écarts-types. Enfin, ce procédé peut également être répété sur plusieurs bases situées à différentes distances du mur de test. Cela permet d'estimer l'évolution de la précision du capteur testé en fonction de la portée de la mesure effectuée.

# Bibliographie

- [1] C. Thorpe, M.H. Hebert, T. Kanade, and S.A. Shafer. Vision and navigation for the carnegie-mellon navlab. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(3) :362–373, may 1988.
- [2] Ernst D. Dickmanns. Vehicles capable of dynamic vision : a new breed of technical beings ? *Artificial Intelligence*, 103(1-2) :49–76, aug 1998.
- [3] Michael Montemerlo, Jan Becker, Suhrid Bhat, Hendrik Dahlkamp, Dmitri Dolgov, Scott Ettinger, Dirk Haehnel, Tim Hilden, Gabe Hoffmann, Burkhard Huhnke, Doug Johnston, Stefan Klumpp, Dirk Langer, Anthony Levandowski, Jesse Levinson, Julien Marcil, David Orenstein, Johannes Paefgen, Isaac Penny, Anna Petrovskaya, Mike Pflueger, Ganymed Stanek, David Stavens, Antone Vogt, and Sebastian Thrun. Junior : The stanford entry in the urban challenge. *Journal of Field Robotics*, 25(9) :569–597, sep 2008.
- [4] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, jun 2012.
- [5] CNRS. Gdr robotique, 2017. [www.gdr-robotique.org](http://www.gdr-robotique.org) (Dernier acces : 05/04/2021).
- [6] O.M.J. Carsten and F.N. Tate. Intelligent speed adaptation : accident savings and cost–benefit analysis. *Accident Analysis & Prevention*, 37(3) :407–416, may 2005.
- [7] Berkeley DeepDrive Industrial Consortium. Berkeley.deepdrive, 2020.
- [8] smartmicro. Smartmicro automotive radars, 2020. <https://www.smartmicro.com/automotive-radar> (Dernier acces 05/04/2021).
- [9] Francisca Rosique, Pedro J. Navarro, Carlos Fernández, and Antonio Padilla. A systematic review of perception system and simulators for autonomous vehicles research. *Sensors*, 19(3) :648, feb 2019.
- [10] Sean Campbell, Niall O'Mahony, Lenka Krpalcova, Daniel Riordan, Joseph Walsh, Aidan Murphy, and Conor Ryan. Sensor technology in autonomous vehicles : A review. In *2018 29th Irish Signals and Systems Conference (ISSC)*. IEEE, jun 2018.
- [11] Jessica Van Brummelen, Marie O'Brien, Dominique Gruyer, and Homayoun Najjaran. Autonomous vehicle perception : The technology of today and tomorrow. *Transportation Research Part C : Emerging Technologies*, 89 :384–406, apr 2018.
- [12] Wei Zhou, Julie Stephany Berrio, Stewart Worrall, and Eduardo Nebot. Automated evaluation of semantic segmentation robustness for autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 21(5) :1951–1963, may 2020.
- [13] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes : A multimodal dataset for autonomous driving. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2020.

- [14] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. April 2016.
- [15] Oliver Mitchell. The five senses of robotics, 2017. <https://robotrabbi.medium.com/the-five-senses-of-robotics-a94366b1f2ba> (Dernier acces : 05/04/2021).
- [16] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, mar 2004.
- [17] Samira Pouyanfar, Saad Sadiq, Yilin Yan, Haiman Tian, Yudong Tao, Maria Presa Reyes, Mei-Ling Shyu, Shu-Ching Chen, and S. S. Iyengar. A survey on deep learning. *ACM Computing Surveys*, 51(5) :1–36, jan 2019.
- [18] Sorin Grigorescu, Bogdan Trasnea, Tiberiu Cocias, and Gigel Macesanu. A survey of deep learning techniques for autonomous driving. *Journal of Field Robotics*, 37(3) :362–386, apr 2020.
- [19] Christian Hane, Lionel Heng, Gim Hee Lee, Friedrich Fraundorfer, Paul Furgale, Torsten Sattler, and Marc Pollefeys. 3d visual perception for self-driving cars using a multi-camera system : Calibration, mapping, localization, and obstacle detection. *Image and Vision Computing*, 68 :14–27, dec 2017.
- [20] Alireza Asvadi, Luis Garrote, Cristiano Premebida, Paulo Peixoto, and Urbano J. Nunes. Multimodal vehicle detection : fusing 3d-LIDAR and color camera data. *Pattern Recognition Letters*, 115 :20–29, nov 2018.
- [21] Annalisa Milella, Giulio Reina, James Underwood, and Bertrand Douillard. Visual ground segmentation by radar supervision. *Robotics and Autonomous Systems*, 62(5) :696–706, may 2014.
- [22] G Di Leo, C Liguori, and A Paolillo. Covariance propagation for the uncertainty estimation in stereo vision. *IEEE Transactions on Instrumentation and Measurement*, 60(5) :1664–1673, may 2011.
- [23] Alireza Kashani, Michael Olsen, Christopher Parrish, and Nicholas Wilson. A review of LIDAR radiometric processing : From ad hoc intensity correction to rigorous radiometric calibration. *Sensors*, 15(11) :28099–28128, nov 2015.
- [24] Andrés Serna, Beatriz Marcotegui, Francois Goulette, and Jean-Emmanuel Deschaud. Paris-rue-madame database : a 3d mobile laser scanner dataset for benchmarking urban detection, segmentation and classification methods. In *4th International Conference on Pattern Recognition, Applications and Methods ICPRAM*, Angers, France, March 2014.
- [25] Raffi Mardirosian. Lidar vs. camera : driving in the rain, 2020. <https://ouster.com/blog/lidar-vs-camera-comparison-in-the-rain/> (Dernier acces : 05/04/2021).
- [26] Saurabh Gupta, Ross Girshick, Pablo Arbeláez, and Jitendra Malik. Learning rich features from RGB-d images for object detection and segmentation. In *Computer Vision – ECCV 2014*, pages 345–360. Springer International Publishing, 2014.
- [27] H. Moravec and A. Elfes. High resolution maps from wide angle sonar. In *Proceedings. 1985 IEEE International Conference on Robotics and Automation*. Institute of Electrical and Electronics Engineers, 1985.
- [28] Marek Kurdej, Julien Moras, Veronique Cherfaoui, and Philippe Bonnifait. Map-aided evidential grids for driving scene understanding. *IEEE Intelligent Transportation Systems Magazine*, 7(1) :30–41, 2015.
- [29] F. Leberl, A. Irschara, T. Pock, P. Meixner, M. Gruber, S. Scholz, and A. Wiechert. Point clouds. *Photogrammetric Engineering & Remote Sensing*, 76(10) :1123–1134, oct 2010.
- [30] Jeong heon Song, Soo hee Han, Kiyun Yu, and Yong il Kim. Assessing the possibility of landcover classification using lidar intensity data. In *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information*, 2002.

- [31] Martin Liggins. *Handbook of multisensor data fusion : theory and practice*. CRC Press, Boca Raton, FL, 2009.
- [32] Donald Meagher. Geometric modeling using octree encoding. *Computer Graphics and Image Processing*, 19(2) :129–147, jun 1982.
- [33] Armin Hornung, Kai M. Wurm, Maren Bennewitz, Cyrill Stachniss, and Wolfram Burgard. OctoMap : an efficient probabilistic 3d mapping framework based on octrees. *Autonomous Robots*, 34(3) :189–206, feb 2013.
- [34] Hanan Samet. The quadtree and related hierarchical data structures. *ACM Computing Surveys*, 16(2) :187–260, jun 1984.
- [35] Mohammed A. Quddus, Washington Y. Ochieng, and Robert B. Noland. Integrity of map-matching algorithms. *Transportation Research Part C : Emerging Technologies*, 14(4) :283–302, aug 2006.
- [36] Marie-Anne Mittet. *Cameras 3D pour la localisation d'un systeme mobile en environnement urbain*. PhD thesis, Universite de Strasbourg, 2015.
- [37] Françoise Duquenne. *GPS : localisation et navigation par satellites*. Hermès Science publications, Paris, 2005.
- [38] Jean-Marie Zogg. *GPS : essentials of satellite navigation : compendium : theory and principles of satellite navigation, overview of GPS/GNSS systems and applications*. U-Blox, Thalwil, 2009.
- [39] Sameh Nassar. Improving the inertial navigation system (ins) error model for ins and ins/dgps applications, 2003.
- [40] Michael Cramer. Gps/ins integration. *Photogrammetric Week*, 1997.
- [41] J. Borenstein, B. Everett, and L. Feng. Where am i ? sensors and methods for mobile robot positioning. 1996.
- [42] R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1) :35–45, mar 1960.
- [43] D. B. Cox. Integration of gps with inertial navigation systems. In *Journal of The Institute of Navigation*, volume 25, pages 236–245, 1978.
- [44] Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(8) :1362–1376, aug 2010.
- [45] Institut National de l'Information Geographique et Forestiere. Systemes geodesiques, 2021.
- [46] John P. Snyder. Map projections : A working manual, 1987.
- [47] Liang Xiao, Bin Dai, Daxue Liu, Dawei Zhao, and Tao Wu. Monocular road detection using structured random forest. *International Journal of Advanced Robotic Systems*, 13(3) :101, jan 2016.
- [48] Lukas Rummelhard, Anshul Paigwar, Amaury Negre, and Christian Laugier. Ground estimation and point cloud segmentation using SpatioTemporal conditional random field. In *2017 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, jun 2017.
- [49] Jian Yao, Srikumar Ramalingam, Yuichi Taguchi, Yohei Miki, and Raquel Urtasun. Estimating drivable collision-free space from monocular video. In *2015 IEEE Winter Conference on Applications of Computer Vision*. IEEE, jan 2015.
- [50] Xiaozhi Chen, Kaustav Kundu, Ziyu Zhang, Huimin Ma, Sanja Fidler, and Raquel Urtasun. Monocular 3d object detection for autonomous driving. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2016.
- [51] Zhidong Liang, Ming Zhang, Zehan Zhang, Xian Zhao, and Shiliang Pu. Rangercnn : Towards fast and accurate 3d object detection with range image representation. September 2020.

- [52] Min Bai, Gellert Mattyus, Namdar Homayounfar, Shenlong Wang, Shrinidhi Kowshika Lakshmikanth, and Raquel Urtasun. Deep multi-sensor lane detection. May 2019.
- [53] Masaru Yoshioka, Naoki Suganuma, Keisuke Yoneda, and Mohammad Aldibaja. Real-time object classification for autonomous vehicle using LIDAR. In *2017 International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS)*. IEEE, nov 2017.
- [54] A. K. Aijazi, A. Serna, B. Marcotegui, P. Checchin, and L. Trassoudaine. Segmentation and classification of 3d urban point clouds : Comparison and combination of two approaches. In *Springer Tracts in Advanced Robotics*, pages 201–216. Springer International Publishing, 2016.
- [55] Andrés Serna, Beatriz Marcotegui, and Etienne Decencière. Segmenting junction regions without skeletonization using geodesic operators and the max-tree. In *Lecture Notes in Computer Science*, pages 456–467. Springer International Publishing, 2019.
- [56] A. J. Trevor, Sumeyra Gedikli, R.B. Rusu, and Henrik Christensen. Efficient organized point cloud segmentation with connected components. In *Proceedings of Semantic Perception Mapping and Exploration*, pages 1–6, January 2013.
- [57] Andrew Tao, Karan Sapra, and Bryan Catanzaro. Hierarchical multi-scale attention for semantic segmentation. May 2020.
- [58] Tiago Cortinhal, George Tzelepis, and Eren Erdal Aksoy. Salsanext : Fast, uncertainty-aware semantic segmentation of lidar point clouds for autonomous driving. March 2020.
- [59] Hyunggi Cho, Young-Woo Seo, B.V.K. Vijaya Kumar, and Ragunathan Raj Rajkumar. A multi-sensor fusion system for moving object detection and tracking in urban driving environments. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, may 2014.
- [60] Jamil Fayyad, Mohammad A. Jaradat, Dominique Gruyer, and Homayoun Najjaran. Deep learning sensor fusion for autonomous vehicle perception and localization : A review. *Sensors*, 20(15) :4220, jul 2020.
- [61] Arjun Balakrishnan, Sergio A. Rodriguez F., and Roger Reynaud. An integrity assessment framework for multi-modal vehicle localization. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, oct 2019.
- [62] Ahmed Frikha and Hela Moalla. Analytic hierarchy process for multi-sensor data fusion based on belief function theory. *European Journal of Operational Research*, 241(1) :133–147, feb 2015.
- [63] Aharon Bar Hillel, Ronen Lerner, Dan Levi, and Guy Raz. Recent progress in road and lane detection : a survey. *Machine Vision and Applications*, 25(3) :727–745, feb 2012.
- [64] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Buló, and Peter Kotschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE, oct 2017.
- [65] A Geiger, P Lenz, C Stiller, and R Urtasun. Vision meets robotics : The KITTI dataset. *The International Journal of Robotics Research*, 32(11) :1231–1237, aug 2013.
- [66] Karlsruhe Institute of Technology. Object tracking evaluation, 2012.
- [67] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Juergen Gall. Semantickitti : A dataset for semantic scene understanding of lidar sequences. In *International Conference on Computer Vision (ICCV)*, apr 2019.
- [68] Minh Pham, Suresh Alse, Craig A. Knoblock, and Pedro Szekely. Semantic labeling : A domain-independent approach. In *Lecture Notes in Computer Science*, pages 446–462. Springer International Publishing, 2016.

- [69] Szilárd Vajda, Yves Rangoni, and Hubert Cecotti. Semi-automatic ground truth generation using unsupervised clustering and limited manual labeling : Application to handwritten character recognition. *Pattern Recognition Letters*, 58 :23–28, jun 2015.
- [70] Lawrence Reeve and Hyoil Han. Survey of semantic annotation platforms. In *Proceedings of the 2005 ACM symposium on Applied computing - SAC '05*. ACM Press, 2005.
- [71] G. Sychay, E. Chang, and K. Goh. Effective image annotation via active learning. In *Proceedings. IEEE International Conference on Multimedia and Expo*. IEEE, 2004.
- [72] Karlsruhe Institute of Technology. Kitti-360, 2020.
- [73] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. BDD100k : A diverse driving dataset for heterogeneous multitask learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2020.
- [74] Ali Al-Sarraf, Bok-Suk Shin, Zezhong Xu, and Reinhard Klette. Ground truth and performance evaluation of lane border detection. In *Computer Vision and Graphics*, pages 66–74. Springer International Publishing, 2014.
- [75] I. Kavasidis, S. Palazzo, R. Di Salvo, D. Giordano, and C. Spampinato. A semi-automatic tool for detection and tracking ground truth generation in videos. In *Proceedings of the 1st International Workshop on Visual Interfaces for Ground Truth Collection in Computer Vision Applications - VIGTA '12*. ACM Press, 2012.
- [76] Amol Borkar, Monson Hayes, and Mark T. Smith. A novel lane detection system with efficient ground truth generation. *IEEE Transactions on Intelligent Transportation Systems*, 13(1) :365–374, mar 2012.
- [77] D. Doermann and D. Mihalcik. Tools and techniques for video performance evaluation. In *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*. IEEE Comput. Soc, 2000.
- [78] Xinyu Huang, Peng Wang, Xinjing Cheng, Dingfu Zhou, Qichuan Geng, and Ruigang Yang. The ApolloScape open dataset for autonomous driving and its application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(10) :2702–2719, oct 2020.
- [79] Manikandasriram Srinivasan Ramanagopal, Cyrus Anderson, Ram Vasudevan, and Matthew Johnson-Roberson. Failing to learn : Autonomously identifying perception failures for self-driving cars. *IEEE Robotics and Automation Letters*, 3(4) :3860–3867, oct 2018.
- [80] Michael Buckland and Fredric Gey. The relationship between recall and precision. *Journal of the American Society for Information Science*, 45(1) :12–19, jan 1994.
- [81] Adrien Gaidon, Qiao Wang, Yohann Cabon, and Eleonora Vig. Virtual worlds as proxy for multi-object tracking analysis. May 2016.
- [82] Stephan R. Richter, Zeeshan Hayder, and Vladlen Koltun. Playing for benchmarks. In *2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE, oct 2017.
- [83] Jorge Beltran, Irene Cortes, Alejandro Barrera, Jesus Urdiales, Carlos Guindel, Fernando Garcia, and Arturo de la Escalera. A method for synthetic LiDAR generation to create annotated datasets for autonomous vehicles perception. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, oct 2019.
- [84] Jin Fang, Dingfu Zhou, Feilong Yan, Tongtong Zhao, Feihu Zhang, Yu Ma, Liang Wang, and Ruigang Yang. Augmented LiDAR simulator for autonomous driving. *IEEE Robotics and Automation Letters*, 5(2) :1931–1938, apr 2020.
- [85] Xiangyu Yue, Bichen Wu, Sanjit A. Seshia, Kurt Keutzer, and Alberto L. Sangiovanni-Vincentelli. A LiDAR point cloud generator. In *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*. ACM, jun 2018.

- [86] Andrew Glassner. *An Introduction to Ray Tracing*. ACADEMIC PR INC, January 1989.
- [87] Nikolaus Mayer, Eddy Ilg, Philipp Fischer, Caner Hazirbas, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. What makes good synthetic training data for learning disparity and optical flow estimation? *International Journal of Computer Vision*, 126(9) :942–960, apr 2018.
- [88] Burkhard Gusefeld, Daniel Kondermann, Christopher Schwartz, and Reinhard Klein. Are reflectance field renderings appropriate for optical flow evaluation? In *2014 IEEE International Conference on Image Processing (ICIP)*. IEEE, oct 2014.
- [89] Jeamin Koo, Jungsuk Kwac, Wendy Ju, Martin Steinert, Larry Leifer, and Clifford Nass. Why did my car just do that? explaining semi-autonomous driving actions to improve driver understanding, trust, and performance. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, 9(4) :269–275, apr 2014.
- [90] Garrett Wilson and Diane J. Cook. A survey of unsupervised deep domain adaptation. *ACM Transactions on Intelligent Systems and Technology*, 11(5) :1–46, sep 2020.
- [91] Kaichao You, Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I. Jordan. Universal domain adaptation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2019.
- [92] Rui Guo, Babajide Ayinde, Hao Sun, Haritha Muralidharan, and Kentaro Oguchi. Monocular depth estimation using synthetic images with shadow removal. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, oct 2019.
- [93] Hugues Hoppe, Tony DeRose, Tom Duchamp, John McDonald, and Werner Stuetzle. Mesh optimization. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques - SIGGRAPH '93*. ACM Press, 1993.
- [94] Radu Bogdan Rusu and Steve Cousins. 3d is here : Point cloud library (PCL). In *2011 IEEE International Conference on Robotics and Automation*. IEEE, may 2011.
- [95] Claudine Badue, Rânik Guidolini, Raphael Vivacqua Carneiro, Pedro Azevedo, Vinicius B. Cardoso, Avelino Forechi, Luan Jesus, Rodrigo Berriel, Thiago M. Paixão, Filipe Mutz, Lucas de Paula Veronese, Thiago Oliveira-Santos, and Alberto F. De Souza. Self-driving cars : A survey. *Expert Systems with Applications*, 165 :113816, mar 2021.
- [96] Liang Zhang, Leqi Wei, Peiyi Shen, Wei Wei, Guangming Zhu, and Juan Song. Semantic SLAM based on object detection and improved octomap. *IEEE Access*, 6 :75545–75559, 2018.
- [97] P. Stepan, M. Kulich, and L. Preucil. Robust data fusion with occupancy grid. *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, 35(1) :106–115, feb 2005.
- [98] Jing Chen and Shaojie Shen. Improving octree-based occupancy maps using environment sparsity with application to aerial robot navigation. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, may 2017.
- [99] Y. Xu, Z. Sun, R. Boerner, T. Koch, L. Hoegner, and U. Stilla. Generation of ground truth datasets for the analysis of 3d point clouds in urban scenes acquired via different sensors. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-3 :2009–2015, apr 2018.
- [100] Michael Gschwandtner, Roland Kwitt, Andreas Uhl, and Wolfgang Pree. BlenSor : Blender sensor simulation toolbox. In *Advances in Visual Computing*, pages 199–208. Springer Berlin Heidelberg, 2011.
- [101] Gilberto Echeverria, Nicolas Lassabe, Arnaud Degroote, and Severin Lemaignan. Modular open robots simulation engine : MORSE. In *2011 IEEE International Conference on Robotics and Automation*. IEEE, may 2011.

- [102] N. Koenig and A. Howard. Design and use paradigms for gazebo, an open-source multi-robot simulator. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*. IEEE, 2004.
- [103] Olivier Michel. Webots : Symbiosis between virtual and real mobile robots. In *Virtual Worlds*, pages 254–263. Springer Berlin Heidelberg, 1998.
- [104] Guido Rossum. *An introduction to Python : release 2.2.2*. Network Theory Ltd, Bristol, 2003.
- [105] E. Grilli, F. Menna, and F. Remondino. A review of point clouds segmentation and classification algorithms. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2/W3 :339–344, feb 2017.
- [106] Anh Nguyen and Bac Le. 3d point cloud segmentation : A survey. In *6th IEEE Conference on Robotics, Automation and Mechatronics (RAM)*. IEEE, nov 2013.
- [107] K. Klasing, D. Althoff, D. Wollherr, and M. Buss. Comparison of surface normal estimation methods for range sensing applications. In *2009 IEEE International Conference on Robotics and Automation*. IEEE, may 2009.
- [108] Eugene F. Krause. *Taxicab Geometry : An Adventure in Non-Euclidean Geometry*. DOVER PUBN INC, January 1987.
- [109] Leo Liberti, Carlile Lavor, Nelson Maculan, and Antonio Mucherino. Euclidean distance geometry and applications. May 2012.
- [110] Leif Peterson. K-nearest neighbor. *Scholarpedia*, 4(2) :1883, 2009.
- [111] Azriel Rosenfeld and Avinash C. Kak. *Digital Picture Processing*. Elsevier, 1982.
- [112] Michael Kampffmeyer, Nanqing Dong, Xiaodan Liang, Yujia Zhang, and Eric P. Xing. ConnNet : A long-range relation-aware pixel-connectivity network for salient segmentation. *IEEE Transactions on Image Processing*, 28(5) :2518–2529, may 2019.
- [113] Florent Tixier, Mathieu Hatt, Catherine Cheze Le Rest, Adrien Le Pogam, Laurent Corcos, and Dimitris Visvikis. Reproducibility of tumor uptake heterogeneity characterization through textural feature analysis in 18f-FDG PET. *Journal of Nuclear Medicine*, 53(5) :693–700, mar 2012.
- [114] Bruno Dutailly, Helene Coqueugnot, Pascal Desbarats, Stefka Gueorguieva, and Remi Synave. 3d surface reconstruction using HMH algorithm. In *2009 16th IEEE International Conference on Image Processing (ICIP)*. IEEE, nov 2009.
- [115] PC Mahalanobis. On the generalized distance in statistics. In *Proceedings of the National Institute of Science, India*, 1936.
- [116] Sunando Sengupta and Paul Sturgess. Semantic octree : Unifying recognition, reconstruction and representation via an octree constrained higher order MRF. In *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, may 2015.
- [117] D. Lang, S. Friedmann, and D. Paulus. Semantic 3d octree maps based on conditional random fields. In *IAPR International Conference on Machine Vision Applications*, Kyoto, JAPAN, May 2013.
- [118] Byung-Soo Kim, Pushmeet Kohli, and Silvio Savarese. 3d scene understanding by voxel-CRF. In *IEEE International Conference on Computer Vision*. IEEE, dec 2013.
- [119] Ni Zhu. *GNSS Propagation Channel Modeling in Constrained Environments : Contribution to the Improvement of the Geolocation Service Quality*. PhD thesis, Université de Lille, 2018.
- [120] Tixiao Shan and Brendan Englot. LeGO-LOAM : Lightweight and ground-optimized lidar odometry and mapping on variable terrain. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, oct 2018.

- [121] Héber Sobreira, Carlos M. Costa, Ivo Sousa, Luis Rocha, José Lima, P. C. M. A. Farias, Paulo Costa, and A. Paulo Moreira. Map-matching algorithms for robot self-localization : A comparison between perfect match, iterative closest point and normal distributions transform. *Journal of Intelligent & Robotic Systems*, 93(3-4) :533–546, jan 2018.
- [122] E.B. Olson. Real-time correlative scan matching. In *IEEE International Conference on Robotics and Automation*. IEEE, may 2009.
- [123] R.B. Rusu, N. Blodow, Z.C. Marton, and M. Beetz. Aligning point cloud views using persistent feature histograms. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, sep 2008.
- [124] Joscha Fossel, Karl Tuyls, Benjamin Schnieders, Daniel Claes, and Daniel Hennes. NOctoS-LAM : Fast octree surface normal mapping and registration. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, sep 2017.
- [125] Michiel Vlamincx, Hiep Luong, and Wilfried Philips. Multi-resolution ICP for the efficient registration of point clouds based on octrees. In *Fifteenth IAPR International Conference on Machine Vision Applications (MVA)*. IEEE, may 2017.
- [126] Yuenan Hou, Zheng Ma, Chunxiao Liu, and Chen Change Loy. Learning lightweight lane detection cnns by self attention distillation. August 2019.
- [127] Zhe Chen, Jing Zhang, and Dacheng Tao. Progressive LiDAR adaptation for road detection. *IEEE/CAA Journal of Automatica Sinica*, 6(3) :693–702, may 2019.
- [128] Fan Chao, Song Yu-Pei, and Jiao Ya-Jie. Multi-lane detection based on deep convolutional neural network. *IEEE Access*, 7 :150833–150841, 2019.
- [129] Sergio Rodriguez, Vincent Fremont, Philippe Bonnifait, and Veronique Cherfaoui. An embedded multi-modal system for object localization and tracking. In *IEEE Intelligent Vehicles Symposium*. IEEE, jun 2010.
- [130] Igor Bogoslavskyi and Cyrill Stachniss. Efficient online segmentation for sparse 3d laser scans. *PFG – Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, 85(1) :41–52, feb 2017.
- [131] Anna Clarke, Eyal Bagon, Igor Tubis, Gideon Stein, Amnon Shashualtai, Ben Shalom, Yoav Taieb, Shai Shalev, and David Huberman. Systems and methods for lane end recognition, 2015.

**Titre :** Evaluation de systèmes d'aide à la conduite. Génération automatique de vérité terrain augmentée à partir d'un capteur haute résolution et d'une cartographie sémantique et 3D ; Evaluation de fonctions de perception tierces.

**Mots clés :** robotique, perception, ADAS, évaluation, caractérisation, véhicules autonomes

**Résumé :** Le véhicule autonome représente l'un des défis technologiques actuels majeurs dans le milieu automobile. Les véhicules actuels se complexifient et intègrent de nouveaux systèmes reposant sur des fonctionnalités clés telles que la perception.

Permettant au véhicule d'appréhender l'environnement dans lequel il évolue, elle est exploitée sous différents aspects pour garantir une mobilité plus sûre. Étant donné le rôle essentiel de la perception dans le bon comportement d'un véhicule autonome, il est nécessaire de s'assurer que les solutions de perception utilisées soient suffisamment performantes pour garantir une circulation sécurisée. L'évaluation de telles solutions de perception reste cependant une tâche complexe et peu explorée.

L'un des points critiques est la difficulté de produire et de disposer de données de référence suffisantes pour mener des évaluations pertinentes. L'objectif de cette thèse est de mettre au point un nouvel outil de validation permettant d'évaluer les performances et niveaux d'erreurs de différentes solutions de perception, tout en utilisant le minimum de traitements manuels.

Via cet outil, il sera alors possible de mettre en concurrence différentes solutions en se basant sur des critères communs. La mise au point de cet outil se décompose selon deux parties principales qui sont : la génération automatisée de données de référence et la méthode d'évaluation des solutions de perceptions testées.

**Title :** Evaluation of driver assistance systems. Automatic generation of augmented ground truth from a high-resolution sensor and semantic and 3D mapping; Evaluation of third-party perception functions.

**Keywords :** robotics, perception, ADAS, evaluation, performances, autonomous driving

**Abstract :** Autonomous driving is one of the current major technological challenges in the automotive sector. Vehicles are becoming more complex and are integrating new systems relying on key functionalities such as perception.

Perception is used in various ways to ensure safer mobility, allowing the main inboard system to understand the environment in which the vehicle evolves. Perception plays a critical role in the proper behavior of an autonomous vehicle. It is necessary to ensure that the embedded perception solutions are effective enough to meet safe driving requirements. However, the evaluation of such solutions remains a complex and little explored task.

One of the critical issues is the difficulty of generating and having sufficient reference data to conduct relevant evaluations. The purpose of this thesis is to develop a new validation tool to evaluate the performances and error levels of different perception solutions, while minimizing the manual annotations.

With this tool, it will be possible to lead benchmark studies on different solutions based on common criteria. The development of this tool is split into two main parts: the automated production of reference data and the evaluation method of the tested perception solutions.