

# Reconstruction Planaire de Zone Carrossable de Route

Laurent DELOBEL  
Institut PASCAL

Université Clermont-Auvergne  
Aubière, France  
laurent.delobel@uca.fr

Thierry CHATEAU  
Institut PASCAL

Université Clermont-Auvergne  
Aubière, France  
thierry.chateau@uca.fr

Eric ROYER  
Institut PASCAL

Université Clermont-Auvergne  
Aubière, France  
eric.royer@uca.fr

François MARMOITON  
Institut PASCAL

Université Clermont-Auvergne  
Aubière, France  
francois.marmoiton@uca.fr

**Abstract**—Nous présentons une méthode d’estimation d’un modèle 3D planaire de la route à partir de capteurs de vision embarqués sur un véhicule. L’objectif est d’aller vers une automatisation de la détermination de l’espace navigable 3D dans les applications de véhicules autonomes. Nous proposons les contributions suivantes : 1) une combinaison de modèles géométriques et d’analyse vidéo par réseaux de neurones profonds pour extraire un ensemble de points épars et les filtrer pour conserver ceux dont le label sémantique présente une forte probabilité d’appartenir à la route, et 2) une base de données de type position de route géo-référencée (OpenStreetMap) servant de réalité terrain pour une première validation de la méthode.

**Index Terms**—SLAM, Segmentation sémantique, SPPNet, reconstruction 3D

## I. INTRODUCTION

La conduite autonome est un champ de recherche particulièrement actif et qui nécessite, pour la plupart des applications, l’utilisation d’une carte définissant les voies navigables. Des approches comme [1] se décomposent en deux phases : 1) un premier passage est réalisé sur la trajectoire à rejouer. Une phase d’apprentissage construit alors un modèle 3D basé capteur et 2) un asservissement du véhicule le long de cette trajectoire à partir d’une localisation temps réelle dans la carte basé capteur. Cette stratégie garantit, dans l’hypothèse où le guidage possède une précision correcte, que la navigation s’opère sur une voie carrossable et autorisée (si la phase manuelle 1 respecte ces condition). L’ajout d’un module d’évitement d’obstacle impose de générer des trajectoires s’éloignant de celle apprise et nécessite donc de connaître les limites de la zone sur laquelle le véhicule peut naviguer.

Dans ce document, nous proposons une technique de reconstruction 2D/3D de la structure de la chaussée carrossable des routes. Ce modèle servirait alors de zone sur laquelle le véhicule peut naviguer. L’objectif à terme est également de développer un système, permettant à partir d’images acquises à partir de véhicules (autonomes ou non), de construire une carte géométrique et sémantique de l’environnement.

Les principales contributions sont :

- La combinaison d’un modèle géométrique avec un réseau de segmentation sémantique,
- La construction d’une base de données et une validation qualitative de la méthode proposée.

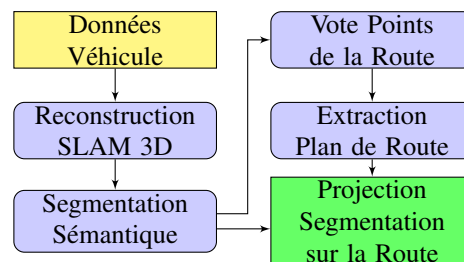


Fig. 1. Description de la méthode.

Le modèle géométrique est issu d’une technique de reconstruction de type SLAM visuel. Celle-ci permet de reconstruire un modèle 3D avec un facteur d’échelle proche de 1.

Après cette introduction, La deuxième section dresse un état de l’art de la problématique abordée. La méthode proposée est présentée dans une troisième section. La quatrième section décrit la base de données acquise et les expérimentations réalisées pour quantifier les performances de la méthode.

## II. TRAVAUX ANTÉRIEURS

Historiquement, les premiers travaux liés à la détection de bords de route utilisent des capteurs LIDAR. Certains travaux utilisent un capteur LIDAR mononappe plan 2D, que l’on positionne dans un plan vertical, ce dernier pivotant sur un axe vertical. Utilisant un tel montage, on parvient à identifier les murs verticaux ainsi que le sol.

Plus récemment, des techniques opérant des LIDARS 3D multi-nappes ont été utilisées afin d’extraire ces informations ([2], [3], [4]). Plusieurs nappes LIDAR orientées par élévation croissante permettent d’identifier, soit par rayon d’impact croissant, soit par azimuth croissant (en valeur absolue par rapport au cap véhicule), la position des bords de chaussée. L’élévation  $z$  varie en effet de manière discrète dans ce cas, entre la zone chaussée et la zone trottoir. Des couplages avec des données LIDAR géométriques, et vision (réflectance LIDAR, ou image caméra) ont également été réalisés ([4], [5]).

Le couplage de segmentation sémantique utilisant des images couleur, avec des données géométriques permettrait de généraliser ce procédé. En effet, les méthodes paramétriques géométriques LIDAR peuvent échouer dans certains cas :

- en cas de "bateau" (la hauteur trottoir rejoint celle de la route à la limite)
- dans le cas où le trottoir est matérialisé par un tracé au sol (marquage au sol)
- lorsque la route est sinueuse, ou que sa planéité est mise en défaut,
- en cas d'accotement.

Le cas des voies piétonnes est également à traiter, dans des cas exceptionnels il s'entend.

Les méthodes géométriques peinent car ces environnements ne sont pas suffisamment structurés. La segmentation sémantique d'images par des techniques d'apprentissage permet de mieux prendre en compte ces situations, dans le sens où des modèles complexes peuvent être entraînés. Il s'agit alors d'avoir à disposition une base de données annotée expliquant le plus de variabilité possible.

Il existe de nombreux travaux en segmentation sémantique basée apprentissage et utilisant des jeux de données issues du domaine automobile ([6], [7], [8]). Ces méthodes sont principalement basées sur des structures de réseaux de neurones profonds convolutifs (DCNN).

### III. MÉTHODE PROPOSÉE

Nous proposons d'utiliser conjointement des modèles géométriques de reconstruction d'environnement 3D et de navigation basé caméra, et des modèle basés apprentissage de segmentation sémantique d'images couleur, dans le but d'identifier les bords de voies de circulation pour des véhicules autonomes.

La chaîne de traitement se décompose en quatre parties :

- reconstruction d'une carte éparsée de points 3D à l'échelle à partir d'un SLAM visuel (multi-caméras) aidé par l'utilisation d'odomètres,
- segmentation sémantique des images afin d'extraire l'ensemble des *pixels* appartenant aux zones de route,
- identification robuste, pour chaque image, du plan 3D **local** de la route,
- reprojexion sur le plan de la route des zones d'image labellisées ROUTE.

Le but est d'obtenir un modèle 3D de la route sous forme d'un ensemble de polygones en 3 dimensions. Ces polygones 3D pourront avoir des plans, des normales associées différentes et ne représenteront qu'une autre manière de reconstruire la route. Cette manière de reconstruire est de plus haute sémantique en ce sens que les primitives sont de plus haute dimensionalité.

#### A. Construction d'une carte éparsée basée SLAM

Nous utilisons la méthode de reconstruction visuelle 3D basée sur caméras décrite dans ([9], [10]), et qui reconstruit, à partir d'une ou plusieurs caméras calibrées une carte 3D de l'environnement. Les odomètres du véhicule sont également pris en compte.

La carte ainsi générée contient:

- un raffinement des paramètres de calibration intrinsèque et extrinsèque des caméras utilisées,

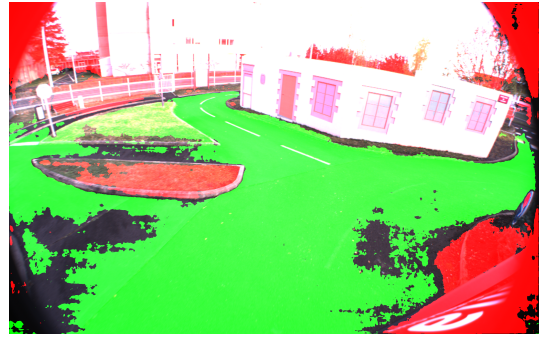


Fig. 2. La zone segmentée route (en vert), et les autres catégories (rouge). Noter entre les deux zones l'existence d'un glacis (identification indéterminée).

- la liste des points 3D identifiés dans un repère lié à la pose courante,
- pour chaque pose/image clé:
  - la pose de chaque caméra,
  - pour chaque caméra et pose associée, la liste des points 3D vus, ainsi que leurs coordonnées 2D dans l'image.

#### B. Segmentation Sémantique

Nous utilisons la segmentation sémantique décrite dans [6], avec un réseau de type **SSPNet**. Les paramètres utilisés sont les paramètres officiels TensorFlow publiés pour le dataset **CityScape**.

Comme nous utiliserons ce réseau sur une base différente de celle sur lequel il a été entraîné, les performances seront moindres à cause du changement de domaine. Nous proposons d'introduire une notion d'incertitude liée aux mesures issues de la segmentation afin de définir une notion de facteur de sécurité lié à la classe associée à un pixel. On définit pour chaque pixel de l'image:

$$\text{OutClass} = \arg \max_{\text{class } i} \text{ResultVector}[i] \quad (1)$$

Nous avons préféré prendre une marge de sécurité sur le résultat de retour, en exigeant que pour la classe choisie (route), la différence entre sa réponse et le maximum des autres classes soit supérieur à une valeur positive, que nous nommons  $L$ .

$$\text{OutClass} = \text{Road ssi}$$

$$\arg \max_{\text{class } i \neq \text{Road}} \text{ResultVector}[i] + L \leq \text{ResultVector}[\text{Road}] \quad (2)$$

Dans ce cas, on prend une marge de sécurité en terme de réponse pour éviter toutes les zones dans lesquelles le maximum de réponses des classes est mal défini. Cela permet d'éviter d'utiliser des portions de l'image dans lesquelles la catégorisation des pixels n'est pas certaine.

Un exemple d'une segmentation prenant en compte un facteur de sécurité est visible sur la figure 2.

### C. Estimation du Plan Médian de la Route

Les données contenues dans la carte générée sont une liste de points en 3 dimensions. Afin de définir la projection de la zone segmentée, il faut définir une portion localement plane associée à la route. Certains a-prioris ont été pris:

- la route est, sur une échelle de quelques mètres localement plane,
- le véhicule circule toujours sur la route,
- les caméras sont toujours capables d'observer la route.

Dans ce cas, nous proposons de fitter un plan sur un patch de quelques mètres autour de la position de passage d'un véhicule. Ce plan est ensuite utilisé pour filtrer des points du patch, et conserver uniquement ceux qui appartiennent à la route. Il suffit alors d'estimer, par une méthode robuste, le plan qui passe au mieux par ces points. Nous proposons un fit par moindres carrés médians, en utilisant une technique de tirage aléatoire (du type RANSAC).

Afin de ne prendre pour la détermination du plan de la route que des points 3D ayant une probabilité raisonnable d'y appartenir, un filtrage par vote est effectué:

- pour chaque image 2D couleur (capture), on identifie la zone vue comme route,
- pour chacun des points appartenant à la capture (voir figure 3):
  - si les coordonnées 2D du points capturé sont incluses dans la zone segmentée route de l'image caméra, on incrémente le vote route ainsi que le nombre de vues,
  - dans le cas contraire, on incrémente le nombre de vues du point, en laissant inchangé son vote
- lorsque toutes les vues caméra ont été analysées, chaque point est catégorisé:
  - lorsque le nombre de vote route dépasse 50% du nombre de vues du point, celui-ci est catégorisé "ROUTE" (points verts sur la figure 4),
  - dans le cas contraire, il est catégorisé "AUTRE" (points rouges sur la figure 4).

Chaque zone segmentée comme zone de route à partir de l'image caméra couleur est par la suite gérée comme un polygone dont les sommets sont extraits par la procédure de segmentation. Ces polygones sont bidimensionnels (dans le plan focal caméra).

Ensuite, pour chaque pose véhicule, on interpole le plan de route locale ainsi:

- un patch de  $D \times D$  mètres autour de la position du véhicule est sélectionné,
- les points catégorisés route et appartenant à ce patch sont utilisés,
- un fit moindres-carrés-médians randomisé sur l'équation du **plan de la route** (un point du plan, barycentre par exemple, plus une normale unitaire) est réalisé basés sur ces points; le nombre de tirages aléatoires de séquences de points utilisé est donné comme N (voir [11]).

### D. Projection de la zone Route

À cette étape, nous disposons:

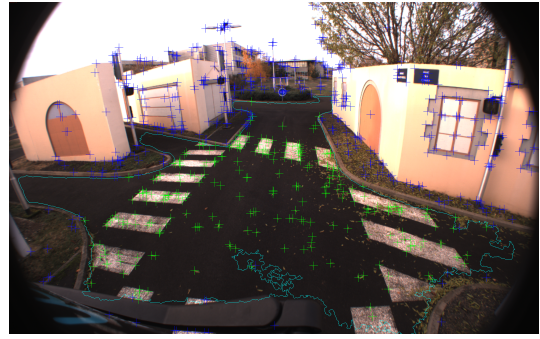


Fig. 3. Image des points identifiés comme appartenant à la route (croix vertes) à partir de la vue caméra. Les points bleu n'y appartiennent pas.

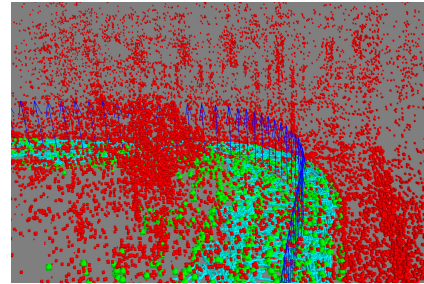


Fig. 4. Points 3D (en vert) à vote majoritaire comme route. Les points rejetés sont en rouge. Les polygones de route en 3D sont dessinés en cyan. Le trièdre route (axe z en couleur bleue) correspond au passage du véhicule et constitue la trajectoire au sol de ce dernier.

- des patches locaux des plans de route,
- des zones reconnues comme segmentées (dans les images)

Le but de cette étape consiste à apparier une vue caméra avec un plan de route suffisamment proche, observable directement. L'idée est alors de choisir une correspondance entre chaque vue caméra dont on a extrait la zone segmentée comme route, et un plan de route correspondant. Puis, une fois cette correspondance trouvée, on projette le polygone route segmenté sur ce plan par un procédé de type caméra projective.

Étant donné que nous avons équipé notre véhicule avec une paire de caméras stéréo à l'avant du véhicule (le véhicule se déplaçant vers l'avant), nous allons commencer à partir d'une position antérieure et avancer au fur et à mesure jusqu'à trouver le centre du patch route le plus proche du centre optique des caméras.

Le polygone entourant la zone délimitée comme route dans l'image couleur est alors projeté par un procédé de type projection centrale par rapport à la position caméra, sur le plan de route identifié. Afin de ne pas créer de points projetés vers l'arrière de la caméra (ce qui est typique d'un modèle projectif au-delà de la ligne d'horizon du plan route dans l'image), nous limitons en position Y (haut) l'extension du polygone de route reconstruit. On peut observer sur la figure 5 le résultat d'une telle reconstruction.

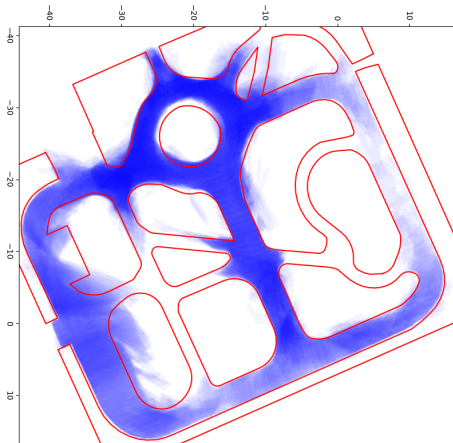


Fig. 5. Superposition des polygones segmentés comme route (en bleu), avec le bords des trottoirs (en rouge) en vue de dessus. Carte orientée NORD vers le haut.

### E. Accumulation des Polygones en Vue 2D

Les polygones reconstruits permettent une visualisation 3D de la carte ainsi que de la route reconstruite. Une vue plus intéressante consiste à accumuler les polygones de bord de route en vue 2D.

Nous utilisons comme repère de référence le repère de départ du véhicule à l'initialisation. Nous amenons chaque polygone 3D dans le repère en question, puis le projetons sur le plan XY. L'affichage des polygones est réalisé en accumulation (plus le nombre de polygones est important, plus la couleur sera marquée).

## IV. EXPÉRIMENTATIONS

Dans cette section, nous présentons les résultats obtenus dans le cadre de la reconstruction sémantique de bord de route.

### A. Équipements et Données Utilisées

Les expérimentations ont été réalisées à partir de données acquises sur le site d'expérimentation PAVIN<sup>1</sup>, en utilisant un enregistrement des données capteur à bord d'un véhicule EZ10 (voir figure 6). Le véhicule EZ10 est équipé entre autres capteurs:

- de deux caméras Noir et blanc de résolution  $640 \times 480$  (une à l'avant, l'autre à l'arrière du véhicule) opérant à une fréquence de 10 Hz,
- de deux caméras couleur (type Bayer) de résolution  $1920 \times 1200$ , opérant également à 10 Hz (mais non-synchronisées avec les précédentes),
- de capteurs d'odométrie intégrés au véhicule (bus CAN),
- d'un capteur de type GPS (UBLOX6T).

Ce véhicule utilise l'intergiciel **Robot Operating System** (ROS), commun en robotique. Une des nombreuses fonctionnalités de ce type d'environnement est de permettre l'enregistrement des données capteurs sous un format universel

<sup>1</sup>Plateforme d'Auvergne pour les Véhicules Intelligents, simulant un environnement urbain à échelle 0.5.



Fig. 6. Véhicule expérimental EZ10. Deux caméras global-shutter couleur de type GS3-U3-23S6C, d'une résolution 1920 par 1200 pixels sont montées en paire stéréo sur le toit à l'avant du véhicule.

permettant leur ré-utilisation par toute application utilisant l'environnement ROS.

Les données acquises par les capteurs ont été stockées dans un fichier de type **ROSbag** permettant la réutilisation des données. Nous avons travaillé à partir d'un fichier **ROSbag** issu d'une trajectoire en S, comprenant des images de vues stéréo couleur 24 bits au format  $1920 \times 1200$ , synchronisées à 10 ms et acquises à une fréquence de 10 Hz. La durée de la trajectoire est de 3 minutes.

Des acquisitions LIDAR 3D partielles du site d'expérimentation ont été acquises, mais **n'ont actuellement pas encore été exploitées** (travaux futurs). Celles-ci contiennent des nuages de points 3D au format Lambert-RGF93 munis d'une altitude. Nos analyses n'ont exploité pour l'instant que l'information visuelle.

### B. Mode Opérateur

Le mode opératoire appliqué est le suivant:

- calibration de la paire caméra stéréo couleur par SLAM sur une trajectoire fermée et auto-étalonnage après une fermeture de boucle ([12]),
- extraction des images couleurs (fichier **ROSbag**) et segmentation sémantique,
- exécution de notre analyse de reconstruction de bords de voie (carte 3D) à partir des données précédemment générées,
- aplanissement 2D horizontal et extraction des bords de route du jeu de données.

La calibration de la paire stéréo avant du jeu de données a été réalisée itérativement par optimisation globale répétée, et utilise le même jeu de données que la reconstruction. Les paramètres du SLAM sont réglés pour avoir un grand nombre de points d'intérêt (5 fois plus que ce que nous utilisons habituellement pour la navigation autonome) de façon à avoir suffisamment de points détectés sur la route.

La trajectoire utilisée fait une longueur d'approximativement 200 mètres et génère un peu plus de 170 000 points dans la carte. Cette carte induit des dérives au niveau du parcours. Nous tentons de limiter celles-ci en utilisant des trajectoires re-bouclant sur elles-mêmes, et nous

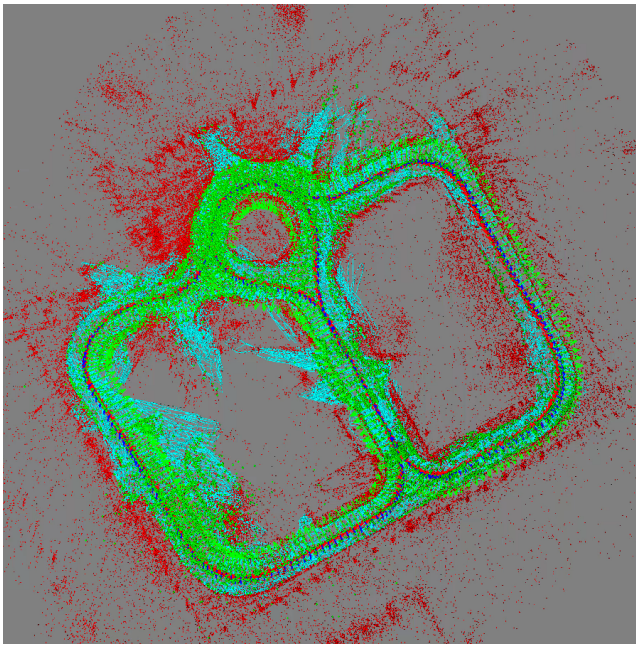


Fig. 7. Vue de dessus d'une reconstruction 3D de la route sur note site d'expérimentation PAVIN, carte orientée NORD vers le haut.

opérons une optimisation ultérieure (également hors-ligne) sur les poses compatibles (fermeture de boucle).

Les segmentations de zone de route ont été réalisées de manière déportée sur un cluster GPU de notre méso-centre de calcul **CRRI-UCA** et rapatriées en local.

### C. Résultats Obtenus

On peut voir sur la figure 7 la reconstruction des routes principales de note site d'expérimentation PAVIN. Ceci est à comparer à la carte de reconstruction bidimensionnelle 5.

Les trièdres au niveau de l'orientation de la route matérialisent le passage du véhicule au niveau de la route (vecteur Z normal vers le haut par rapport à la route, vecteur X dans le sens de parcours).

Nous avons manuellement aligné les données de la carte géométrique SLAM avec la carte des bords de trottoirs avec une erreur sur le facteur d'échelle de l'ordre de 1% .

Nous avons de même manuellement aligné nos données de polygones de zone route avec la même base de données, voir figure 5, et, hormis la reconstruction de la zone circulaire autour du rond-point, l'alignement (sans changement de facteur d'échelle) s'avère satisfaisant moyennant une tolérance de 1% également.

On remarque sur les figures 5 et 7:

- sur la portion OUEST du site reconstruit la segmentation sémantique déborde nettement sur la zone non-carrossable,
- la statistique de polygones de segmentation est faible sur la route côté EST (taux échecs de segmentation important des images droite et gauche simultanée).

Le premier point est assez aisément mis en évidence sur une image de segmentation opérée dans cette zone, présentée

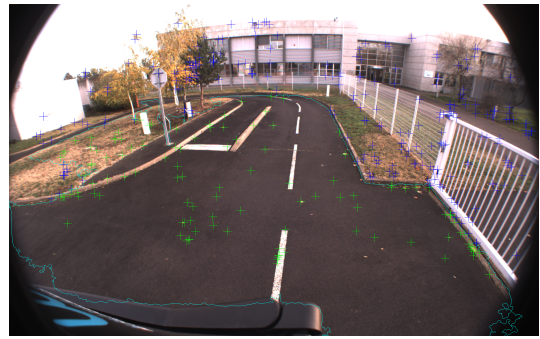


Fig. 8. Exemple d'une segmentation débordant au-delà du tracé de la route (sur le côté gauche de la route, remarquer les points marqués en vert sur le terre-plein).

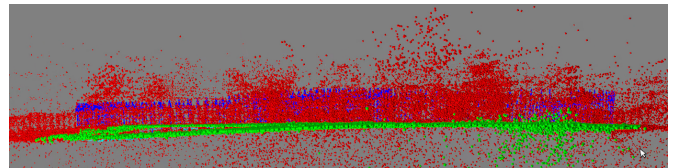


Fig. 9. Vue par la tranche de la pente descendante allant du rond-point vers l'entrée piéton (gauche de l'image).

sur la figure 8. Il s'agit d'un cas où la segmentation échoue. Il est possible que les échecs de ce type soient attribuables aux conditions d'éclairages de la scène tout comme le rendu colorimétrique sur les caméras. Un apprentissage spécifique permettrait sans doute d'améliorer ces résultats, tout comme le choix d'un autre réseau de segmentation.

Le second item, quant à lui, est relié à des résultats de segmentation inutilisables (la totalité de l'image classée dans une seule classe).

On notera que la technique que nous proposons est capable de reconstruire la structure 3D (incluant la pente et l'altitude) de la portion de route générée. Sur la figure 9, on observe, vue depuis le NORD, la structure EST-OUEST v.s. altitude de notre site d'expérimentation.

Une branche apparaît en pente descendante en allant vers la gauche de la vue (portion allant du rond-point vers l'EST, en pente descendante). Une autre (côté SUD du site, allant également vers l'EST), est quasiment horizontale. On est donc capable de reconstruire la déclivité d'une scène à partir de données visuelles acquises sur véhicule.

### D. Test Manuel sur des données KITTI

Afin de pouvoir valider notre extraction de zones carrossables, nous avons utilisé le dataset numéroté 00 du dataset **KITTY**, sous-base de données **ODOMETRY** [13].

La comparaison a été réalisée à partir de données géoréférencées de manière absolue dans la base de données OpenStreetMap [14].

Grâce aux images de ce jeu de données 00, nous avons identifié de manière manuelle la zone dans laquelle les données ont été extraites (il s'agit de Weinerfeld, dans la banlieue de Karlsruhe). Une extraction de la base de données Open-

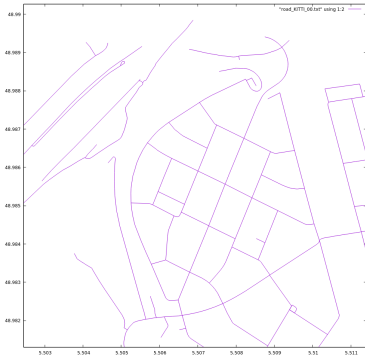


Fig. 10. Extraction des rues de Weinerfeld, dans la banlieue de Karlsruhe.

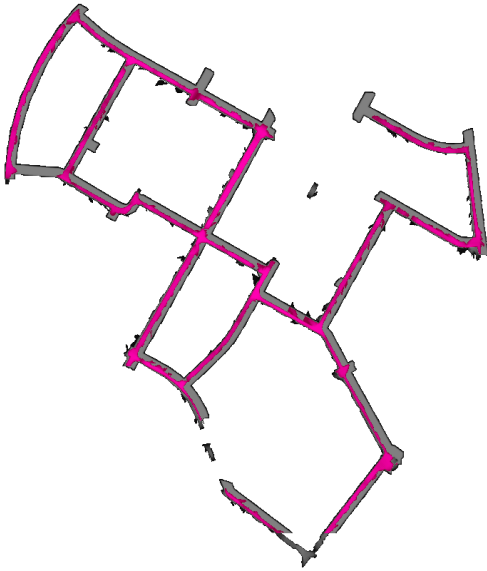


Fig. 11. Alignement des données entre la base de données absolue OpenStreetMap, et notre reconstruction de route (recouvrement en rose). Noter que les rues provenant de la base de données OSM sont fortement élargies afin de prendre en compte les potentielles erreurs locales.

StreetMap nous a permis de reconstruire les zones de route utilisées pour l’acquisition des données (voir la figure 10).

Une reconstruction de route a été menée sur un peu plus de la moitié du jeu de données. Sur une partie non-négligeable, la segmentation échoue, de ce fait, une partie d’une longue rue n’apparaît pas dans les données. Après avoir élargi les segments de rue obtenus d’OpenStreetMap, nous les avons alignés selon une transformée de Fourier-Mellin (voir [15]) avec la reconstruction réalisée. La superposition des deux zones est indiquée en vert sur la figure 11. Le facteur d’échelle est également correct à 1 % près.

#### CONCLUSION

Nous proposons une méthode multi-techniques permettant, à partir de données véhicule, d’une reconstruction géométrique, ainsi que d’une opération de segmentation sémantique basée sur un réseau convolutionnel, de reconstruire une zone carrossable de route. Certains bords de route peuvent être identifiés de manière correcte, en partie en raison de résultats erronés

de l’algorithme de segmentation choisi. Malgré les limites de l’algorithme de segmentation utilisé, nous pensons qu’avec l’amélioration de qualité et de temps de calcul de ces techniques, ces résultats sont à même d’être améliorés à l’avenir. Une validation par alignement sur des données 3D de type métrologie (acquisitions LIDAR mentionnées en IV-A) fait également partie des travaux futurs.

#### REMERCIEMENTS

Ces travaux ont été financés par le programme de recherche ”Investissements d’Avenir” et l’Agence Nationale de la Recherche, à travers le laboratoire d’excellence IMobS3 (ANR-10-LABX-16-01). Les calculs ont été effectués sur les ressources du Mésocentre Clermont Auvergne.

#### REFERENCES

- [1] E. Royer, M. Lhuillier, M. Dhome, and T. Chateau, “Localization in urban environments : monocular vision compared to a differential gps sensor,” in *IEEE CVPR, Computer Vision and Pattern Recognition*, (San Diego, USA), juin 2005.
- [2] F. Ghallabi, F. Nashashibi, G. El-Haj-Shhade, and M.-A. Mittet, “Lidar-based lane marking detection for vehicle positioning in an hd map,” in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 2209–2214, IEEE, 2018.
- [3] A. Y. Hata, F. S. Osorio, and D. F. Wolf, “Robust curb detection and vehicle localization in urban environments,” in *2014 IEEE Intelligent Vehicles Symposium Proceedings*, pp. 1257–1262, IEEE, 2014.
- [4] C. Fernández, D. F. Llorca, C. Stiller, and M. Sotelo, “Curvature-based curb detection method in urban environments using stereo and laser,” in *2015 IEEE Intelligent Vehicles Symposium (IV)*, pp. 579–584, IEEE, 2015.
- [5] A. Hata and D. Wolf, “Road marking detection using lidar reflective intensity data and its application to vehicle localization,” in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 584–589, IEEE, 2014.
- [6] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, “Simultaneous detection and segmentation,” in *European Conference on Computer Vision*, pp. 297–312, Springer, 2014.
- [7] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia, “Icnet for real-time semantic segmentation on high-resolution images,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 405–420, 2018.
- [8] R. P. Poudel, S. Liwicki, and R. Cipolla, “Fast-scnn: fast semantic segmentation network,” *arXiv preprint arXiv:1902.04502*, 2019.
- [9] E. Royer, J. Bom, M. Dhome, B. Thuilot, M. Lhuillier, and F. Marmoton, “Outdoor autonomous navigation using monocular vision,” in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1253–1258, IEEE, 2005.
- [10] E. Royer, *Cartographie 3D et localisation par vision monoculaire pour la navigation autonome d’un robot mobile*. PhD thesis, 2006.
- [11] H. Barreto and D. Maharry, “Least median of squares and regression through the origin,” *Computational statistics & data analysis*, vol. 50, no. 6, pp. 1391–1397, 2006.
- [12] P. Lébraly, E. Royer, O. Ait-Aider, C. Deymier, and M. Dhome, “Fast calibration of embedded non-overlapping cameras,” *2011 IEEE International Conference on Robotics and Automation*, pp. 221–227, 2011.
- [13] A. Geiger, P. Lenz, and R. Urtasun, “Are we ready for autonomous driving? the kitti vision benchmark suite,” in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.
- [14] OpenStreetMap contributors, “Planet dump retrieved from <https://planet.osm.org>.” <https://www.openstreetmap.org>, 2017.
- [15] O. Fourt, R. Rouveure, and P. Faure, “Odometry with a panoramic radar in outdoor environment: use of a radon-fourier transform for the estimation of the move without external sensors,” *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 7, no. 2, pp. 35–50, 2014.