Construction précise de bases d'amers géo-référencés pour la localisation d'un véhicule en milieu urbain

Dorra Larnaout[1], Vincent Gay-Bellile[1], Steve Bourgeois[1], Michel Dhome[2]

1. CEA, LIST, LVIC
Point Courrier 173, F-91191, Gif-Sur-Yvette, France

2. Institut Pascal, UMR 6602
Université Blaise Pascal/CNRS/IFMA

dorra.larnaout@gmail.com

RÉSUMÉ. *L'aide à la navigation par Réalité Augmentée RA nécessite une estimation précise des six degrés de liberté relative au déplacement de la caméra. Pour ceci, les solutions de localisation par vision passent souvent par une étape de modélisation hors ligne des grands environnements. Tandis que les solutions existantes exigent des matériels coûteux et/ou un temps d'exécution très important, nous proposons dans cet article un processus qui crée automatiquement une modélisation précise de l'environnement en utilisant uniquement une caméra standard, un GPS bas coût et des modèles SIG (Système d'Information Géographique) disponibles gratuitement.*

ABSTRACT. *To provide high quality Augmented Reality AR service, accurate 6DoF localization is required. To ensure such accuracy, most of current vision-based solutions rely on an off-line large scale modeling of the environment. While existing solutions require expensive equipments and/or a prohibitive computation time, we propose in this paper a complete framework that automatically builds an accurate city scale database of landmarks using only a standard camera, a GPS and GIS Geographic Information System.*

MOTS-CLÉS : *SLAM monoculaire, GPS, modèles SIG.*

KEYWORDS: *monocular SLAM, GPS, GIS models.*

DOI:10.3166/TS.32.147-167

1. Introduction

Afin d'offrir des services d'aide à la navigation par réalité augmentée RA, des solutions de localisation basées vision peuvent être utilisées (Arth *et al*., 2009), (Zamir, Shah, 2010)... Ces solutions nécessitent souvent la création préalable d'une base précise d'amers géo-référencés à grande échelle (Kaminsky *et al*., 2009), (Strecha *et al*., 2010), (Li *et al*., 2012). Cette base est utilisée par la suite pour assurer une localisation en ligne par mise en correspondance des amers de la base avec les primitives 2D extraites des images traitées en ligne. Pour créer la base en question, ces solutions exploitent généralement soit des matériels coûteux (LIDAR, GPS RTK,...) soit des collections d'images disponibles sur Internet ((Agarwal *et al*., 2011), (Wang *et al*., 2013), (Frahm *et al*., 2010)...), souvent concentrées sur des sites d'intérêt (tour Eiffel, le Colisée, ...). Or, pour qu'une base d'amers soit exploitable dans le cadre de la localisation de véhicule, celle-ci doit être précise et couvrir des points de vues semblables à ceux des utilisateurs finaux à savoir les conducteurs des véhicules.

Dans cet article, nous proposons une solution qui exploite uniquement un GPS standard, une caméra bas coût et des modèles SIG largement disponibles. Plus précisément, la base d'amers est créée à travers la fusion du SLAM visuel avec des données d'un GPS standard et celles d'un MNT simplifié (modèle numérique de terrain). La base résultante est par la suite améliorée en exploitant les contraintes géométriques fournies par le MNT et les modèles 3D des bâtiments. Cette solution étend les travaux de (Lothe *et al*., 2009) et de (Lanaout *et al*., 2013).

Les modèles 3D des bâtiments utilisés sont des modèles géométriques où chaque façade de bâtiment est représentée par un plan non texturé. Le MNT utilisé est également un modèle géométrique simplifié où chaque route est représentée par son axe principal comme le montre la figure 2. Pour faciliter l'utilisation du MNT, chaque route sera modélisée, dans la suite de cet article, par le plan déterminé à partir de son axe principal. Étant donné que l'information sur l'inclinaison des routes n'est pas fournie dans le MNT utilisé, nous supposons que le véhicule se déplace dans un monde plan par morceaux.

Dans ce qui suit, un état de l'art des méthodes existantes est présenté dans la section 2. Le positionnement de notre approche par rapport à celle de (Lothe *et al*., 2009) est exposé dans la section 3. Dans la section 4, la solution proposée est détaillée. Enfin, des évaluations sur des données réelles ainsi qu'un exemple d'application de RA sont présentés dans la section 5.

Notation. Dans cet article, deux repères différents sont utilisés : le repère monde correspondant au référentiel du GPS et le repère associé à chaque route du MNT où l'axe des X et l'axe des Y définissent le plan du sol tandis que l'axe des Z représente sa normale. Par exemple, si $\mathbf{q_1}$ est le vecteur contenant les coordonnées du point 3D Q_1 dans le repère monde, nous notons $\hat{\mathbf{q}}_\mathbf{1}$ le vecteur contenant ses coordonnées dans le repère de la route. Dans la suite, nous supposons que la reconstruction de la scène observée est estimée par le SLAM monoculaire introduit dans (Mouragnon *et al*., 2006). Il s'agit d'un SLAM basé sur le principe des images-clés et qui utilise un

ajustement de faisceaux local pour raffiner l'erreur de re-projection de M points 3D observés dans les N dernières images clés. En supposant que les distorsions optiques ont été corrigées a priori, la fonction de coût résultante est donnée par :

$$E_1(\mathbf{x}) = \sum_{i=1}^{i=M} \sum_{j \in \mathcal{A}_i} d^2\left(u_{i,j}, KP_j\mathbf{q_i}\right), \quad (1)$$

où K est la matrice des paramètres intrinsèques de la caméra et P_j est sa $j^{ème}$ pose. $u_{i,j}$ est l'observation 2D du $i^{ème}$ point 3D Q_i dans la $j^{ème}$ image clé. $\mathcal{A}_i$ est l'ensemble des indices des images clés observant Q_i. $\mathbf{x}$ représente le vecteur contenant les paramètres à optimiser. Ce vecteur est organisé tel que $\mathbf{x}^T = \left(\mathbf{c_1}^T, \mathbf{c_2}^T, \mathbf{q}^T\right)$ où $\mathbf{c_1} = \left(\mathbf{t_x}^T, \mathbf{t_y}^T, \psi^T\right)$ concatène les paramètres *dans le plan* de la caméra (*i.e* le déplacement 2D et l'angle lacet), $\mathbf{c_2} = \left(\mathbf{t_z}^T, \alpha^T, \gamma^T\right)$ concatène les paramètres *hors plan* (*i.e* l'altitude de la caméra, les angles roulis et tangage) tandis que $\mathbf{q}$ contient les positions 3D de tous les points observés.

Finalement, les mesures *dans le plan* fournies par le GPS sont respectivement stockées dans les vecteurs $\mathbf{t}_x^{gps}$ et $\mathbf{t}_y^{gps}$.

2. Travaux connexes

Plusieurs méthodes ont été proposées afin de créer des bases d'amers géo-référencés pour des grands environnements. Ces bases contiennent souvent un ensemble d'amers 3D que nous désignons par le terme "nuage de points" obtenu généralement par un algorithme de type *Structure from Motion* (SfM). Dans ce qui suit nous nous intéresserons en particulier aux méthodes qui géo-référencent un nuage de points 3D.

Pour assurer le géo-référencement de la base d'amers, certaines méthodes exploitent conjointement l'information du GPS disponible et la géométrie multivues. Par exemple, (Li *et al*., 2012), (Agarwal *et al*., 2011) et (Frahm *et al*., 2010) construisent une reconstruction SfM à partir d'une collection images géo-référencées disponibles sur Internet ou encore des images extraites de *Google Street View* (Wang *et al*., 2013). Toutefois, la précision de la géo-localisation obtenue reste limitée à celle du GPS qui se dégrade considérablement en ville à cause des phénomènes de canyon urbain (*ie* réflexion du signal GPS sur les façades de bâtiments).

Pour améliorer cette précision, d'autres méthodes proposent d'exploiter en plus les informations géométriques et géographiques apportées par des modèles SIG. Ceci est réalisé en estimant une transformation rigide qui recale au mieux la reconstruction 3D avec les empreintes de bâtiments obtenues soit à partir d'images satellite dans (Kaminsky *et al*., 2009) ou directement à partir d'une carte 2D des bâtiments (Strecha *et al*., 2010). Toutefois, une transformation rigide ne modélise pas les déformations complexes liées aux dérives sur la trajectoire inhérente aux algorithmes SfM.

Pour faire face à cette limitation, d'autres méthodes notamment (Wang *et al*., 2013) et (Lothe *et al*., 2009) proposent de recaler le nuage de points résultant de

la reconstruction SfM sur des modèles SIG via des transformations non rigides. Par exemple, (Lothe *et al*., 2009) introduit un nouvel ajustement de faisceaux intégrant les contraintes géométriques apportées par les modèles 3D des bâtiments. Il en résulte une géo-localisation hors ligne plus précise que celle obtenue par une simple transformation rigide. Cette solution semble mieux répondre à notre problématique. Pour cette raison, nous adoptons le même concept et nous l'étendons afin de traiter ses limitations qui seront détaillées ci-dessous.

3. Positionnement

Dans (Lothe *et al*., 2009), la possibilité de corriger et géo-référencer une reconstruction SLAM en exploitant des modèles 3D des bâtiments grossiers à travers des transformations non-rigides a été démontrée. Pour ceci, ils proposent un processus se déroulant en deux étapes : une première correction grossière de la reconstruction SLAM suivie par une optimisation plus fine.

Concernant la première étape, son objectif est de fournir une reconstruction SLAM géo-référencée avec peu de dérives. Pour ceci, (Lothe *et al*., 2009) considèrent que la dérive du SLAM apparaît principalement au niveau des virages. Sous cette hypothèse, ils proposent de modéliser la dérive en question à travers des similitudes par morceaux avec contraintes de jointure aux extrémités. Chaque morceau est défini par la sous reconstruction 3D résultante du déplacement de la caméra le long d'une même rue. Par conséquent, l'ensemble des poses de la caméra appartenant à une même sous reconstruction subissent la même correction. Les poses de la caméra aux extrémités, correspondant aux virages et donc appartenant quant à elles à deux sous reconstructions, doivent respecter la transformation associée à chacun des deux morceaux. Ceci garantit que la trajectoire SLAM reste unie. Pour estimer les transformations en question, les données GPS au niveau des virages sont exploitées.

Une fois le premier recalage grossier terminé, la reconstruction résultante est raffinée à l'aide d'un algorithme d'ICP non rigide. Au cours de cette étape, l'hypothèse de dérive uniquement au niveau des virages n'est pas remise en cause. Toutefois, des nouvelles données plus précises que le GPS, notamment les modèles 3D des bâtiments, sont exploitées afin de raffiner la reconstruction SLAM résultante. Ceci est réalisé en minimisant la distance 3D entre le nuage de points corrigé et les modèles 3D de la ville. La reconstruction SLAM n'incluant pas uniquement des points appartenant aux façades, une étape de segmentation du nuage de points est utilisée pour identifier l' ensemble de points 3D concerné par l'ICP non rigide. Cette segmentation est basée sur un simple lancer de rayon, ainsi si le rayon reliant le centre optique de la caméra et le point 3D intersecte le modèle 3D des bâtiments alors le point en question est considéré comme appartenant à une façade d'un bâtiment. A l'issue de cette segmentation, seuls les points appartenant à des façades de bâtiments sont utilisés durant l'ICP non rigide.

Afin d'accroître la précision de la reconstruction, une seconde étape d'optimisation reposant sur un ajustement de faisceaux global contraint aux bâtiments est utilisée.

Cette seconde étape permet de remettre en cause l'hypothèse de dérive principalement au niveau des virages. Par ailleurs, cette étape favorise un respect de la contrainte multi-vues sur l'ensemble de la trajectoire, y compris au niveau des virages ce qui n'était pas le cas pour l'ICP non rigide. Seuls les points identifiés comme appartenant à une façade de bâtiment sont utilisés au cours du raffinement.

Malgré les résultats prometteurs, cette solution présente certaines limitations. En effet, son étape d'initialisation sous-exploite les données GPS (uniquement aux virages) et ne traite pas les variations d'altitude ce qui entraîne une reconstruction initiale peu précise pouvant perturber les étapes de raffinement à l'aide des modèles de bâtiments (*ie* l'ICP non rigide suivi de l'ajustement de faisceaux global). De plus, utiliser uniquement des points associés aux modèles des bâtiments rend cette solution limitée aux zones urbaines denses. En effet, quand peu de bâtiments sont observables, l'ensemble de points associés aux façades des bâtiments et donc participant aux étapes de raffinement peut ne pas être suffisant pour contraindre convenablement la reconstruction SLAM. Par ailleurs, durant l'optimisation, tous les degrés de liberté sont raffinés tandis que les bâtiments contraignent principalement les degrés de liberté *dans le plan*. Par conséquent, les paramètres *hors plan* (principalement l'altitude et l'angle tangage) peuvent être détériorés ceci est d'autant plus vrai si la qualité de l'initialisation est mauvaise comme nous le démontrerons dans la partie résultat (section 5.3.3).

Pour résoudre ces problèmes, plusieurs améliorations sont introduites dans cet article :

– **Au niveau de l'initialisation :** nous proposons de tirer plus profit du GPS afin d'améliorer l'estimation des degrés de liberté *dans le plan* au cours de l'initialisation. De plus, pour apporter plus de précision sur les degrés de liberté *hors plan*, le MNT est également exploité.

– **Au niveau du raffinement :** dans le but d'étendre la solution de (Lothe *et al*., 2009) aux milieux péri-urbains et garantir plus de robustesse quand peu de bâtiments sont observables, nous proposons de prendre en compte à la fois les contraintes géométriques des points 3D associés aux modèles des bâtiments et les contraintes multivues fournies par l'ensemble de points 3D représentant le reste de l'environnement. Par ailleurs, pour apporter plus de précision à l'estimation des degrés de liberté *hors plan*, nous exploitons également les contraintes fournies par le MNT. Étant donné qu'il n'est pas trivial de fusionner simultanément toutes ces contraintes sans perturber la convergence du processus de raffinement, nous proposons une solution qui se focalise dans un premier temps sur une correction grossière de la dérive du SLAM avant d'optimiser plus finement la base d'amers résultante.

4. Solution proposée

Pour permettre la création automatique d'une base d'amers géo-référencés précise, notre solution comprend trois étapes que nous allons détailler ci-dessous.

1. Une reconstruction initiale est obtenue en fusionnant le SLAM visuel introduit dans (Mouragnon *et al*., 2006) avec les données fournies par le GPS et le MNT (section 4.1).

2. Les contraintes apportées par les modèles des bâtiments sont utilisées pour raffiner le nuage de points et les degrés de liberté *dans le plan* de la caméra via un ajustement de faisceaux global contraint aux modèles 3D des bâtiments (section 4.2).

3. Un dernier raffinement est effectué sur tous les degrés de liberté de la reconstruction en appliquant un deuxième ajustement de faisceaux global contraint aux modèles 3D des bâtiments et le MNT (section 4.3).

4.1. Reconstruction initiale

Un SLAM monoculaire basé sur un ajustement de faisceaux local (Mouragnon *et al*., 2006) fournit en ligne une représentation 3D de l'environnement. Cependant, la reconstruction obtenue n'est pas géo-référencée et souffre souvent de dérive de facteur d'échelle et d'accumulation des erreurs.

Pour faire face à ce problème, nous choisissons d'utiliser l'approche introduite dans (Lanaout *et al*., 2013). Il s'agit d'un SLAM visuel dont l'étape d'ajustement de faisceaux local a été modifiée afin d'inclure à la fois les données GPS et les informations fournie par le MNT. Tandis que les mesures GPS contraignent la position *dans le plan* de la caméra, le MNT permet de contraindre sur son altitude δ qui est supposée fixe par rapport au plan de la route (où évolue le véhicule) étant donné que la caméra est rigidement embarquée dans le véhicule. Afin de garantir plus de robustesse face aux données aberrantes du GPS et aux incertitudes du MNT, l'ajustement de faisceaux utilisé intègre une contrainte d'inégalité inspirée de la méthode introduite dans (Lhuillier, 2012). Celui-ci est réalisé en deux étapes. La première étape consiste à effectuer un ajustement de faisceaux classique où l'erreur de re-projection standard $E_1(\mathbf{x})$ est minimisée. Au cours de la deuxième étape, une seconde optimisation non linéaire est effectuée pendant laquelle, la distance entre les positions *dans le plan* de la caméra et les mesures GPS ainsi que la distance entre l'altitude de la caméra estimée par le SLAM, exprimée dans le repère de la route associée (la route la plus proche), et la hauteur souhaitée δ sont minimisées. Afin, de conserver une cohérence vis à vis de la géométrie multivues, cette seconde optimisation intègre, en plus du terme d'accroche aux données GPS et MNT, un terme de régularisation basé sur l'erreur de re-projection $E_1(\mathbf{x})$: ce terme interdit toute dégradation de l'erreur de re-projection au delà d'un seuil prédéfini e_t basé sur le résultat de la première optimisation. Plus le seuil e_t est élevé, plus la dégradation de la contrainte multi-vue est tolérée pour respecter les contraintes GPS et MNT. Cette dégradation est fixée à $5\,\%$ de l'erreur de re-projection initiale comme (Lhuillier, 2012) l'explique. La fonction de coût résultante est donnée par:

$$C_I(\mathbf{x}) = \frac{\omega}{e_t - E_1(\mathbf{x})} + \left\| \begin{pmatrix} \mathbf{t_x} \\ \mathbf{t_y} \\ \hat{\mathbf{t}}_\mathbf{z} \end{pmatrix} - \begin{pmatrix} \mathbf{t}_x^{gps} \\ \mathbf{t}_y^{gps} \\ \mathbf{h} \end{pmatrix} \right\|^2, \qquad (2)$$

où ω est une constante positive permettant de normaliser notre fonction de coût. Cette constante est déterminée à partir des valeurs initiales (*ie* après le premier ajustement de faisceaux classique) de l'erreur de re-projection ainsi que celle correspondante au terme d'accroche aux données GPS et MNT. Par ailleurs, $\mathbf{h} = (\underbrace{\delta \dots \delta}_{\text{N times}})^T$ et $\hat{\mathbf{t}}_\mathbf{z} = \left(\hat{t}_z^1 \dots \hat{t}_z^N\right)^T$ est le vecteur concaténant les altitudes de la caméra pour N images clés optimisées dans l'ajustement de faisceaux local. Notons que chaque altitude $\hat{t}_z^j$ ($j \in [1..N]$) est exprimée dans le repère de la route la plus proche de la j^{eme} pose de la caméra. Étant donné que la fréquence de GPS (1Hz) est faible par rapport à celle de la caméra utilisée (30Hz), une interpolation des données GPS est réalisée pour associer une mesure GPS à chaque image clé.

Cette approche permet de créer en ligne et automatiquement une reconstruction initiale géo-référencée et cohérente. Cependant, sa précision reste limitée à l'incertitude du GPS. Pour cette raison, la base obtenue est optimisée, *a posteriori*, à travers deux ajustements de faisceaux globaux contraints à des modèles SIG (*i.e* les modèles 3D des bâtiments et le MNT).

4.2. Recalage 2D exploitant les modèles 3D des bâtiments

Pour traiter les imprécisions caractérisant les degrés de liberté *dans le plan*, (Lothe *et al*., 2009) proposent d'utiliser un ajustement de faisceaux global contraint aux modèles 3D des bâtiments où seuls les points 3D associés aux modèles sont exploités. Ceci peut entraîner des problèmes de convergence quand peu de bâtiments sont visibles. Pour faire face à cette limitation, nous choisissons d'adopter plutôt la méthode introduite par (Tamaazousti *et al*., 2011). Celle-ci exploite dans l'ajustement de faisceaux à la fois les points 3D associés aux modèles des bâtiments $Q_i \in \mathcal{M}$ et ceux appartenant au reste de l'environnement $Q_i \in \mathcal{U}$. La fonction de coût utilisée est donc composée par deux termes. Le premier terme est associé à l'ensemble de points $Q_i \in \mathcal{U}$ où deux observations $(u_{i,j}, u_{i,k})$ d'un même point Q_i sont liées par la matrice fondamentale. Le deuxième terme correspond à l'ensemble de points $Q_i \in \mathcal{M}$ où deux observations d'un même point Q_i, supposé appartenir à une façade de bâtiment, sont liées par une Homographie.

Notons qu'à l'instar des solutions introduites dans (Lothe *et al*., 2009) et (Tamaazousti *et al*., 2011), nous utilisons la méthode de segmentation de nuage de point basée sur le lancer de rayon. Cependant, contrairement à ces deux méthodes, au lieu d'optimiser tous les degrés de liberté de la reconstruction durant l'ajustement de faisceaux contraint, seuls les paramètres *dans le plan* sont raffinés afin d'éviter de détériorer les paramètres *hors plan* non contraints par les modèles 3D des bâtiments.

$$E_2(\mathbf{c_1}) = \sum_{i \in \mathcal{U}} \sum_{(j,k) \in \mathcal{A}_i}^{j \neq k} \rho\left(d_l\left(u_{i,j}, F_{j,k}(\mathbf{c_1})\, u_{i,k}\right), s_1\right) + \sum_{i \in \mathcal{M}} \sum_{(j,k) \in \mathcal{A}_i}^{j \neq k} \rho\left(d\left(u_{i,j}, H_{j,k}(\mathbf{c_1})\, u_{i,k}\right), s_2\right), \tag{3}$$

avec $d_l(u, l)$ est la distance euclidienne entre un point et une droite et $d(u_1, u_2)$ est la distance euclidienne entre deux points. $\rho(\mathbf{v}, s)$ est le M-estimateur de Geman-McClure dont le seuil s est calculé en utilisant le MAD *Median Absolute Deviation* du vecteur des résidus $\mathbf{v}$. Pour une meilleure convergence, un processus d'optimisation itératif est adopté pour cette étape où les associations point-façade de bâtiment sont remises en cause après la minimisation de la fonction de coût assurée par l'algorithme du Levenberg Marquardt (Marquardt, 1963).

4.3. Raffinement précis exploitant les modèles 3D des bâtiments et le MNT

Une fois que le recalage *dans le plan* est réalisé, seules des imperfections, causées par une mauvaise estimation des degrés de liberté *hors plan*, restent notables. Pour traiter ces imperfections, un deuxième ajustement de faisceaux exploitant un modèle SIG complet est réalisé. Durant cette étape, tous les degrés de liberté de la reconstruction sont optimisés et contraints simultanément aux modèles 3D des bâtiments et au MNT. Tandis que les bâtiments contraignent les degrés de liberté *dans le plan*, le MNT fournit des informations relatives aux degrés de liberté restants. En effet, en plus de la contrainte explicite en altitude, le MNT permet, globalement, de contraindre implicitement l'angle tangage dans les lignes droites et l'angle roulis et tangage au niveau des virages. Pour introduire progressivement les contraintes du MNT sans perturber le résultat du recalage *dans le plan* effectué dans l'étape précédente, un ajustement de faisceaux avec une contrainte d'inégalité (voir le principe dans la section 4.1) est utilisé. La fonction de coût associée est composée par l'erreur de re-projection prenant en compte la contrainte des bâtiments et un terme d'accroche aux données calculé en se basant sur la contrainte MNT. Ce terme représente la distance entre l'altitude de chaque pose de la caméra $\mathbf{t}_j^{k_j}(z)$ exprimée dans le repère de la route la plus proche et l'altitude δ souhaitée. Par conséquent, la fonction de coût résultante est donnée par :

$$F_I(\mathbf{x}) = \frac{\omega}{e_t - E_2(\mathbf{x})} + \|\hat{\mathbf{t}}_\mathbf{z} - \mathbf{h}\|^2 \tag{4}$$

avec $E_2(\mathbf{x})$ est la fonction de coût introduite dans la section 4.2 mais cette fois ci, tous les degrés de liberté sont optimisés contrairement à l'étape précédente où seuls les paramètres dans le plan sont optimisés. Un processus d'optimisation itératif semblable à celui introduit dans la section 4.2 est adopté. En effet, les associations point-façade de bâtiment et caméra-plan de route sont remises en cause après la minimisation de la fonction de coût assurée par l'algorithme du Levenberg Marquardt.

Le processus complet d'optimisation (*ie* recalage 2D suivi du raffinement précis) est résumé dans l'algorithme 1.

Algorithme 1. *Processus complet d'optimisation de la base d'amers géo-référencés intégrant les informations du MNT et des modèles 3D des bâtiments*

$\texttt{nouveauMAD} = 0$;
$\texttt{ancienMAD} = 0$;
repeat
  Segmenter $\{Q_i\}_{i=1}^{M}$ en $\{Q_i\}_{i\in\mathcal{U}}$ et $\{Q_i\}_{i\in\mathcal{M}}$;
  Projeter les points 3D $\{ Q_i\}_{i\in\mathcal{M}}$ sur leurs plans correspondants;
  $\texttt{ancienMAD} = \texttt{MAD}$ calculé sur les erreurs de re-projection de points $\{Q_i\}_{i\in\mathcal{M}}$;
  Calculer les seuils de rejet du M-estimateur s_1 et s_2;
  Minimiser la fonction de coût Eq. 3 en utilisant l'algorithme de Levenberg Marquardt;
  $\texttt{nouveauMAD} = \texttt{MAD}$ calculé sur les erreurs de re-projection de points $\{Q_i\}_{i\in\mathcal{M}}$;
  Triangulation des points $\{Q_i\}_{i\in\mathcal{M}}$ en prenant en compte les nouvelles poses de la caméra;
until $(\texttt{nouveauMAD} - \texttt{ancienMAD}) < 0.1$;
$\texttt{nouveauMAD} = 0$;
$\texttt{ancienMAD} = 0$;
Calculer le poids ω et la dégradation maximale tolérée e_t (une dégradation de 5% de l'erreur de re-projection initiale $E_2(\mathbf{x})$);
repeat
  Associer chaque caméra au plan de la route le plus proche dans le MNT;
  Segmenter $\{Q_i\}_{i=1}^{M}$ en $\{Q_i\}_{i\in\mathcal{U}}$ et $\{Q_i\}_{i\in\mathcal{M}}$;
  Projeter les points 3D $\{ Q_i\}_{i\in\mathcal{M}}$ sur leurs plans correspondants;
  $\texttt{ancienMAD} =$ MAD calculé sur les erreurs de re-projection des points $\{Q_i\}_{i\in\mathcal{M}}$;
  En utilisant l'algorithme du Levenberg Marquardt, minimiser la fonction de coût Eq. 4 avec la contrainte d'inégalité;
  $\texttt{nouveauMAD} =$ MAD calculé sur les erreurs de re-projection des points $\{Q_i\}_{i\in\mathcal{M}}$;
until $(\texttt{nouveauMAD} - \texttt{ancienMAD}) < 0.1$ *et* (*les associations caméra/plan de routes soient stabilisées*);

5. Résultats

Dans cette section, nous proposons une évaluation complète de notre processus de création de base d'amers sur des données de synthèse et d'autres réelles (section 5.1). L'objectif de cette évaluation expérimentale est double :

1. Tout d'abord, nous souhaitons démontrer la robustesse et la précision de notre méthode en l'évaluant sur des séquences de synthèse et d'autres réelles de plusieurs kilomètres (voir section 5.2).

2. Ensuite, nous souhaitons démontrer les performances de notre algorithme en comparant les résultats de ce dernier avec la méthode de (Lothe *et al.*, 2009) (voir section 5.3).

Dans ce qui suit nous commençons par présenter les séquences utilisées.

5.1. Séquences utilisées

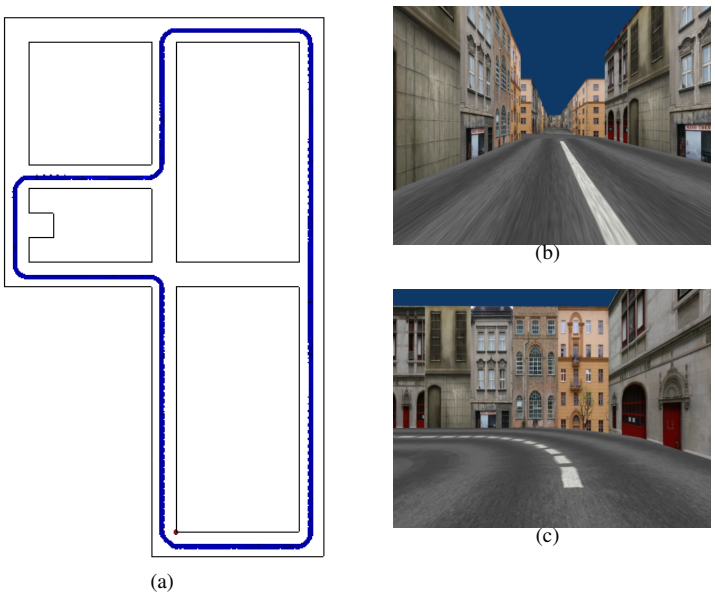


Figure 1. Illustrations de la séquence de synthèse. (a) En noir vue de dessus du modèle 3D des bâtiments, en bleu la trajectoire réelle de la caméra. (b), (c), Illustrations de la séquence utilisée

Pour évaluer notre algorithme de création de base d'amers géo-référencés à travers la fusion hors ligne des contraintes, multivues, GPS, MNT et modèles 3D des bâtiments, nous avons eu recours à des données synthétiques et d'autres réelles.

5.1.1. Séquence de synthèse

La séquence de synthèse utilisée est illustrée dans la figure 1. Cette séquence simule le parcours d'un véhicule dans un milieu urbain dense. La caméra est embarquée sur le véhicule à une altitude de 1.5 m par rapport à la route supposée parfaitement plate. Pour simuler les données GPS, les positions *dans le plan* de la caméra, fournies par la vérité terrain, sont perturbées comme suit: à chaque position *dans le plan*

fournie par la vérité terrain, un biais d'amplitude 5 m (dont la direction change aux virages) ainsi qu'un bruit Gaussien d'amplitude 1 m sont rajoutés.

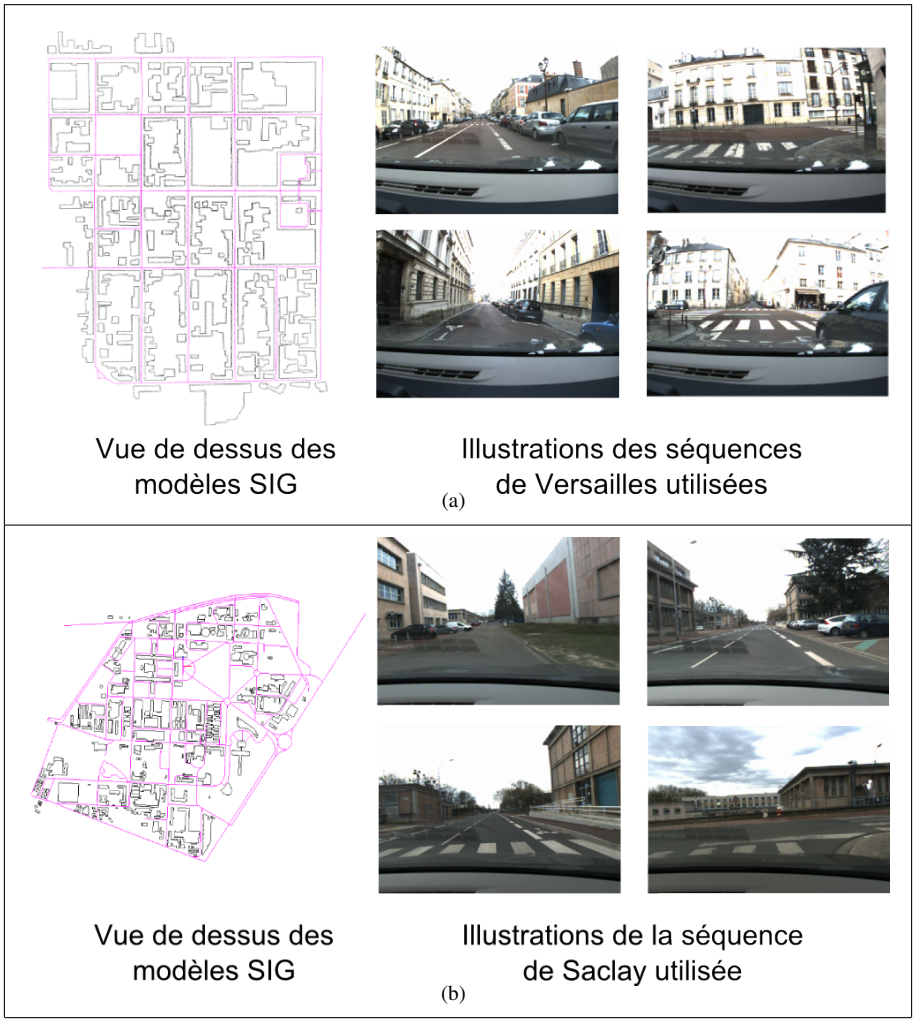


Figure 2. Illustrations des séquences réelles utilisées et les modèles SIG associés. (a) Séquences de Versailles. (b) Séquence de Saclay. La colonne de gauche représente une vue de dessus des modèles SIG utilisés pour chacune des séquences: en noir les modèles 3D des bâtiments, en rose les MNT (axes 3D des routes à partir des quelles le plan chaque route a été déterminé)

5.1.2. Séquences réelles

Les performances de notre méthode sont également vérifiées sur trois séquences réelles, voir figure 2. Ces séquences sont enregistrées dans deux quartiers différents

(deux séquences dans la ville de Versailles et une séquence dans la ville de Saclay) dans des conditions de conduite normale (50 Km/h). Pour ceci, le véhicule a été équipé par un GPS standard 1 Hz et une caméra RGB fournissant 30 images par seconde et avec un champ de vision de 90°. Les modèles SIG utilisés sont issus de la base Géoportail de l'IGN et ont une erreur ne dépassant pas les 2 m.

Les trois séquences utilisées représentent des parcours de 2400 m, de 1800 m et 1200 m. Notons que même si ces séquences ne couvrent pas tout un quartier, il est possible, comme le montre la figure 3, de fusionner plusieurs bases d'amers d'un même quartier pour créer une base d'amers à l'échelle d'une ville.

Figure 3. Concaténation de deux bases de données différentes (bleue et verte) dans le quartier de Versailles. Les point 3D redondants dans les deux bases sont détectés et fusionnés. Un ajustement de faisceaux global peut être par la suite réalisé afin d'améliorer la précision de la base résultante aux zones de recouvrement

5.2. Évaluation de l'ensemble de notre solution pour la création d'une base d'amers géo-référencés

5.2.1. Protocole expérimental

Pour évaluer la capacité de notre algorithme à fonctionner dans des conditions normales de circulation, nous proposons d'étudier la qualité de la reconstruction fournie par notre méthode sur les séquences réelles décrites ci-dessus. Ne disposant pas de vérité terrain pour ces expériences, cette première évaluation se limite à une appréciation visuelle (*ie.* résultats qualitatifs), que ce soit à travers la visualisation de la trajectoire

et des points 3D reconstruits par rapport aux modèles 3D des bâtiments (voir figure 4) ou la projection de ces modèles dans différentes vues des séquences de test (voir figure 5). Afin de mettre en évidence l'impact du processus d'optimisation, ces résultats seront présentés pour les différentes étapes du processus.

5.2.2. Résultat sur des données réelles

Pour créer la base d'amers souhaitée, seules quelques minutes sont nécessaires en utilisant un code non optimisé exécuté sur un Intel(R) Xeon(R) CPU quad cores 2.4 GHz. En effet, pour la séquence de Versailles parcourant une distance de 2400 m, la base initiale, de volume 8 Mo, contenant 548 vues géo-référencées et 34178 points 3D est obtenue en ligne. Ensuite, 50 secondes sont uniquement demandées pour réaliser le recalage *dans le plan* décrit dans la section 4.2 et 90 secondes sont nécessaires pour effectuer le dernier raffinement introduit dans la section 4.3.

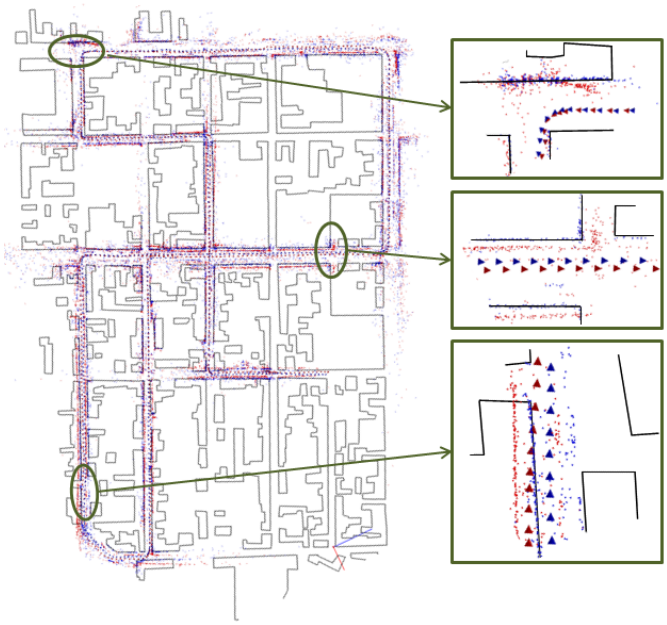
Comme le montre la figure 4, la fusion du SLAM avec les données GPS et MNT, décrit dans la section 4.1, permet de créer automatiquement des bases d'amers géo-référencés (les reconstructions rouges dans la figure 4). Toutefois, ces reconstructions initiales présentent d'importantes imprécisions au niveau des degrés de liberté *dans le plan* comme il est montré dans la figure 4 et mis en évidence dans les figures 6(d) et 6(a). Une fois le recalage *dans le plan* effectué, ces incertitudes sont corrigées. Par conséquent, les re-projections des modèles des bâtiments sont mieux alignées avec les façades observées dans le flux vidéo (voir les figures 6(e) et 6(b)). Enfin, le dernier raffinement exploitant le modèle SIG complet permet de corriger les imperfections restantes sur les degrés de liberté *hors plan* (voir figures 6(f) et 6(c)). Cette correction est réalisée sans perturber les paramètres *dans le plan* comme le montre la figure 4 où le nuage de points final représenté en bleu est parfaitement recalé sur les modèles des bâtiments.

Une fois les bases d'amers créées, elles peuvent être utilisées pour assurer une localisation en ligne par mise en correspondance des amers 3D de la base avec les primitives 2D extraites des images traitées (Middelberg *et al*., 2014) (Arth *et al*., 2009),(Gay-Bellile *et al*., 2010). Pour nos évaluations nous avons choisi d'utiliser la méthode proposée par (Gay-Bellile *et al*., 2010) qui fusionne la mise en correspondance 2D-3D avec l'algorithme du SLAM incrémental. Comme le montre la figure 6, la précision de la localisation obtenue a permis des applications de RA. Une vidéo résultat est également disponible ici https://www.youtube.com/watch?v=YkZeuPSu7AQ

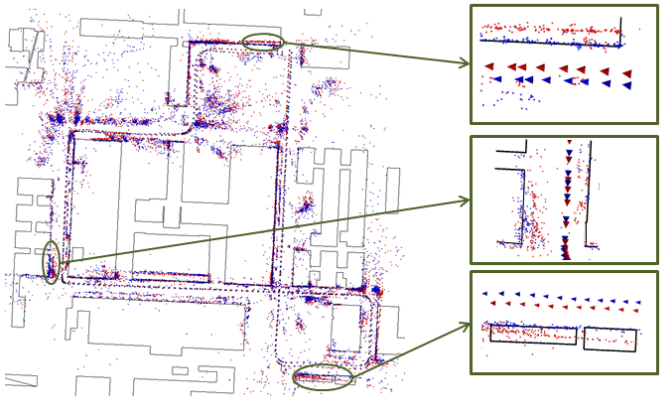
5.3. Comparaison avec l'approche de Lothe et al.

5.3.1. Protocole expérimental

Dans cette section, nous comparons la précision de nos bases d'amers avec celles obtenues avec la méthode de (Lothe *et al*., 2009). Principalement, nous souhaitons comparer les deux méthodes d'optimisation utilisées: la fusion des contraintes aux bâtiments et au MNT à travers une optimisation intégrant une contrainte d'inégalité



(a)



(b)

Figure 4. Validation de notre processus de création de bases d'amers géo-référencés. Vues de dessus des bases de données obtenues dans les quartiers de Versailles et Saclay après la première étape (SLAM contraint aux données GPS et MNT, section 4.1) (rouge) et à l'issue de notre processus d'optimisation complet (section 4.2 et section 4.3) (bleu)

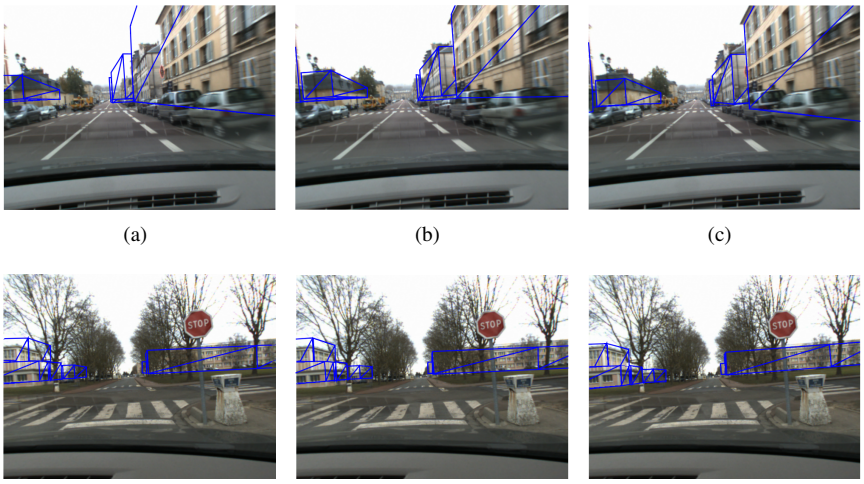
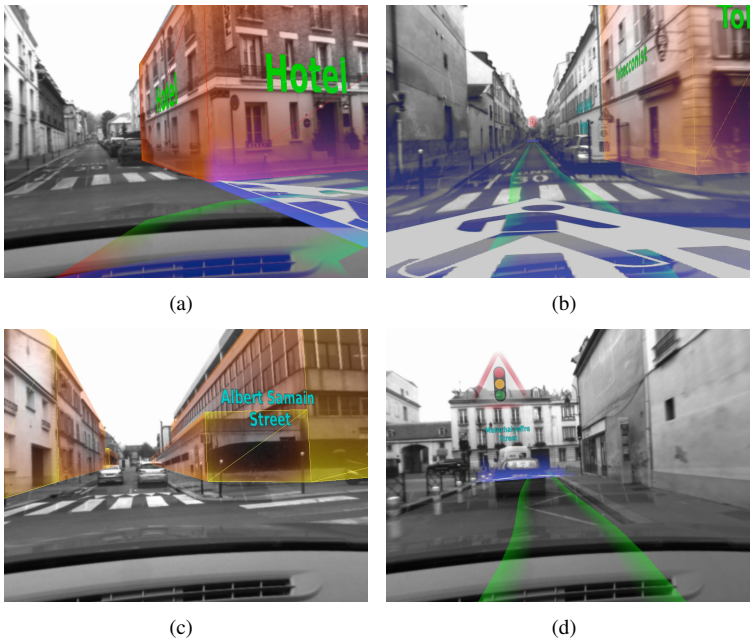


Figure 5. Re-projection des modèles des bâtiments. Les résultats d'une des séquences de Versailles sont présentés au niveau de la ligne de dessus. Les résultats de la séquence de Saclay sont présentés dans la ligne de dessous. (a) et (d) présentent les résultats après la première étape de la création de la base de données (SLAM contraint aux données GPS et MNT, section 4.1); (b) et (e) sont les résultats après le premier ajustement de faisceaux global (section 4.2), (c) et (f) présentent les résultats finaux obtenus après le deuxième ajustement de faisceaux global (section 4.3)

contre l'ICP non rigide suivi par l'ajustement de faisceaux global contraint uniquement aux modèles des bâtiments introduit par (Lothe *et al.*, 2009). Pour ceci, les deux méthodes d'optimisation sont initialisées en utilisant le SLAM contraint aux données GPS et la contrainte en altitude qui fournit une reconstruction initiale assez précise. La comparaison en question est effectuée dans un premier temps sur la séquence de synthèse dont nous possédons la vérité terrain. Une évaluation sur les séquences réelles est par la suite proposée.

Pour la séquence de synthèse, l'évaluation est réalisée en analysant l'évolution de l'erreur de la localisation *dans le plan*, l'erreur en altitude, ainsi que l'erreur angulaire pour les deux bases obtenues.

La comparaison entre notre solution et celle proposée par (Lothe *et al.*, 2009) est également établie sur les séquences réelles. Étant donné que la vérité terrain n'est pas disponible pour ces séquences, nous labellisons manuellement les coins des bâtiments(plusieurs clics sont réalisés pour chaque coin. Ensuite, la moyenne de ces clics est considérée comme la vérité terrain) dans quelques images extraites des flux vidéos (environs 20 images distribuées uniformément dans chaque séquence) comme vérité terrain. Nous calculons par la suite l'erreur de re-projection entre les coins labellisés et la re-projection des coins des modèles des bâtiments.



(a) (b)

(c) (d)

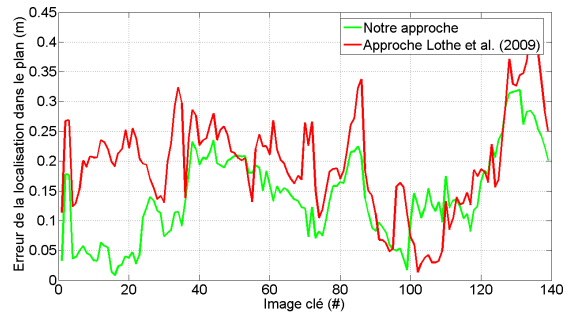
Figure 6. Exemples d'applications de RA: projection des modèles des bâtiments, insertion des informations routières et la trajectoire du véhicule

5.3.2. Évaluation sur la séquence de synthèse

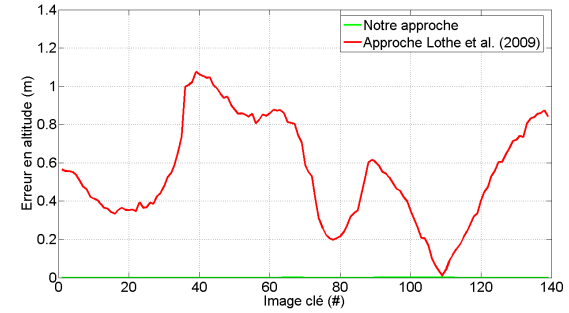
La figure 7 inclut l'erreur de la localisation *dans le plan*, l'erreur en altitude, ainsi que l'erreur angulaire pour les deux bases obtenues. Nous remarquons que la localisation dans le plan est assez précise pour l'algorithme de (Lothe *et al.*, 2009). Toutefois, notre solution améliore cette précision en réduisant la moyenne des erreurs de $0,3$ m à $0,1$ m. La même observation est notée pour les degrés de liberté hors plan. En effet, notre solution réduit la moyenne des erreurs en altitude de $0,5$ m à une erreur négligeable ($\simeq 0$ m) tandis que la moyenne des erreurs en orientation est passée de $0,012$ rad $\sim 0,68$ deg pour l'algorithme de Lothe *et. al* à $0,003$ rad $\sim 0,17$ deg en utilisant notre approche.

5.3.3. Évaluation sur les séquences réelles

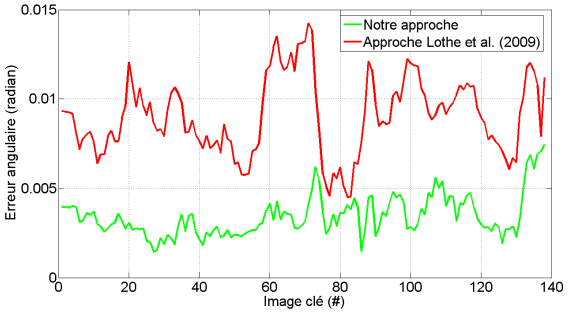
Les résultats de comparaison entre notre solution et celle proposée par (Lothe *et al.*, 2009) sont résumés dans le tableau 1. La mesure d'erreur de re-projection inclut sûrement les incertitudes des modèles des bâtiments et celles de la labellisation manuelle. Cependant, ces incertitudes restent faibles en comparaison avec l'amélioration notable que notre solution apporte. En effet, pour les séquences de Versailles, l'approche proposée réduit de moitié la moyenne des erreurs de re-projections obtenues par l'algorithme de (Lothe *et al.*, 2009). L'écart-type des erreurs mesurées a également baissé considérablement en utilisant notre méthode de $13,3$ pixels à $1,91$



(a)

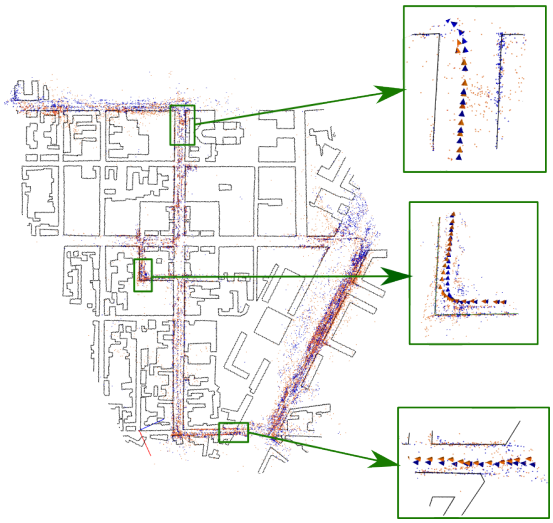


(b)

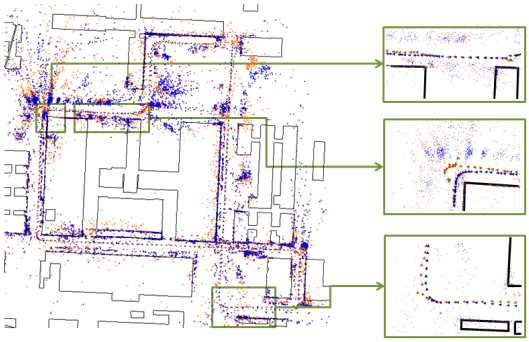


(c)

Figure 7. Comparaison avec la méthode de Lothe et al. en utilisant la séquence de synthèse. (a), (b) et (c) représentent respectivement l'évolution de l'erreur de la localisation dans le plan, en altitude et en orientation. Les résultats obtenus avec la méthode de Lothe et al. sont tracés en rouge tandis que les résultats obtenus avec notre algorithme sont représentés en vert



(a)



(b)

Figure 8. Comparaison avec la méthode de Lothe et al. sur des séquences réelles. Vue de dessus des bases de données obtenues par notre méthode (en bleu) et celles obtenues par la méthode proposée par Lothe et al. (en orange). (a) représente les résultats obtenus pour la deuxième séquence de Versailles. (b) représente les résultats obtenus pour la séquence de Saclay

pixels. Ces résultats mettent en évidence la bonne précision que notre solution assure contrairement à la méthode de (Lothe *et al.*, 2009) qui présente des imprécisions locales et globales. En effet, utiliser uniquement les points 3D associés aux modèles des bâtiments cause des imprécisions locales quand peu de bâtiments sont observables comme le montre la figure 8 où le nuage de points (en orange) n'est pas aligné avec les empreintes des bâtiments. De plus, optimiser tous les degrés de liberté à travers un ajustement de faisceaux contraint uniquement aux modèles des bâtiments entraîne des

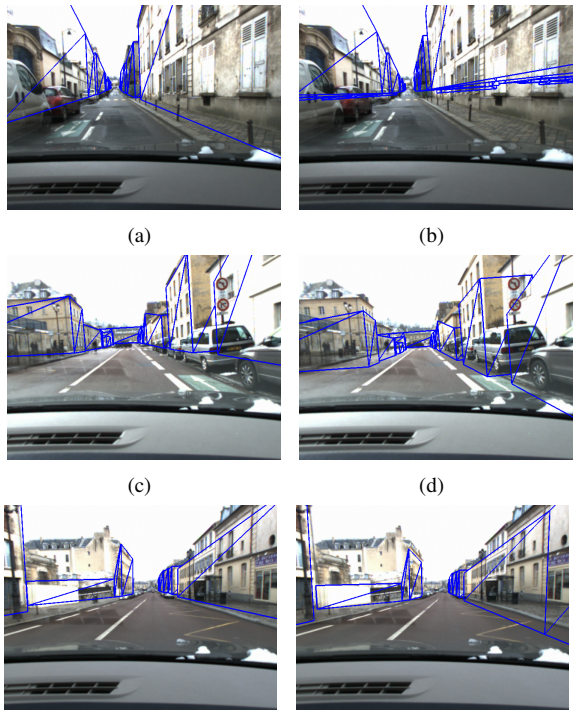


Figure 9. Exemples de re-projection des modèles des bâtiments sur des images extraites de la première séquence de Versailles. La colonne de gauche représente le résultat de la méthode proposée. La colonne de droite représente le résultat obtenu avec la méthode introduite par Lothe et al.. Les erreurs de re-projections associées sont: (a) 8,23 pixels, (b) 37,66 pixels, (c) 6,50 pixels, (d) 13,95 pixels (e) 6,24 pixels and (f) 7,19 pixels

imprécisions globales principalement observées au niveau des paramètres *hors plan* (voir figure 9).

Tableau 1. Résultats quantitatifs pour les séquences réelles. Comparaison des erreurs de re-projection entre les coins labellisés et les coins des modèles de bâtiments re-projetés

		Erreur de re-projection (pixels)			
		Moyenne	Écart type	Max	Min
Versailles	(Lothe *et al.*, 2009)	17.85	13,30	40,90	4,11
	Nous	7,32	1,91	10.90	4,04
Saclay	(Lothe *et al.*, 2009)	14,92	7,02	30,14	5,56
	Nous	8,37	3,25	16,31	3,52

6. Conclusion

Dans cet article, nous avons proposé une nouvelle solution précise et robuste pour modéliser automatiquement des grands environnements. Cette approche fusionne les contraintes multivues, celles apportées par le GPS et des modèles SIG. Notre solution est facile à déployer puisqu'elle ne nécessite pas des matériels coûteux contrairement à la majorité des solutions existantes. Les résultats quantitatifs sur les séquences de synthèses ainsi que les résultats qualitatifs sur les séquences réelles illustrent la précision élevée des bases d'amers construites. Dans nos prochains travaux, nous nous intéresserons aux imperfections affectant l'angle roulis qui sont parfois visibles à cause des courbures éventuelles des routes et qui ne sont pas modélisées dans le MNT utilisé. Nous essayerons également de modéliser les incertitudes liées aux différentes sources d'information et d'en tenir compte dans la fusion afin d'améliorer la précision des bases reconstruites. Des travaux d'amélioration de la segmentation du nuage de points sont également envisageables afin d'être plus robuste au problème d'occultation des bâtiments notamment par les voitures garées ou les arbres.

Bibliographie

Agarwal S., Furukawa Y., Snavely N., Simon I., Curless B., Seitz S. M. *et al.* (2011). Building rome in a day. *Commun. ACM*, vol. 54, n° 10, p. 105-112.

Arth C., Wagner D., Klopschitz M., Irschara A., Schmalstieg D. (2009). Wide area localization on mobile phones. In *Ieee international symposium on mixed and augmented reality*, p. 73 - 82.

Frahm J. M., Fite-Georgel P., Gallup D., Johnson T., Raguram R., Wu C. *et al.* (2010). Building rome on a cloudless day. In *Ieee european conference on computer vision*, p. 368-381.

Gay-Bellile V., Lothe P., Bourgeois S., Royer E., Naudet-Collette S. (2010). Augmented reality in large environments: Application to aided navigation in urban context. In *Ieee international symposium on mixed and augmented reality*, p. 225-226.

Kaminsky R., Snavely N., Seitz S., Szeliski R. (2009). Alignment of 3d point clouds to overhead images. In *Ieee conference on computer vision and pattern recognition*, p. 63 - 70.

Lanaout D., Gay-Bellile V., Bourgeois S., Dhome M. (2013). Vehicle 6-dof localization based on slam constrained by gps and digital elevation model information. In *Ieee international conference on image processing*, p. 2504-2508.

Lhuillier M. (2012). Incremental fusion of structure-from-motion and gps using constrained bundle adjustments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, n° 12, p. 2489-2495.

Li Y., Snavely N., Huttenlocher D., Fua P. (2012). Worldwide pose estimation using 3d point clouds. In *Ieee european conference on computer vision*, p. 15-29.

Lothe P., Bourgeois S., Dekeyser F., Royer E., Dhome M. (2009). Towards geographical referencing of monocular slam reconstruction using 3d city models: Application to real-time accurate vision-based localization. In *Ieee conference on computer vision and pattern recognition*, p. 2882-2889.

Marquardt D. W. (1963). An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial & Applied Mathematics*, vol. 11, p. 431-441.

Middelberg S., Sattler T., Untzelmann O., Kobbelt L. (2014). Scalable 6-dof localization on mobile devices. In *Ieee european conference on computer vision*, p. 268-283.

Mouragnon E., Lhuillier M., Dhome M., Dekeyser F., Sayd P. (2006). Real time localization and 3d reconstruction. In *Ieee conference on computer vision and pattern recognition*, p. 363-370.

Strecha C., Pylvänäinen T., Fua P. (2010). Dynamic and scalable large scale image reconstruction. In *Ieee conference on computer vision and pattern recognition*, p. 406 - 413.

Tamaazousti M., Gay-Bellile V., Naudet-Collette S., Bourgeois S., Dhome M. (2011). Nonlinear refinement of structure from motion reconstruction by taking advantage of a partial knowledge of the environment. In *Ieee conference on computer vision and pattern recognition*, p. 3073-3080.

Wang C., Wilson K., Snavely N. (2013). Accurate georegistration of point clouds using geographic data. In *Ieee conference 3d vision*, p. 33-40.

Zamir A. R., Shah M. (2010). Accurate image localization based on google maps street view. In *Ieee european conference on computer vision*, p. 255-268.

Reçu le 5/12/2014
Accepté le 2/06/2015