# Fusion de données Gps, Sig et Vidéo pour la reconstruction d'environnements urbains

# Gps, Gis and Video Fusion for Building Reconstruction

Gaël Sourimant[1] Luce Morin[2] Kadi Bouatouch[3]

IRISA / INRIA Rennes
Campus Universitaire de Beaulieu,
Avenue du Général Leclerc, 35042 RENNES Cedex - France
{gael.sourimant,luce.morin,kadi.bouatouch}@irisa.fr

## Résumé

*La modélisation en 3D d'environnements urbains est un sujet largement étudié depuis plusieurs années, son attrait étant lié aux applications diverses d'une telle modélisation : navigation virtuelle, réalité augmentée, planification architecturale, etc. L'une des difficultés à ce jour dans ce contexte reste l'acquisition et le traitement de données à grande échelle si l'on cherche à obtenir une reconstruction précise non seulement géométriquement, mais également photométriquement (on veut les véritables textures de chaque bâtiment). Nous présentons dans cet article un système permettant de calculer les positions géo-référencées et les orientations d'images de bâtiments issues de séquences vidéo non calibrées, en tant que préalable indispensable au bon conditionnement de la reconstruction* 3D *précise d'environnements urbains à grande échelle, notre méthode étant basée sur la fusion de données multimodales, et plus précisément de positions* GPS*, de modèles* 3D *polyédriques simples de bâtiments ainsi que de séquences d'images de ces bâtiments.*

## Mots Clef

Modélisation urbaine, modélisation basée images, géo-localisation, réalité virtuelle.

## Abstract

3D *reconstruction of urban environments is a widely studied subject since several years, since it can lead to many useful applications : virtual navigation, augmented reality, architectural planification, etc. One of the most difficult problem nowadays in this context is acquisition and treatment of data if very large scale and precise reconstruction is aimed. In this paper we present a system for computing geo-referenced positions and orientations if images of buildings from non calibrated videos. Providing such information is a mandatory step to well conditioned large scale and precise* 3D *reconstruction of urban areas. Our method is based on the fusion of multimodal datasets, namely* GPS *measures, video sequences and rough* 3D *models of buildings.*

## Keywords

City modeling, image-based modeling, geo-localization, virtual reality.

# 1 Introduction

Le succès récent de Google Earth montre que l'ajout de textures photo-réalistes sur une carte 2D ajoute beaucoup d'information pour l'utilisateur par rapport à une carte symbolique traditionnelle. Les fonctionnalités 3D offertes par cet outil, telles que la navigation ou la représentation en 3D des bâtiments est une autre raison de son succès. Cependant, les modèles 3D fournis sont peu réalistes (ce sont des parallélépipèdes gris). De même que dans le cas de photos aériennes superposées à des cartes 2D, il serait intéressant d'avoir accès à des modèles 3D photo-réalistes. La modélisation 3D d'environnements urbains a d'autres applications, telles que les jeux, le tourisme virtuel, le géo-positionnement ou la réalité virtuelle. Malheureusement, la modélisation manuelle par un graphiste est un processus long qui ne peut être appliqué à la modélisation à grande échelle d'environnements urbains.

Nous présentons dans cet article un système permettant de calculer des positions et orientations géo-référencées d'images de bâtiments. Notre approche est basée sur la fusion de données multimodales, à savoir des images prises au sol de bâtiments acquises avec des mesures GPS, ainsi qu'une base de donnée de type SIG composée d'un ensemble de modèles 3D de bâtiments décrits par leur empreinte au sol et leur élévation. Ce type de données SIG existe pour de nombreuses villes en France[1]. Si ce type de modélisation est pertinent pour une visualisation aérienne, il n'est pas satisfaisant pour une navigation 3D au niveau

[1] Selon l'IGN, la visualisation 3D des villes françaises sera publique en 2007

du sol. La vidéo et la base SIG contiennent des informations complémentaires : la vidéo fournit le photo-réalisme et les détails géométriques des bâtiments, tandis que les modèles SIG donnent une géométrie "propre" et complète de la scène, structurée en bâtiments individuels. Des mesures GPS sont également acquises de façon synchronisée avec la vidéo. Afin de combiner ces types de données différents, la première étape est de les mettre en correspondance dans le même système de coordonnées. La mise en correspondance est de fait le point sensible du système, étant donnée qu'elle requiert des correspondances géométriques entre des données de type tout à fait différent.
La suite de l'article est organisée de la façon suivante : la section 2 présente des travaux similaires sur la reconstruction de zones urbaines. Les données utilisées en entrée sont décrites dans la section 3 puis la méthode de recalage des données est explicitée dans la section 4. Après avoir donné quelques résultats dans la section 5 nous concluons sur la viabilité de la méthode et donnons quelques perspectives (section 6).

# 2 Travaux antérieurs et approche proposée

Plusieurs études ont déjà abordé le problème de la reconstruction 3D d'environnements urbains. Selon le niveau de détail et l'ampleur de la reconstruction, différents types de données ont été utilisés, par exemple des images aériennes, satellites ou acquises au sol.

**Reconstruction aérienne.** Les travaux précédents sur la modélisation urbaine impliquent généralement l'utilisation d'images aériennes ou satellites. Ces méthodes sont généralement décomposées en deux étapes. Tout d'abord, les bâtiments sont détectés en utilisant des algorithmes de segmentation ou d'extraction et connexion de lignes [5, 9, 21], puis les bâtiments sont effectivement reconstruits en utilisant de la stéréovision [5] ou une mise en correspondance entre les images et des primitives 3D simples fusionnées dans une structure d'arbre CSG [18]. Ces méthodes donnent généralement des modèles 3D pauvres géométriquement, dans le sens où même si la forme globale est bien estimée (dimensions, angles, etc.), aucune information sur la texture ou les détails géométriques des façades ne peut être extraite.

**Reconstruction au sol.** La reconstruction au sol fait référence à la modélisation 3D en utilisant des données acquises au sein même des environnements urbains : images fixes ou vidéos. Ce type d'approches permet l'extraction de données photométriques (textures des bâtiments) et géométriques (positions et forme des fenêtres, portes, etc.).
Les méthodes permettant de modéliser des scènes 3D à partir d'un ensemble d'images ont été largement étudiées en vision par ordinateur, que ce soit dans le contexte de la *Structure à partir du mouvement* qui permet de retrouver le calibrage et la pose de la caméra ainsi que la structure de la scène [8, 11], ou dans le contexte de la *Modélisation basée images* qui permet de créer directement des modèles 3D à partir d'un ensemble d'images, comme par exemple le système *Façade* [2] qui a été utilisé pour modéliser certaines parties du campus de l'université de Berkeley. Plusieurs travaux visent une reconstruction géométrique locale [23, 15], alors que d'autres projets visent une reconstruction à grande échelle d'environnements urbains à partir d'images au sol. Dans le projet *MIT City Scanning* [19], des images hémisphériques calibrées sont utilisées pour extraire des plans correspondant aux façades, qui sont alors texturés et raffinés géométriquement en utilisant des techniques de reconnaissance des formes et de vision par ordinateur. Dans le projet *UrbanScape* [1], un système complètement automatique pour une reconstruction temps-réel et précise à partir de flux vidéos est présenté, en utilisant à la fois le CPU et le GPU. Le projet 4D *Cities* [14, 12] cherche à créer des modèles 3D variant avec le temps à partir d'une collection d'images prises d'endroit différents, à des époques également différentes.
L'inconvénient principal de ces approches est qu'elles restent parfois locales ou qu'elles nécessitent des acquisitions complexes et des temps de traitement qui peuvent devenir prohibitifs dans le cadre d'une reconstruction à très grande échelle.

**Tirer profit des deux approches.** Peu d'études cherchent à tirer profit des deux approches. Cependant, un projet intéressant allant dans ce sens est celui de Frueh et Zakhor [6], où sont utilisées conjointement des caméras vidéo et laser pour l'acquisition de données. Une caméra laser orientée verticalement mesure un nuage dense de points des façades, tandis que la caméra vidéo est utilisée pour le texturage. Une caméra laser orientée horizontalement est également utilisée pour estimer la position lors de l'acquisition, et est mise en correspondance avec une carte aérienne des contours de bâtiments pour assurer une cohérence globale des différents modèles 3D entre eux. Le principal inconvénient de cette approche est la complexité du système d'acquisition ainsi que le post traitement complexe des données acquises, principalement concernant le nuage de points 3D extrêmement dense.

**Approche proposée.** Nous résolvons le problème avec une approche de type raffinement, dans le sens où nous partons de modèles 3D existant qui sont simples géométriquement et non texturés, mais exprimés dans un repère géoréférencé, et nous ajoutons graduellement de l'information géométrique et photométrique en utilisant des données extraites de vidéos en utilisant des algorithmes issus de la robotique et de la vision. Contrairement aux solutions basées uniquement sur des données aériennes, qui ne fournissent qu'une structure brute des bâtiments, notre système commence avec ces modèles et vise à les raffiner en utilisant des informations extraites de vidéos de ces bâtiments. Enfin, notre approche vise une reconstruction à grande échelle en utilisant des données à la fois aériennes et au sol tout en restant simple du point de vue de l'acquisition (nous utilisons une simple caméra et un récepteur GPS du commerce). De

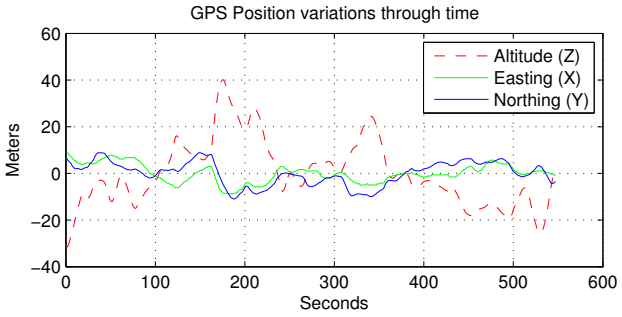



FIG. 1 – Positions GPS exprimées dans le repère UTM, par rapport au temps, pour un point fixe

plus, la procédure de post-traitement des données acquises reste plus simple que pour les méthodes décrites dans [6].

# 3 Données utilisées

Dans cette section, nous présentons brièvement les différents types de données utilisés par la suite afin de donner une base pour la compréhension des parties suivantes. Nous utilisons une base de données SIG qui fournit les modèles 3D bruts de bâtiments géo-référencés, des vidéos desquelles sont extraites des images RVB pour le texturage ainsi que des images de luminance pour l'extraction et le suivi de points, et pour finir nous utilisons des mesures GPS qui sont enregistrées simultanément avec le flux vidéo, et qui fournissent une première approximation pour la géo-localisation des différentes images. Nous rappelons ci-dessous certaines particularités des données GPS et SIG.

**Gps.** Le GPS (*Global Positioning System*) donne des mesures de positionnement mais avec une précision limitée. Les signaux satellites utilisés sont généralement réfléchis sur les murs, montagnes, etc., et ralentis par les conditions atmosphériques, le tout menant à une précision annoncée de 5 mètres dans 95% des cas. En vue d'estimer la variation de l'erreur sur les mesures GPS en fonction du temps, nous avons acquis des mesures sur un point fixe, dans des conditions difficiles (au pied d'un grand bâtiment, sous un temps couvert), durant environ 10 minutes. La figure 1 montre les variations de la position estimée de ce point fixe, décomposée en *easting* ($X$), *northing* ($Y$) et altitude ($Z$) dans le système de coordonnées UTM. Les valeurs sont centrées sur leur moyenne sur le graphe. Comme on peut le voir, l'écart type en altitude est beaucoup plus important ($\sigma_Z = 14.02m$) et donc moins fiable que celui dans le plan horizontal ($\sigma_X = 3.92m$, $\sigma_Y = 5.05m$). Les mesures GPS ne fournissent donc qu'une estimation peu précise du chemin initial emprunté par la caméra.

**Sig.** L'acronyme SIG, signifiant *Système d'Information Géographique*, se réfère à une collection de tous type d'information géographique géo-référencée. Dans notre cas, nous utilisons une base de données où chaque bâtiment est décrit par son altitude, sa hauteur, et son empreinte au sol exprimée comme une liste fermée de points 2D, dont les

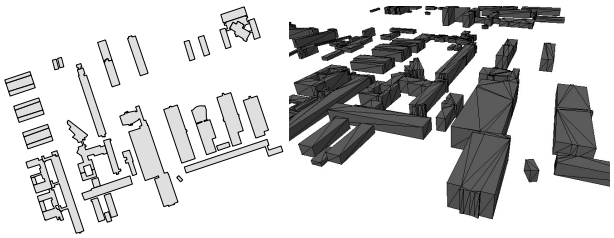


FIG. 2 – Représentation SIG utilisée comme modélisation initiale des bâtiments

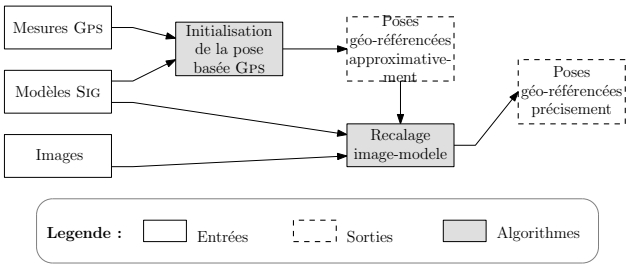



FIG. 3 – Principe du recalage.

coordonnées $XY$ sont données dans le repère UTM. Cette base fournit donc une approximation de la géométrie de la scène, les bâtiments étant modélisés par des polyèdres simples (voir figure 2). On peut déjà voir que ces modèles ne contiennent donc aucune information sur la géométrie locale des façades (fenêtre, portes, toits, etc.) ni sur les textures des bâtiments. C'est la raison pour laquelle on introduit la vidéo pour raffiner ces modèles.

# 4 Recalage des données

## 4.1 Principe général

Le principe général de la méthode proposée est présenté sur la figure 3. La première étape est un recalage initial de la caméra à l'aide des données GPS. Elle fournit une estimation grossière de la position et de l'orientation de la caméra dans le repère géo-référencé du SIG, pour chaque image de la séquence vidéo. La deuxième étape consiste à raffiner cette estimation à l'aide des données image. Pour cela des primitives images sont mises en correspondance avec les modèles 3D du SIG. On cherche alors les paramètres caméra qui permettent de réaliser la superposition du modèle SIG lorsqu'il est projeté sur les images de la vidéo.

## 4.2 Initialisation de la pose basée Gps

L'acquisition GPS fournit une position géographique (latitude/longitude/altitude) exprimée dans un repère géographique universel. Cette position est tout d'abord convertie en coordonnées $(X, Y, Z)$ dans le système géo-référencé UTM du SIG par les équations issues de [17]. Les coordonnées $(X, Y)$ ainsi obtenues sont interpolées linéairement pour fournir la position horizontale de la caméra pour chaque instant temporel correspondant à une image de la séquence vidéo. La coordonnée verticale $Z$ fournie par le

GPS étant très peu fiable, une meilleure estimation est donnée par une initialisation arbitraire à une altitude constante (1,5 m) au dessus du sol. Les données SIG ne contenant pas la surface du sol, celle-ci est modélisée par interpolation par triangulation de Delaunay sur les sommets au sol des bâtiments. On obtient ainsi pour chaque image de la séquence une position $p_t = (X_t, Y_t, Z_t)$. Enfin, l'orientation de la caméra est également estimée à partir des données de position fournies par le GPS, en faisant l'hypothèse que l'axe de vue est parallèle à la trajectoire de la caméra, i.e. à l'instant $t$, l'orientation de la caméra est donnée par $(p_{t+1} - p_t)$.

## 4.3 Recalage Sig et vidéo

La trajectoire grossière obtenue par les données GPS est ensuite raffinée à l'aide des données vidéo. Le recalage entre données vidéo et modèle GIS consiste à déterminer les paramètres caméra qui permettent de superposer le modèle SIG avec les images des bâtiments présents dans la vidéo. Pour la première image de la séquence, le recalage est réalisé à l'aide d'une procédure semi-automatique. Pour les images suivantes, le recalage est réalisé automatiquement par extraction et suivi de points d'intérêt et par asservissement visuel virtuel. Dans la suite de cette section, nous donnons les bases théoriques de l'asservissement visuel virtuel puis nous détaillons l'étape de recalage pour la première image et l'étape de suivi automatique pour les images suivantes.

**Fondements théoriques : modèle de caméra.** Le modèle de projection perspective est utilisé (on suppose corrigées ou négligeables les distorsions radiales). Un point 3D de coordonnées homogènes $\mathbf{P}$ se projette dans l'image au point 2D de coordonnées homogènes $\mathbf{p}$ données par :

$$\mathbf{p} = \mathbf{K}.{}^c\mathbf{M}_o.\mathbf{P} \tag{1}$$

$$\text{avec} \quad \mathbf{K} = \begin{bmatrix} \frac{f}{p_x} & 0 & u_0 \\ 0 & \frac{f}{p_y} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{et} \quad {}^c\mathbf{M}_o = \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix}$$

où $f_x$ et $f_y$ représentent la focale exprimée en largeur et hauteur de pixels, et $[u_0\, v_0]^\top$ sont les coordonnées image du point principal. La pose de la caméra ${}^c\mathbf{M}_o$ est définie par la matrice d'orientation $3\times3$ $\mathbf{R}$ et le vecteur de position $\mathbf{t}$.

**Fondements théoriques : asservissement visuel virtuel.** L'alignement entre le projeté d'un objet 3D et l'image de ce même objet a été largement étudié dans le domaine de la vision par ordinateur et de la robotique. On notera par exemple les approches de Dementhon (POSIT [4], puis SoftPOSIT [3]) ou de Lepetit [22]. Marchand a proposé une méthode de recalage basée sur l'asservissement visuel [10], qui consiste à estimer la pose de l'objet observé dans l'image, en modifiant la pose d'une caméra virtuelle observant le modèle 3D. La solution proposée pour estimer le trajet de la caméra et recaler le modèle 3D GIS avec les image de la séquence vidéo est basée sur une telle approche d'asservissement visuel virtuel.

Dans le cas général, l'estimation de pose peut être considérée comme un problème d'estimation non-linéaire, faisant intervenir un ensemble de primitives 3D et leur projections 2D dans le plan image. L'objectif est de minimiser l'erreur de projection (dans l'image) entre les données observées $\mathbf{s}^*$ et la position de ces même données $\mathbf{s}$, calculée par projection sur le plan image des primitives 3D correspondantes. La pose de la caméra ${}^c\mathbf{M}_o$ est alors estimée par :

$${}^c\tilde{\mathbf{M}}_o = \text{argmin}(\|\mathbf{s}({}^c\mathbf{M}_o) - \mathbf{s}^*\|^2) \tag{2}$$

La résolution est itérative : la pose est initialisée par ${}^{ci}\mathbf{M}_o$, et raffinée jusqu'à convergence à ${}^{cf}\mathbf{M}_o$ par la loi de commande :

$$v = -\lambda(\mathbf{L_s})^+(\mathbf{s}({}^c\mathbf{M}_o) - \mathbf{s}^*) \tag{3}$$

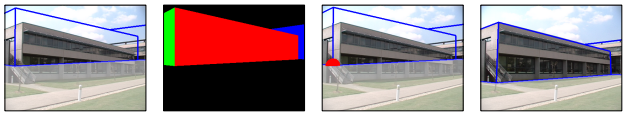
où $v$ est un vecteur définissant la pose et fonction de $\mathbf{R}$ et $\mathbf{t}$, $\lambda$ est un scalaire et $\mathbf{L_s}$ est le Jacobien de la fonctionnelle à minimiser. Cette méthode est générique par rapport au type des primitives utilisées, à condition que l'erreur puisse être calculée à partir des données image.

Dans notre cas, les primitives utilisées sont un ensemble de points d'intérêt. $\mathbf{s}^*$ représente alors un ensemble de points 2D $\mathbf{p}_i$, et $\mathbf{s}$ est l'ensemble des points 3D correspondants $\mathbf{P}_i$ projetés dans l'image pour une pose donnée ${}^c\mathbf{M}_o$ et pour une matrice des paramètres internes $\mathbf{K}$ donnée. Si $N$ est le nombre de points considérés, alors $\mathbf{s}^* = \{\mathbf{p}_i | i \in 1 \ldots N\}$ et $\mathbf{s} = \{\mathbf{K}{}^c\mathbf{M}_o\mathbf{P}_i | i \in 1 \ldots N\}$. De ce fait, si nous pouvons produire un ensemble de correspondance entre points 2D dans l'image courante et points 3D du modèle de la base SIG, alors il est possible d'estimer la pose de la caméra pour cette image, exprimée dans le repère du SIG.

La précision de la pose estimée par asservissement visuel virtuel est très sensible aux erreurs introduites par l'extraction des primitives (bruit dans les images, variations d'illumination, occultations, position 3D des points du modèle, ...). Nous utilisons de ce fait une estimation robuste au niveau de la loi de contrôle, donnée sous la forme d'un M-estimateur permettant de quantifier la confiance associée à chaque information géométrique utilisée. La loi de de commande s'écrit alors :

$$v = -\lambda(\mathbf{DL_s})^+\mathbf{D}(\mathbf{s}({}^c\mathbf{M}_o) - \mathbf{s}^*) \tag{4}$$

où $\mathbf{D} = diag(w_1, w_2, \ldots, w_N)$ est une matrice diagonale contenant les poids $w_i$ correspondant à la mesure de confiance associée à chaque information visuelle. Ils sont calculés par la fonction robuste de Cauchy. Pour assurer qu'un nombre suffisant de primitives n'est pas rejeté par l'estimateur robuste, on vérifie que la matrice $\mathbf{DL_s}$ est de rang plein, (*i.e.* rang 6 puisque la pose a 6 degrés de liberté : 3 pour la position et 3 pour l'orientation), grâce à la décomposition SVD utilisée lors du calcul de la pseudo-inverse $(\mathbf{DL_s})^+$.



1: Pose approximative 2: Color coding 3: Détection / m.e.c. des points visibles 4: Calcul de la pose

FIG. 4 – Calcul de la pose pour la première image

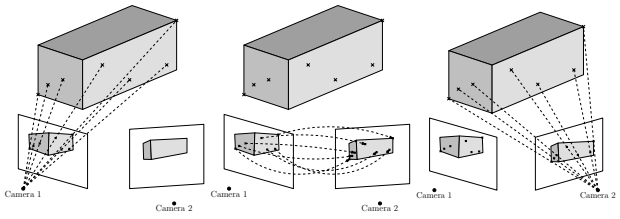
**Calcul de la pose pour la première image.** Nous décrivons ici la méthode semi-automatique permettant de recaler le SIG avec la première image de la séquence. Les différentes étapes de la procédure sont illustrées sur la figure 4.

A ce point, seules une position et une orientation approximatives de la caméra sont connues pour la première image. L'utilisateur corrige tout d'abord ces valeurs grâce à une interface OpenGL, qui affiche à la fois l'image et la projection du modèle SIG des bâtiments visualisés. Ces derniers sont rendus en mode filaire à l'aide d'une caméra virtuelle. L'utilisateur translate et oriente cette caméra virtuelle manuellement de telle sorte que le modèle projeté soit visuellement proche du contenu de l'image. La pose initiale de la caméra est raffinée en utilisant des correspondances 2D-3D. Les seuls points 3D qui peuvent être extraits de manière fiable des modèles SIG sont les coins des bâtiments (*i.e.* les points au niveau du sol et du toit qui appartiennent à l'empreinte des bâtiments). Ceux qui sont visibles dans le rendu en filaire sont automatiquement détectés en utilisant une procédure de *color coding*. Une version polygonale de la base SIG est stockée dans la mémoire graphique, et à chaque façade est associée une unique couleur RVB :

$$R = b_i \div 256 \quad V = b_i \, mod \, 256 \quad B = f_i$$

où $b_i$ est l'index du bâtiment dans la base, et $f_i$ l'index de la façade dans l'empreinte du bâtiment. La couleur noire est réservée pour dénoter l'absence de bâtiment. Ce modèle coloré est rendu dans le tampon arrière d'OpenGL à la pose courante approximative. Lire le contenu de ce tampon permet d'identifier quelles sont les façades visualisées, et le couple $(b_i, f_i)$ donne alors un accès direct dans la base SIG aux coordonnées 3D des coins des façades. Ceux qui se projettent en dehors de l'image ou qui sont occultés par une autre façade sont éliminés automatiquement. Pour chaque point 3D $X_i$ sélectionné, l'interface affiche un marqueur dans le modèle SIG, et attend que l'utilisateur fournisse en cliquant sur l'image son correspondant 2D $x_i$. Une fois que toutes les correspondances 2D-3D sont effectuées, la pose est calculée automatiquement en utilisant un algorithme d'asservissement visuel virtuel à partir de l'équation 3. Au moins quatre de ces correspondances sont nécessaires pour calculer la pose, le résultat étant plus pertinent dans le cas de points non coplanaires.

**Suivi de la pose.** Une fois que la pose a été calculée pour la première image de la vidéo, recaler le modèle SIG avec les images suivantes devient un problème de suivi, qui est traité ici de façon automatique en utilisant également une




(a) (b) (c)

FIG. 5 – Suivi de la pose au long de la vidéo

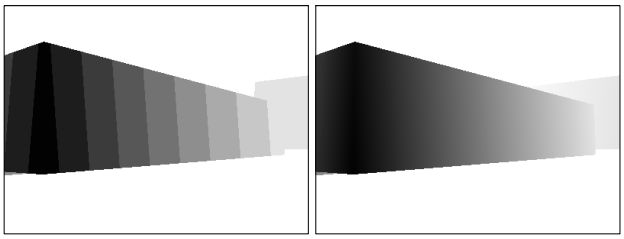
approche basée asservissement. Soit $I_t$ une image pour laquelle le recalage avec le modèle SIG est effectué, et $I_{t+1}$ l'image suivante pour laquelle nous cherchons à calculer la pose. Comme pour le recalage de la première image, nous avons besoin pour cette image de correspondances 2D-3D. Celles-ci sont effectuées en se basant sur un schéma de *transfert de points* qui utilise les données extraites de $I_t$. La procédure complète est illustrée sur la figure 5.

Tout d'abord, des points 2D sont extraits de l'image $I_t$. Pour l'extraction et le suivi de points d'intérêt, nous utilisons une implémentation du tracker de Kanade-Lucas-Tomasi (KLT)[2] [20]. Etant donné que tous les points extraits n'appartiennent pas à un bâtiment, ils sont classifiés en points leur appartenant ou non. Aucune estimation explicite de la profondeur des points 2D extraits n'est effectuée pour savoir s'ils intersectent ou non le modèle SIG. Celle qui leur est assignée est celle du tampon de profondeur, qui a déjà été calculée par OpenGL pour afficher le modèle 3D recalé avec l'image $I_t$ (voir figure 5(a)). Si la valeur assignée est nulle alors le point est classifié comme n'appartenant pas à une façade, et inversement. Nous avons donc à ce point des correspondances 2D-3D pour l'image $I_t$, *qui est déjà recalée avec le modèle*. On voit d'ailleurs que l'on n'est plus limité à l'utilisation des coins de bâtiments pour avoir une information 3D, puisque que la correspondance image-modèle donne potentiellement une information de profondeur pour tout pixel se situant à l'intérieur de la projection du modèle.

Dans l'optique de mieux conditionner le suivi, et puisque la majorité des points appartiennent souvent à une unique façade, le modèle estimé du sol est également utilisé pour introduire de nouvelles correspondances de primitives qui sont situées globalement sur un plan orthogonal aux plans de façades.

De plus, il faut prendre en compte le fonctionnement du tampon de profondeur lors de l'estimation des points 3D pour que leur mesure soit suffisamment précise. Dans notre cas, peu de précision est allouée pour les façades si l'on utilise des valeurs génériques pour les plans de clipping. Nous laissons alors le soin à l'utilisateur de définir la distance au plan de clipping lointain ($\pi_f$), mais celle du plan de clipping proche ($\pi_n$) est déplacée automatiquement à la valeur correspondant au point de bâtiment visible le plus proche de la caméra. Une comparaison des valeurs stockées dans

[2] http ://www.ces.clemson.edu/∼stb/klt/



(a) $\pi_n = 1$ et $\pi_f = 75$ (b) $\pi_n = 20$ et $\pi_f = 75$

FIG. 6 – Influence des plans de clipping sur l'estimation de la profondeur

le tampon est représentée sur la figure 6, pour une valeur fixe de $\pi_f$ et différentes valeurs de $\pi_n$. Comme on peut le voir, plus $\pi_n$ est éloigné de la caméra, plus la précision allouée aux points 3D des façades sera grande (le point le plus proche du bâtiment est situé à environ 23 mètres sur la figure).

En utilisant le KLT, on suit les points d'intérêt entre les images $I_t$ et $I_{t+1}$ (voir figure 5(b)). Si $\mathbf{x}_t$ représente l'ensemble des points 2D extraits de $I_t$ et $\mathbf{X}$ leur position 3D correspondante, puisque l'on connaît des correspondances entre $\mathbf{x}_t$ et $\mathbf{x}_{t+1}$ on peut calculer des correspondances 2D-3D pour $I_{t+1}$, entre $\mathbf{x}_{t+1}$ et $\mathbf{X}$. En les utilisant dans l'équation 4 on peut alors calculer la pose de la caméra pour $I_{t+1}$ (figure 5(c)). Cependant, le tracker KLT perd des points au cours du recalage. Pour faire face à ce problème, on introduit une mesure sur le nombre de points perdus. Si à un instant $t$ on estime avoir perdu trop de points (typiquement 60%), on extrait de nouveaux points d'intérêt en lisant leur profondeur de nouveau dans le tampon de profondeur en utilisant la dernière image recalée ($I_{t-1}$). On garde cependant les points que l'on n'avait pas perdu, et on contraint les nouveaux points à être suffisamment distants dans l'image des anciens.

# 5 Expérimentations

Nous présentons dans cette section des expérimentations de notre méthode sur plusieurs façades de bâtiments. Après avoir donné quelques détails sur le calibrage de la caméra, des résultats de recalage sont présentés pour deux séquences de test. Les résultats présentés ont été obtenus sur un Pentium IV cadencé à 2.5 GHz avec 512 Mo de RAM, et en utilisant une carte graphique nVidia Quadro2 EX pour le rendu.

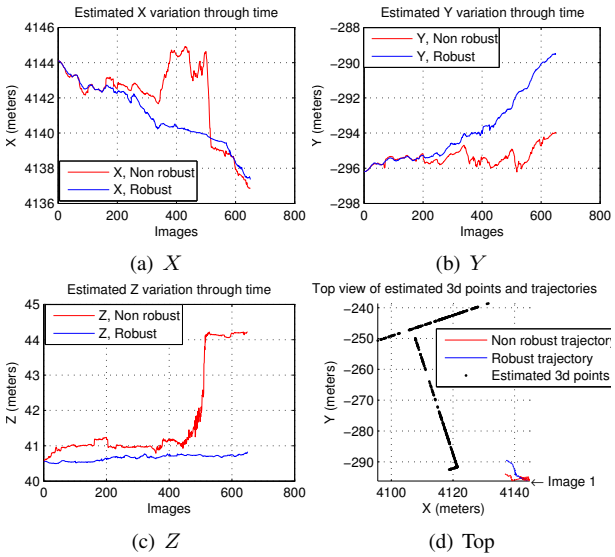
**Calibrage de la caméra.** Dans notre contexte, et grâce au rapport entre la taille des pixels et celle des objets visualisés, nous n'avons pas besoin d'un calibrage extrêmement précis de la caméra (voir également [7]). Le centre de projection est initialisé à $[0\,0]^\top$, et pour la focale nous pouvons utiliser celle donnée par les paramètres constructeur ou des données EXIF[3] contenues dans les images, comme dans [16].

[3] *Exchangeable Image File Format*

**Résultats de recalage.** La séquence de test présentée dans cette section est composée d'images basse résolution ($400\times300$ pixels). Elle a été acquise avec une caméra vidéo numérique du commerce, et contient 650 images où l'on voit plusieurs façades. Le mouvement de la caméra est générique, et ne vise à suivre aucune façade en particulier, ce qui rend le tracking d'autant plus difficile. Deux résultats de suivi sont présentés. Tout d'abord une version simple du tracker a été utilisée (on parlera de version *non robuste* par la suite). Seuls les points de façade extractibles sont pris en compte, aucune optimisation du z-buffer n'est calculée, et la version originale de la loi de commande (équation 3) est utilisée. Bien que cet algorithme donne de bons résultats quand la caméra reste pointée sur l'objet à suivre, une dérive importante apparaît quand cet objet est seulement partiellement visible, disparaît dans quelques images, ou quand par exemple de nombreuses spécularités sont présentes dans la scène. Nous présentons donc des résultats de suivi utilisant la version *robuste* du tracker présentée dans la section 4.3. Une fois que les correspondances sont données pour la première image, le calcul de la pose est effectué en approximativement 0,2 secondes. Des résultats de suivi sont indiqués sur la figure 7. Les positions $(X, Y, Z)$ estimées de la caméra sont données pour chaque version du tracker sur les figures 7(a) 7(b) 7(c). Une vue du dessus de la trajectoire estimée de la caméra dans en coordonnées UTM est également illustrée avec les positions estimées des points 3D sur la figure 7(d). Enfin, un rendu du modèle SIG en surimpression avec les images correspondantes est illustré sur la figure 8 (le modèle 3D estimé du sol est représenté en gris transparent). Le suivi est calculé en 441 secondes pour la version non robuste, et en 637 secondes pour la version robuste. On peut noter que les différentes améliorations apportées rendent le suivi beaucoup moins sensible à la dérive que dans la version classique (non robuste) de l'algorithme d'asservissement visuel. Ceci est particulièrement visible par exemple pour la variation de l'altitude estimée (7(d)), qui n'est pas supposée varier de plus de quelques centimètres. On notera toutefois que, bien que grandement diminuée, une dérive dans l'estimation de la pose est encore légèrement visible, et doit être supprimée.

Des résultats de suivi sont également présentés sur la figure 9. Dans ce cas, une unique façade est visualisée et suivie dans la séquence, et aucun objet non modélisable (comme des piétons, voitures, etc.) ne vient masquer tout ou partie du bâtiment. Dans ce cas, les versions robustes et non robustes donnent des résultats similaires, étant donné que l'on se place ici dans un cas quasi-idéal.

Les résultats présentés sur la figure 10 décrivent un scénario similaire, à l'exception du fait qu'une voiture masque une partie de la façade. Les points 2D suivis lui appartenant sont donc toujours associés à de fausses coordonnées 3D, ce qui met en échec la version non robuste du tracker (qui est complètement perdu après la $110^e$ image), au contraire du tracker robuste qui assigne des poids suffisamment faibles à ces fausses correspondances 2D-3D pour




(a) $X$ (b) $Y$

(c) $Z$ (d) Top

FIG. 7 – Résultats de suivi pour la séquence *Ifsic*

suivre correctement la façade tout au long de la séquence. Des vidéos de ces résultats de suivi sont par ailleurs disponibles en ligne [4]

# 6 Conclusion et travaux futurs

Nous avons présenté dans cet article une méthodologie permettant de mettre en correspondance différents types de données, en tant qu'étape obligatoire à une reconstruction à grande échelle de modèles urbains, en interprétant des mesures GPS par rapport à une base de données SIG pour donner une première approximation de la position de la caméra ayant acquis des données au sol, puis en raffinant l'estimée de la pose de cette caméra en utilisant des algorithmes d'asservissement visuel virtuel appropriés. Nous calculons alors une position et une orientation géo-référencées pour chaque image de la vidéo, desquelles on peut alors extraire des informations pertinentes pour la reconstruction de la géométrie locale des façades ou pour l'extraction de leur texture.

Cependant, des améliorations peuvent encore être apportées à cette méthode. Nous voudrions tout d'abord supprimer la partie manuelle du processus pour la première image, en développant une procédure automatique qui fasse ce premier recalage. De plus, avec une telle procédure, nous pourrions réduire la dérive lors du processus de recalage en l'appelant en fonction de critères d'erreurs à définir. De tels travaux sont actuellement en cours d'étude. Ils se basent sur une estimation de la pose de la première caméra par *ego-motion*, puis une mise en correspondance de lignes extraites des images avec la projection de segments issus du SIG. Il sera alors intéressant de comparer ces travaux avec ceux de Drummond et al. [13].

Enfin, dans un futur proche, nous envisageons d'exploiter ces images géo-référencées avec les modèles SIG pour raffiner leur géométrie et leurs textures.

[4] http ://www.irisa.fr/temics/staff/sourimant/tracking

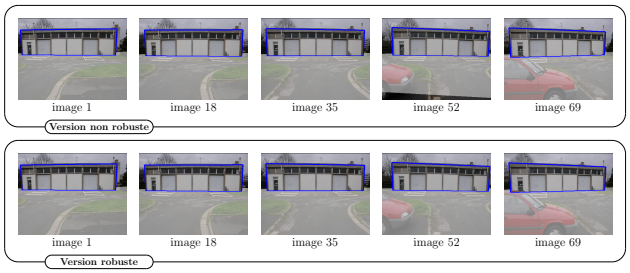



FIG. 9 – Résultats de suivi avec modèle 3D en surimpression (*Beaulieu*)

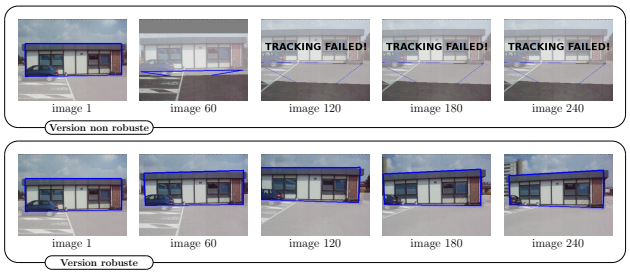



FIG. 10 – Résultats de suivi avec modèle 3D en surimpression (*Beaulieu2*)

# Références

[1] A. Akbarzadeh, J.-M. Frahm, P. Mordohai, B. Clipp, C. Engels, D. Gallup, P. Merrell, M. Phelps, S. Sinha, B. Talton, L. Wang, Q. Yang, H. Stewéius, R. Yang, G. Welch, H. Towles, D. Nisté, and M. Pollefeys. Towards urban 3d reconstruction from video. In *Third International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT)*, 2006.

[2] P. E. Debevec, C. J. Taylor, and J. Malik. Modeling and rendering architecture from photographs : a hybrid geometry- and image-based approach. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, 1996.

[3] D. DeMenthon, P. David, and H. Samet. Softposit : An algorithm for registration of 3d models to noisy perspective images combining softassign and posit. Technical report, Center for Automation Research, 2001.

[4] D. DeMenthon and L. S. Davis. Model-based object pose in 25 lines of code. In *European Conference on Computer Vision*, 1992.

[5] A. F. Elaksher, J. S. Bethel, and E. M. Mikhail. Reconstructing 3d building wireframes from multiple images. In *Proceedings of the ISPRS Commission III Symposium on Photogrammetric Computer Vision*, 2002.

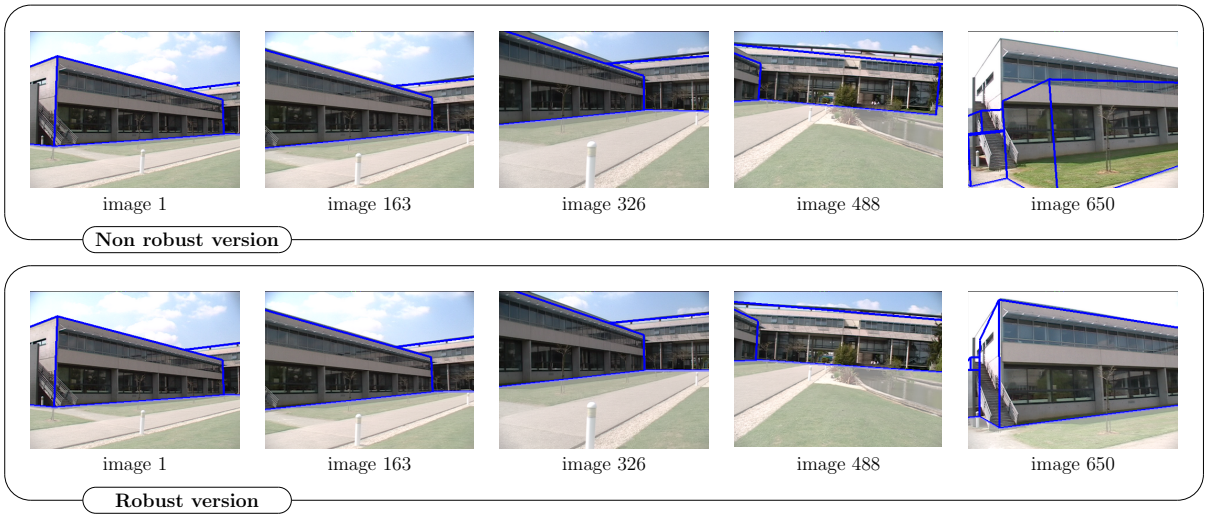



FIG. 8 – Résultats de suivi avec modèle 3D en surimpression (*Ifsic*)


[6] A. Frueh and A. Zakhor. Data processing algorithms for generating textured 3d building facade meshes from laser scans and camera images. *IJCV*, 2005.

[7] J.-F. Vigueras Gomez, G. Simon, and M.-O. Berger. Calibration errors in augmented reality : A practical study. In *ISMAR '05 : Proceedings of the Fourth IEEE and ACM International Symposium on Mixed and Augmented Reality*, 2005.

[8] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN : 0521540518, 2004.

[9] A. D. Hofman, H.-G. Maas, and A. Streilein. Knowledge-based building detection based on laser scanner data and topographic map information. In *Proceedings of the ISPRS Technical Commission III Syposium on Photogrammetric Computer Vision*, 2002.

[10] E. Marchand. *Commande d'une caméra réelle ou virtuelle dans des mondes réels ou virtuels*. Habilitation à diriger les recherches, Université de Rennes 1, Mention informatique, 2004.

[11] M. Pollefeys and L. V. Gool. Visual modeling : from images to images, 2002.

[12] K. Quennesson and F. Dellaert. Rao-blackwellized importance sampling of camera parameters from simple user input with visibility preprocessing in line space. In *3DPVT*, 2006.

[13] T.W. Reitmayr, G. ; Drummond. Going out : robust model-based tracking for outdoor augmented reality. *IEEE/ACM International Symposium on Mixed and Augmented Reality*, 2006.

[14] G. Schindler, P. Krishnamurthy, and F. Dellaert. Line-based structure from motion for urban environments. In *3DPVT*, 2006.

[15] K. Schindler and J. Bauer. A model-based method for building reconstruction. In *Proceedings of the ICCV Workshop on Higher-Level Knowledge in 3D Modeling and Motion*, 2003.

[16] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism : exploring photo collections in 3d. In *SIGGRAPH '06 : ACM SIGGRAPH 2006 Papers*, 2006.

[17] J. P. Snyder. *Map projections - A working manual*. US Geological Survey Professional Paper 1395, 1987.

[18] I. Suveg and G. Vosselman. Localization and generation of building models. In *Proceedings of the ISPRS Technical Commission III Syposium on Photogrammetric Computer Vision*, 2002.

[19] S. Teller, M. Antone, Z. Bodnar, M. Bosse, S. Coorg, M. Jethwa, and N. Master. Calibrated, registered images of an extended urban area. *Int. J. Comput. Vision*, 2003.

[20] C. Tomasi and T. Kanade. Detection and tracking of point features. Technical report, Carnegie Mellon University, 1991.

[21] F. Tupin and M. Roux. Detection of building outlines based on the fusion of sar and optical features. *PandRS*, 2003.

[22] L. Vacchetti V. Lepetit, D. Thalmann, and P. Fua. Fully automated and stable registration for augmented reality applications. *International Symposium on Mixed and Augmented Reality*, 2003.

[23] T. Werner and A. Zisserman. Model selection for automated architectural reconstruction from multiple views. In *Proceedings of the British Machine Vision Conference*, 2002.