

Reconstruction tridimensionnelle de scènes structurées par des contraintes géométriques

3D Reconstruction of Structures Defined by Geometrical Constraints

S. Cornou^{1,2}, M. Dhome¹ et P. Sayd².

¹LASMEA, UMR 6602 du CNRS – Université Blaise Pascal- Clermont-Ferrand

²CEA Saclay – DRT/LIST Lab. Calculateurs Embarqués et Images

sebastien.cornou@cea.fr, patrick.sayd@cea.fr, dhome@lasmea.univ-bpclermont.fr

Résumé

Dans cet article, nous présentons une approche supervisée de reconstruction 3D d'objets à partir de plusieurs vues non calibrées. Dans cette approche, l'utilisateur définit un modèle de l'objet à partir de règles géométriques élémentaires et localise les projections des sommets 3D dans les images. En parallèle à cette modélisation, un graphe orienté décrivant les contraintes existant entre les différents sommets du modèle 3D est construit. Par la suite, les paramètres inconnus de ce graphe et les paramètres associés à chaque image sont estimés au moyen d'une minimisation non-linéaire de type ajustement de faisceaux adapté à ce problème de reconstruction de modèle contraint.

Du fait de la modélisation choisie, les contraintes affectées au modèle sont toujours parfaitement vérifiées. De plus, la représentation des données permet de reconstruire indépendamment différentes parties de l'objet puis de fusionner l'ensemble des modèles 3D indépendants obtenus afin de construire un modèle complet et multi-échelle. Un exemple de reconstruction d'un bâtiment réalisée à partir d'une séquence d'images réelles est présenté en fin d'article.

Mots Clef

Vision 3D, auto-calibrage, reconstruction interactive, contraintes, ajustement de faisceaux.

Abstract

We present a supervised approach to recover 3D models of objects from multiple uncalibrated views. With this method the user defines the 3D model of the object with the help of elementary and intuitive geometric constraints and matches 3D vertices in the images. At the same time, a graph describing relationships between vertices is built. Then, unknown parameters of this graph and the unknown parameters associated to the cameras are estimated non-linearly through a bundle adjustment adapted to this problem of constrained model reconstruction.

This method asserts that geometric rules are perfectly respected. This approach is used to recover independently 3D parts of the object with suitable images. Then, all these

independent 3D elements are merged to obtain a full multi-scale model. An example is given on real images representing a building.

Keywords

3D vision, auto-calibration, interactive reconstruction, constraints, bundle adjustment.

1. Introduction

Dans ce travail, notre objectif est de proposer une méthode interactive de reconstruction d'objets structurés. Notre contribution principale est une modélisation simple d'emploi basée sur la définition de quelques contraintes élémentaires. Cette méthode permet d'obtenir un modèle 3D caractérisé par le plus petit nombre de paramètres possibles, respectant parfaitement les contraintes indiquées, et permettant de fusionner simplement les modèles entre eux pour obtenir un modèle plus complet.

L'obtention d'un modèle tridimensionnel à partir d'images est un domaine actif de la vision artificielle. De nombreuses méthodes existent déjà, et s'organisent en trois ensembles : les approches s'appuyant sur des primitives de bas-niveaux, les méthodes utilisant des primitives géométriques complexes et les approches hybrides combinant les deux précédentes.

Les méthodes basées sur l'exploitation de primitives de bas-niveaux (points, ...) décrivent le bâtiment comme un ensemble de primitives élémentaires. La plupart du temps, des appariements automatiques de points homologues sont réalisés entre les images de la séquence. Ces informations sont utilisées de définir les attitudes et les paramètres des prises de vue ainsi que les positions des primitives 3D associées aux points homologues détectés. L'appariement automatique de primitives est un domaine très étudié (cf. [10] [12] [13]) (notamment dans le cas des points) et est particulièrement performant dans le cas où les points de vue sont proches les uns des autres. Cette méthode est utilisée par Pollefeys [15] sur des séquences vidéo et par Werner et Zisserman [20] sur des images statiques. Schaffalitzky et Zisserman [17] ont traité le cas des séquences d'images avec des vues désordonnées dont les points de vue sont éloignés les uns des autres. Le résultat de ces méthodes est un nuage de primitives 3D

(souvent des points !) qui doit être transformé en une surface pour être exploité. Différentes stratégies existent afin de lever cette difficulté, Werner et Zisserman [20] segmentent automatiquement le nuage de primitives 3D en plan, Pollefeys [15] effectue un maillage et Morris et Kanade [16] utilisent les informations disponibles dans l'image (texture) afin de déterminer la triangulation la plus pertinente.

Dans l'approche par primitives géométriques complexes, l'utilisateur décrit le modèle de bâtiments au moyen de primitives paramétriques définies dans une bibliothèque de formes (cube, cylindre...). Ces primitives 3D sont localisées dans les images et une minimisation de la distance entre les primitives reprojétées dans les images et leurs positions 2D précédemment identifiées permet de définir l'ensemble des paramètres associés à cette scène. Ces approches, utilisées par Debevec et al. [5] ainsi que Lang et Förstner [9], ont l'avantage de permettre une définition implicite de règles géométriques (définir un cube implique la présence d'orthogonalités et d'égalités de distance). Toutefois, ces méthodes sont limitées par le nombre de primitives définies dans leurs bibliothèques de formes.

Enfin, les méthodes hybrides tentent de combiner les avantages des deux méthodes précédentes. Cipolla et Robertson présentent dans [2] une approche basée sur des estimateurs statistiques. Bartoli et Sturm [1] suggèrent une technique permettant d'introduire des contraintes de co-planarité, Grossman et Santos-Victor [7] ainsi que Wilczkowiak et al. [21] introduisent itérativement les contraintes géométriques de sorte à ne résoudre que des systèmes linéaires à chaque itération du processus de reconstruction. Notre approche utilise une description plus générale des contraintes et un ajustement de faisceaux ne nécessitant pas d'initialisation des paramètres de pose des prises de vues pour l'estimation de l'ensemble des paramètres liés à la scène. De plus, notre méthode permet de reconstruire le modèle à plusieurs échelles et d'obtenir ainsi des reconstructions précises et détaillées pour certains éléments tout en modélisant d'autres éléments à des résolutions moindres.

Dans cet article, nous décrivons tout d'abord la phase de modélisation. Par la suite nous présentons l'ajustement de faisceaux utilisé. La troisième partie ouvre des perspectives d'amélioration en apportant de nouvelles informations au moyen d'appariements de points homologues. Pour finir, nous présentons les résultats obtenus sur une séquence réelle.

2. Modélisation contrainte

Dans cette partie, nous expliquons notre mode de description des contraintes. Tout d'abord, nous présentons un aperçu des différentes contraintes géométriques définies. Par la suite, nous décrivons la structure de données utilisée avec notamment les relations existant entre les contraintes et les sommets 3D du modèle. Enfin, nous expliquons la méthode permettant de fusionner simplement deux modèles afin d'obtenir une reconstruction plus complexe.

2.1 Les contraintes géométriques élémentaires

Notre approche décrit l'objet comme un ensemble de points 3D. Des règles géométriques sont employées afin d'organiser et de structurer ce nuage de points. C'est une différence majeure avec les approches basées sur des primitives élémentaires qui obtiennent en fin de reconstruction un nuage de points dépourvu d'une organisation globale. Dans notre méthode, chaque nouveau point est défini au moyen d'une règle géométrique par rapport aux points 3D déjà existant (appelés *antécédents*). Les contraintes géométriques associant le point à ses antécédents sont décrites dans le Tableau 1. Ce tableau indique les équations strictement vérifiées pour chaque règle ainsi que les degrés de liberté laissés au nouveau sommet. Ces degrés de liberté correspondent aux paramètres devant être estimés au moyen des informations saisies dans les images. Le nombre total de paramètres décrivant de modèle est la somme des degrés de liberté associés à chaque sommet.

Nom	Antécédents	Equations contraintes	Degrés de liberté
Point libre	Aucun	$\underline{A}(x,y,z)$	3 (coordonnées)
Egalité vectorielle	A,B,C	$\underline{CD}=\underline{AB}$	0
Parallélisme	A,B,C	$\underline{CD}=\lambda \underline{AB}$	1 (1 longueur)
Orthogonalité simple	A,B	$(\underline{AC})\perp(\underline{AB})$	2 (1 longueur, 1 angle)
Orthogonalité double	A,B,C	$(\underline{AD})\perp(\underline{AB})$ et $(\underline{AD})\perp(\underline{AC})$	1 (1 longueur)
Planarité	A,B,C	$\underline{AD}=\alpha \underline{AB}+\beta \underline{AC}$	2 (coordonnées)
Egalité de distance	A,B,C	$\ \underline{CD}\ =\ \underline{AB}\ $	2 (2 angles)

Tableau 1. Ensemble des contraintes géométriques disponibles. Les degrés de liberté indiquent le nombre de paramètres à estimer au moyen des appariements 3D/images pour déterminer complètement la position du sommet 3D. Les lettres en caractère gras correspondent aux vecteurs et les lettres en soulignées indiquent le point contraint par cette règle.

A titre d'exemple, nous détaillons la mise en oeuvre de la contrainte 'égalité de distance'. Trois antécédents A, B, C sont requis pour calculer la position du point D contraint. La contrainte traduit la relation $\|\underline{AB}\|=\|\underline{CD}\|$. Dans un repère sphérique centré en C, cette contrainte est paramétrée par deux angles θ et φ . Pour θ et φ données, la position du point D s'obtient en calculant :

$$D = C + \|\underline{AB}\| \begin{pmatrix} \cos(\theta) \cos(\varphi) \\ \sin(\theta) \cos(\varphi) \\ \sin(\varphi) \end{pmatrix}$$

2.2 Structure globale du modèle

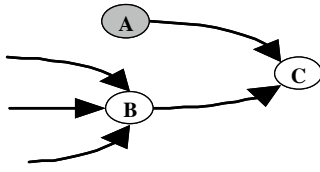


Figure 1. Un graphe décrivant les liens existant entre le sommet C et ses deux antécédents A et B. B possède aussi des antécédents.

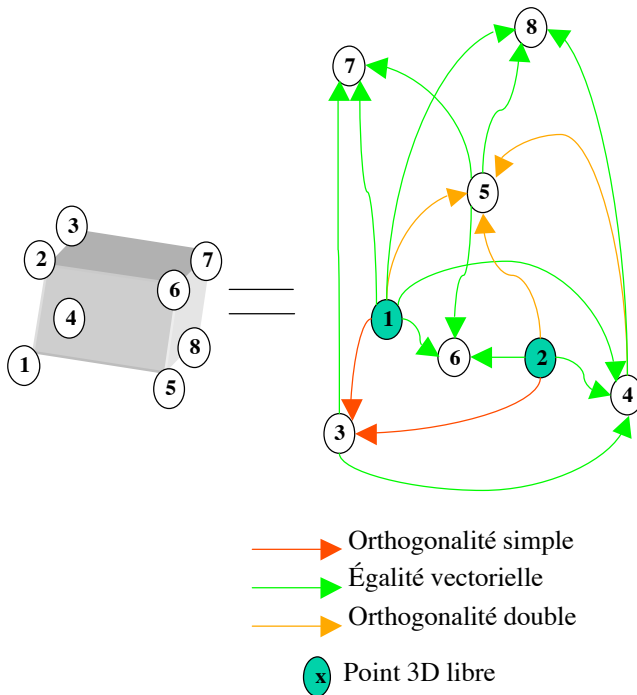


Figure 2. Description d'un parallélépipède au moyen d'un graphe de contraintes. Les chiffres indiquent la correspondance entre les sommets du modèle et ceux du graphe. Les sommets n°1 et n°2 sont la racine de cet arbre et leur position 3D peut être déterminée sans connaissance de celles des autres sommets.

La structure du modèle employée dans notre méthode est un graphe orienté : un nœud représente un sommet 3D et l'ensemble des branches arrivant à ce nœud symbolise la règle géométrique employée pour définir la position 3D de ce sommet. Les nœuds formant l'origine de ces branches sont les antécédents décrits dans le paragraphe précédent. Un point libre n'a pas d'antécédent et sa position 3D peut être calculée sans connaître la position des autres sommets. La Figure 1 décrit les relations entre trois sommets A, B et C. Si les positions 3D des points A et B sont connues alors l'application de la règle reliant C à A et B permet de calculer la position de C (en fonction de la valeur fixée aux paramètres formant les degrés de liberté associés à C). De la même façon, B est positionné en fonction de la position de ses propres antécédents... Le

point A est libre et ne dépend d'aucun autre point. En liant ainsi les différents sommets du modèle, notre méthode permet de modéliser un objet au moyen de contraintes avec un minimum de paramètres. Toutefois, cette description minimale est obtenue au prix d'une interdépendance entre les différents sommets du modèle. Dans la figure 2, nous donnons l'exemple d'un graphe représentant un parallélépipède rectangle. Ce graphe n'est pas l'unique possibilité et sa structure dépend essentiellement des choix de l'utilisateur. Deux points libres (1 et 2) forment la semence du modèle. En fin de modélisation, le modèle comporte 9 degrés de liberté (6 pour la position et l'orientation du modèle et 3 pour les dimensions internes du parallélépipède). Ce graphe est une description d'un objet paramétrique complexe mais reste proche dans sa définition et son utilisation des méthodes basées sur les primitives élémentaires.

2.3 Fusion de différents modèles

Nous venons de décrire notre méthode de modélisation par graphe d'un objet. Pour construire un modèle 3D réaliste d'un objet complexe, le plus simple consiste à reconstruire indépendamment des sous-modèles de l'objet observé en s'appuyant sur des images adéquates, puis à fusionner l'ensemble de ces sous-modèles en un modèle global. Le recalage des modèles implique de les localiser tous dans un même repère. Imaginons que nous souhaitions compléter un modèle M1 (par exemple un plan correspondant à une façade) par un modèle M2 (par exemple le modèle d'une fenêtre reconstruite préalablement avec d'autres images). Notre objectif est de définir le changement de repère entre M1 et M2 (un facteur d'échelle et l'attitude de M2 par rapport à M1). Ceci peut être obtenu en définissant trois points communs dans M1 et M2. L'utilisateur sélectionne trois points dans M2, puis crée dans M1 trois points correspondant à ceux sélectionnés dans M2. Ces trois points sont de nouveaux points dans les graphes de M1 et sont, par conséquent, décrits au moyen des contraintes déjà présentées. A partir de ces trois points communs à M1 et M2, il est possible de définir complètement le passage de M2 vers M1. Le modèle M2 ne subit qu'un changement d'échelle isotrope, une rotation et une translation par conséquent, la précision interne de chacun des modèles est conservée (à un facteur d'échelle près).

3. Ajustement de faisceaux

3.1 Le cas général

Nous présentons, dans un premier temps, le cas de la reconstruction d'un nuage de points 3D libres de toutes contraintes. Connaissant la projection 2D des points 3D dans les images, nous souhaitons définir la position des points 3D et les paramètres de prises de vue (poses et paramètres intrinsèques). Une réponse classique est de réaliser une optimisation non-linéaire appelée ajustement de faisceaux ([6] [8] [11] [18] [19] [22]). L'ajustement de faisceaux consiste rechercher les paramètres d'une

scène qui minimisent la distance entre les points 2D détectés dans les images et la projection des points 3D qui leurs sont associés. Plus précisément, le critère C à minimiser est (i : indice du point 3D, j : indice d'une camera, Pr : matrice de projection, int : paramètres intrinsèques, f : fonction de projection, (Tx, Ty, Tz) : translation, (α, β, γ) : angle d'Euler, δ_{ij} : 0 lorsque le point i n'est pas visible dans l'image j , 1 autrement):

$$C = \min_{\substack{int_j, \\ (Tx, Ty, Tz, \alpha, \beta, \gamma)_j, \\ (x, y, z)_i}} \sum_{i,j} \delta_{ij} \left\| \tilde{p}_{i,j}^{2D} - Pr^j(\tilde{P}_i^{3D}) \right\|$$

$$Pr^j = f(int_j, (Tx, Ty, Tz, \alpha, \beta, \gamma)_j)$$

Avec cette approche la scène est décrite par les paramètres extrinsèques ($6N$) et intrinsèques (kN) des N caméras et par les paramètres (P) associés au modèle 3D. $6N+kN+P$ sont estimés par cette méthode. Cette approche requièrent une initialisation de tout les paramètres pour converger à une solution pertinente.

En fait, la connaissance de la position des points 3D et la connaissance des paramètres intrinsèques associés aux images permettent d'estimer la pose des prises de vue.

A partir de cette remarque, nous avons suggéré un nouvel algorithme, nommé CACHE_POS, (déjà publié dans [3]) qui dissimule les paramètres de pose des caméras et ne requiert donc aucune initialisation de ces paramètres. Cet algorithme minimise le critère C (i : indice des caméras, j : indice des points 3D, int : paramètres intrinsèque, f_2 : fonction d'estimation de pose des caméras, E : matrice extrinsèque estimé avec f_2 , f_1 : fonction exprimant la matrice extrinsèque en fonction des paramètres intrinsèques, I : matrice intrinsèque estimée avec f_1 , foc : longueur focale, (x, y, z) coordonnées 3D du sommet P^{3D} , δ_{ij} : 0 lorsque le point i n'est pas visible dans l'image j , 1 autrement):

$$C = \min_{\substack{int_j, \\ (x, y, z)_i}} \sum_{i,j} \delta_{ij} \left\| \tilde{p}_{i,j}^{2D} - I^j E^j(\tilde{P}_i^{3D}) \right\|$$

$$E^j = f_1(int_j, (x, y, z))$$

$$I^j = f_2(int_j)$$

Cette approche n'évalue que $kN+P$ paramètres, offre un large domaine de convergence et est plus rapide qu'une méthode classique lorsque le nombre de paramètres associés aux capteurs est important. En pratique, nous utilisons une minimisation non-linéaire de type Levenberg-Marquardt (MinPack) et l'algorithme de Davis et Dementhon [4] (fonction f_2) pour estimer les poses. Plusieurs expériences ont été menées pour évaluer les performances de cette approche. Afin de comparer la méthode classique rudimentaire n'exploitant pas l'aspect creux de la matrice quasi-hessienne et cette méthode, nous avons utilisé des données synthétiques. La scène 3D

parfaite est un nuage de vingt points 3D libres de toutes contraintes. La position de chaque point parfait est défini à l'intérieur d'une sphère de rayon unité par un tirage aléatoire suivant une loi uniforme. A partir de ce nuage parfait, un nuage bruité est généré par l'ajout d'un bruit gaussien sur les positions des points parfaits. Cinq caméras observent cette scène. Seules les longueurs focales des caméras sont considérées comme inconnues. Le point principal est supposé au centre de l'image et les pixels sont carrés.

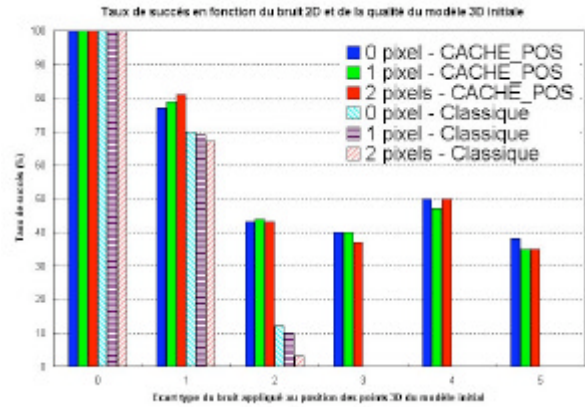


Figure 3. Taux de succès en fonction de l'erreur sur l'initialisation 3D (écart-type sur la distance au point 3D parfait).

La figure 3 présente le taux de succès en fonction du bruit introduit par rapport au modèle parfait et du bruit 2D introduit au niveau de la localisation des projections des points dans les images. Pour la méthode classique, l'algorithme de Davis et Dementhon a été utilisé pour initialiser les poses des images. Lorsque l'erreur initiale sur le modèle 3D est faible les deux méthodes convergent, par contre, lorsque l'erreur 3D initiale s'accroît, la méthode classique a un taux de convergence nul alors que la méthode proposée converge à la solution dans 40% des cas. L'échec de la méthode classique lorsque le modèle initial est très erroné est dû à une initialisation complètement fautive des caméras. Au contraire, la méthode CACHE_POS supporte cette erreur car à chaque itération, l'attitude de chaque caméra est remise en cause.

Le taux de succès de CACHE_POS dans le cas d'initialisations très éloignées de la solution est d'environ 40 %. Afin de comprendre la nature des échecs, nous avons tracé, pour une situation parfaite donnée (20 points, images), la valeur du critère (RMS^1) en fonction de la longueur focale. La valeur correcte menant à un RMS nul est 2000 pixels. La figure 4 présente le résultat obtenu. Pour de faibles valeurs de la longueur focale, le critère n'est absolument plus dérivable. Ceci s'explique, en partie, par le non respect pour des focales courtes de l'hypothèse para-perspective présente dans l'algorithme de Davis et Dementhon. D'un autre côté, pour les longueurs focales importantes, la courbe présente une

¹ RMS: Root Mean Square, distance moyenne entre les projections des sommets 3D et leurs positions localisées dans les images.

asymptote (non visible sur la figure 4) conduisant à des dérivées très faibles et bloquant de ce fait la convergence. Pour résumé, trois cas principaux d'échec ont été identifiés (deux ont été commentés ici) :

- l'arrivée en cours de minimisation à des valeurs de longueurs focales trop faibles (zone non dérivable),
- l'arrivée en cours de minimisation à des valeurs de longueurs focales trop fortes (dérivées quasi-nulles),
- des situations 3D dégénérées (surtout avec les modèles contraints).

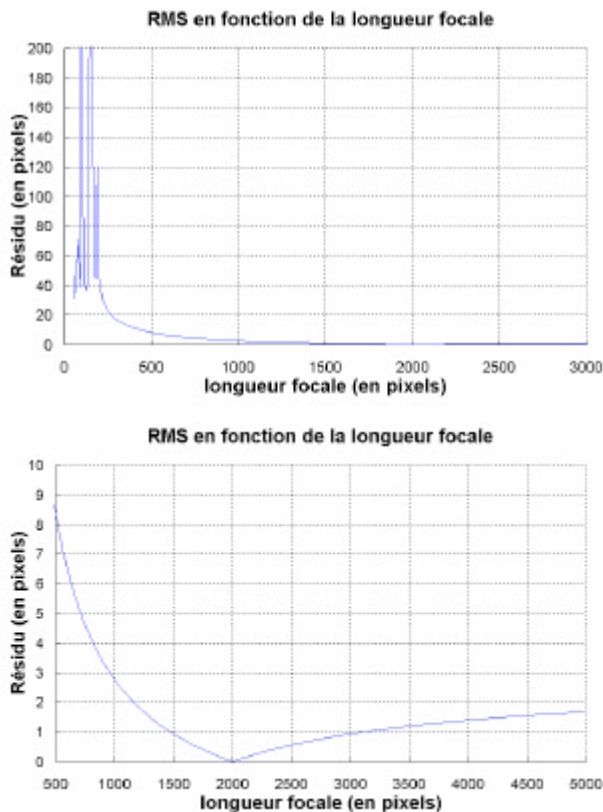


Figure 4. Pour une situation donnée, valeur du RMS en fonction de la longueur focale (valeur réelle 2000 pixels).

Pour conclure, la méthode CACHE_POS présente un taux de succès non nul même lorsque aucune initialisation de la scène n'est disponible. Ceci permet de l'utiliser même lorsque aucune initialisation pertinente n'est disponible (en générant différents états initiaux jusqu'à convergence).

3.2 Estimation de modèle non contraint

Afin d'expérimenter l'algorithme CACHE_POS sur des données réelles, nous avons pris six images d'un bureau. Ces images ont été obtenues avec un appareil photographique équipé d'un objectif fixe. Par conséquent, les paramètres intrinsèques sont identiques pour toutes les images. Ces images sont présentées Figure 5. Elles ont une taille de 1204x801 pixels. Lors de l'estimation, seule la longueur focale associée à la caméra a été estimée, nous avons supposé que le point principal est positionné au centre des images, il n'a pas été estimé.

Les images sont convergentes et toutes prises du même côté du bureau. Vingt deux points ont été définis et localisés manuellement dans les images. Afin de comparer le modèle obtenu suite à la reconstruction avec la réalité, nous avons mesuré au moyen d'un réglet des longueurs spécifiques entre des points 3D facilement identifiables et nous avons positionné une référence métrique dans la scène. Les distances mesurées entre certains points 3D ainsi que la référence métrique (1 meter) sont indiquées dans la Figure 6. Dans chaque image, il y a en moyenne soixante trois pour cent des points qui sont observés. Les paramètres des longueurs focales sont initialisés à 2000 pixels.



Figure 5. Les six images de la séquence.



Figure 6. Les longueurs mesurées avec leurs indices et la distance de référence (1 meter).

Numéro	Longueur mesurée (cm) ($\pm 1\text{mm}$)	Longueur Estimée (cm)	Erreur relative (%)
1	20.4	20.43	0.14
2	23.5	23.34	0.68
3	23.5	23.40	0.42
4	20.4	20.67	1.32
5	14.8	14.74	4.05
6	14.4	14.42	1.38
7	20.6	20.59	0.48
8	23.7	23.57	0.54
9	20.6	20.47	0.63
10	23.6	23.55	0.21
11	120.0	119.79	0.18
moyenne			0.64

Tableau 2: Comparaison entre les longueurs mesurées et les longueurs estimées.

Avant la reconstruction, nous ne disposons pas de la position des points 3D. L'initialisation a donc été effectuée en générant aléatoirement (loi uniforme) leurs positions initiales à l'intérieur d'une sphère de rayon unité. Le taux de succès (dépendant de l'initialisation des points 3D) est de quarante quatre pour cent et le temps d'une reconstruction est de 1,5 secondes.

En fin d'estimation, le RMS vaut 0.81 pixels et les paramètres caractérisant la longueur focale valent $f_x=1844,5$ pixels et $f_y=1847,2$ pixels. Afin de quantifier la qualité de la reconstruction, nous avons évalué les distances de référence en utilisant la métrique disponible. Les résultats sont présentés dans le tableau 2.

3.3 Estimation de modèles contraints

Dans la partie précédente, nous avons présenté l'approche d'ajustement de faisceaux CACHE_POS dans le cas de la reconstruction d'un nuage de points 3D libres. Dans le cas particulier des modèles contraints (section 2), l'ajustement de faisceaux est plus compliqué à réaliser. La mesure est toujours la distance entre les sommets 3D projetés dans les images et leurs positions 2D. Les paramètres estimés non-linéairement sont toujours les degrés de liberté existant dans le modèle du bâtiment et les paramètres associés aux caméras. Toutefois, il n'existe plus une relation directe entre les paramètres inconnus et les mesures. La méthode d'optimisation que nous avons proposée reste utilisable à partir du moment où la fonction, liant les projections des sommets 3D dans les images aux paramètres caractérisant le modèle, est continue et dérivable.

D'autre part, notre méthode de modélisation respecte strictement les contraintes définies dans le modèle et réduit ainsi le nombre de paramètres à estimer. L'introduction de ces contraintes change profondément la topologie sous-jacente du critère d'optimisation (définie à la section 3.1) et peut éventuellement générer des problèmes de convergence.

D'un point de vue expérimental, la méthode converge vers une solution présentant une bonne précision (jugée visuellement sur la projection du modèle dans les images et sur la valeur du RMS final) lorsque le modèle est bien équilibré. Au contraire, lorsque que l'on essaie de reconstruire un modèle défini sur plusieurs échelles (ex : le corps d'un bâtiment et une fenêtre) des difficultés apparaissent lors de l'estimation des paramètres car certaines parties pèsent aussi lourdement au niveau des mesures (et donc sur le critère) que d'autres alors qu'elles ne représentent qu'une petite partie du modèle réel. Par exemple, l'utilisateur peut être amené à sélectionner 4 points dans une image pour positionner une façade et également 4 points dans cette même image pour positionner une fenêtre, dans ce cas, la fenêtre « pèsera » autant que la façade et l'estimation des paramètres risquent d'être biaisée par ce déséquilibre. La solution que nous proposons est de reconstruire chaque élément de l'objet indépendamment en utilisant des images adéquates et en fusionnant dans un second temps l'ensemble des ces éléments afin de constituer le modèle final.

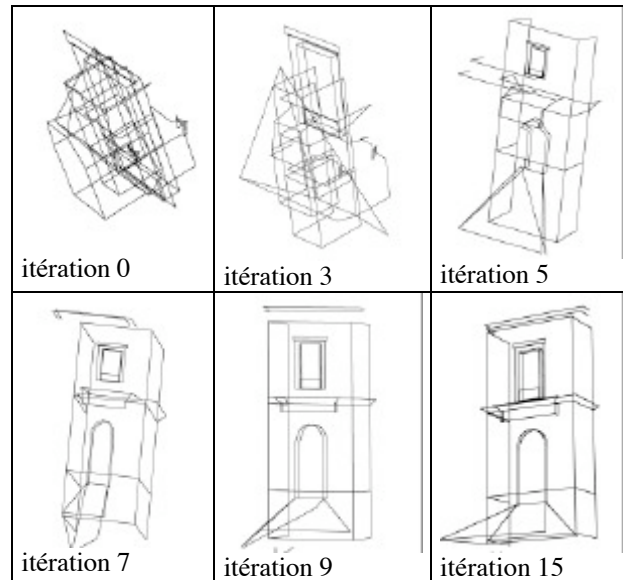


Figure 5. Quelques images montrant l'évolution du modèle au cours des itérations du processus d'optimisation. Le modèle initial (itération 0) a été obtenu en perturbant énormément les longueurs du modèle final.

Il est difficile de définir des conditions de tests équivalentes à celles utilisées pour évaluer les performances de notre critère dans le cas des nuages de points libres. Aussi avons-nous tout simplement pris un modèle 3D relativement complexe (obtenu suite à la reconstruction réalisée en section 4) que nous avons très fortement déformé. Le modèle déformé a servi de point de départ à l'estimation et la Figure 5 montre les différentes étapes de la convergence. La compréhension des conditions de convergence est difficile dans le cas des points libres et, a fortiori, encore plus difficile à percevoir dans le cas des modèles contraints. Diverses expériences nous ont néanmoins permis de mettre en évidence l'importance de la position relative des points les uns par rapport aux autres vis à vis du taux de succès.

Dans la pratique, l'utilisateur alterne entre la phase de modélisation, où il définit un ou plusieurs nouveaux points 3D contraints, et la phase d'optimisation permettant de dimensionner le modèle. Pour l'utilisateur, cette alternance modélisation/reconstruction facilite la perception du modèle 3D et donc la saisie des contraintes par le biais de l'interface. Du point de vue de l'optimisation, cette démarche incrémentale permet rapidement d'avoir un grand nombre de points bien positionnés les uns par rapport aux autres et fournit ainsi un modèle initial dans une configuration, a priori, favorable pour obtenir une convergence réussie.

4. Reconstruction d'un bâtiment à partir d'une séquence réelle

Nous avons choisi l'exemple du château de Sceaux, château français du XVII^{ème} siècle. Plusieurs éléments spécifiques ont été sélectionnés pour la reconstruction. 22 photographies ont été prises avec un appareil

photographique numérique. Deux objectifs de longueurs focales 28 mm et 135 mm ont été utilisés. La résolution des images est de 2000x3008 pixels, les distorsions n'ont pas été corrigées, les focales et les positions de prises de vue sont considérées comme inconnues et les images ont été obtenues à partir du sol.

Si nous observons ce château, il apparaît un grand nombre de détails (fenêtres, gouttières, bas-reliefs...). L'utilisateur est amené à choisir les éléments qu'il souhaite reconstruire (en fonction de ses besoins) car il serait trop coûteux de reconstruire avec précision l'ensemble des détails. Toutefois, dans notre approche, il sera toujours possible de compléter le modèle en apportant de nouveaux éléments (section 2.3) et/ou de nouvelles images.

4.1 Processus de reconstruction et résultats

Dans le cadre de cette modélisation, nous avons reconstruit le corps du bâtiment, les fenêtres du rez-de-chaussée et du premier étage, certains chien-assis et l'avancée au centre de la façade (Figure 6).

Chaque élément a été reconstruit indépendamment des autres en utilisant des images dédiées. La figure 7 présente l'exemple d'une fenêtre du rez-de-chaussée. Seulement trois images étaient disponibles, nous avons défini un modèle contraint en utilisant la description par graphe (section 2) et nous avons utilisé l'ajustement de faisceaux décrit en section 3 afin de définir les paramètres du modèle 3D et des images.

Après une définition manuelle des surfaces (voir [14] pour un algorithme automatique), la texture a été extraite des images (section 4.2). La même procédure a été appliquée à tous les éléments.



Figure 6. Images de la séquence montrant chacun des éléments du château reconstruit indépendamment.

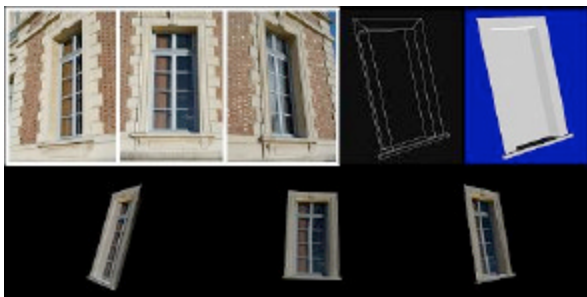


Figure 7. Exemple de la fenêtre du rez-de-chaussée. Les 3 images utilisées pour la reconstruction sont en haut à gauche. En haut à droite se trouve une image du modèle filaire obtenu et une image du modèle surfacique. Enfin, en dessous, il y a 3 images

du modèle texturé.

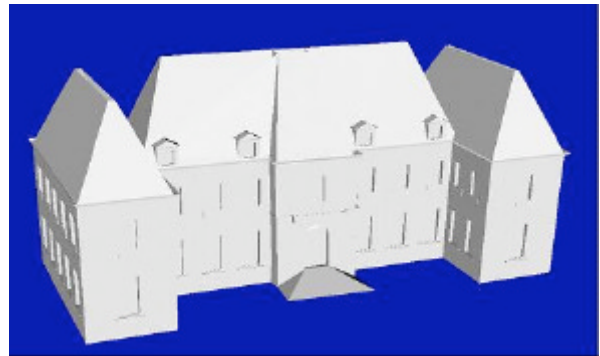


Figure 8. Modèle global sans texture. Tous les éléments spécifiques ont été fusionnés pour obtenir un modèle global multi-échelles.

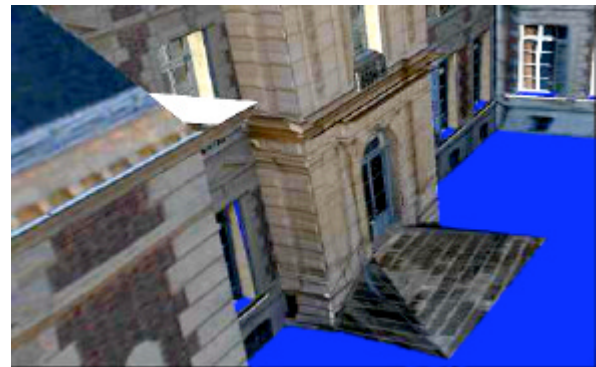


Figure 9. Modèle final avec la texture.

Par la suite, tous les éléments ont été fusionnés afin d'obtenir un modèle global du château de Sceaux. Par exemple, afin de localiser les fenêtres du rez-de-chaussée trois coins de chaque fenêtre ont été localisés dans la façade en utilisant des contraintes. Ces fenêtres sont positionnées à une précision relative à la précision de reconstruction du bâtiment mais leurs proportions et leurs textures ont été définies précédemment en utilisant des images de gros plans. Pour les éléments répétitifs, un seul modèle a été défini. Ce processus de fusion a été appliqué à chaque détail et le résultat est montré dans les figures 8 et 9.

Deux jours de travail ont été nécessaires pour obtenir ces résultats et le RMS final est de 8 pixels. Ce résidu 2D relativement élevé s'explique par la difficulté à définir précisément la position des extrémités du bâtiment dans les images et par la présence de distorsions (ces distorsions n'ont pas été corrigées car aucun calibrage préalable n'a été effectué!). Quoiqu'il en soit, cette approche stratifiée nous a permis de reconstruire un modèle complet du château tout en conservant la précision et la texture propre à chacun des éléments. Il est toujours possible de compléter le modèle et d'affiner sa précision en fonction du besoin et du temps disponible.

4.2 Extraction de texture

L'extraction des textures à partir des images permettant d'obtenir un modèle 3D réaliste a été effectuée au moyen d'un algorithme simple. Comme nous connaissons la surface 3D du modèle et la position de chaque image, nous avons extrait la couleur de chaque zone en tenant compte des occultations. Ce processus de lancer de rayons est très coûteux en temps de calcul. La texture obtenue a une résolution adaptée à chaque élément. L'utilisateur dispose donc d'une texture très précise sur certains éléments (les clous dans les boiseries des fenêtres sont visibles) tout en manipulant des textures beaucoup plus légère pour les zones moins intéressantes.

5. Appariement de points homologues

A la fin de ce processus de reconstruction, l'utilisateur dispose d'un modèle contraint dimensionné et de caméras calibrées et localisées. La qualité de cette reconstruction et de cet auto-calibrage est directement conditionné par la qualité des contraintes définies par l'utilisateur et la qualité de la localisation des sommets dans les images. Cette première reconstruction fournit des résultats visuels satisfaisant mais peut être améliorée en utilisant plus l'information existant dans les images. Des outils permettant d'accroître la qualité de la reconstruction sont en cours d'élaboration.

La première étape commune à l'ensemble de ces outils est la définition d'un ensemble d'appariement de points homologues (2D/2D) entre les différentes images de la séquence. L'exploitation des informations disponibles simplifie très largement le processus d'appariement est très largement simplifié par rapport aux cas classiques. La première opération consiste à extraire de l'image un ensemble de points d'intérêt en conservant uniquement ceux pouvant potentiellement appartenir au bâtiment (intersection de la droite « centre optique-point d'intérêt » et modèle 3D du bâtiment non vide). Pour un point d'intérêt valide, il est facile de définir sa projection dans les autres images de la séquence en tenant compte des occultations éventuelles et de définir ainsi la position approximative des points homologues. De plus, la surface 3D sous-jacente à ce point d'intérêt est connue ce qui permet de définir l'homographie approximative permettant de transformer la texture au voisinage du point d'intérêt en la texture au voisinage des points homologues supposées. A partir de ces connaissances, nous pouvons déterminer la position des points homologues par un critère de corrélation.

Nous avons testé cette méthode sur la séquence du château de Sceaux. Les conditions de l'appariement permettent de fixer un seuil limite de corrélation très élevé et de ne conserver ainsi que les appariements très fortement probable. Dans le cas présenté dans la Figure 10, 20000 points 3D ont été reconstruits à partir des appariements obtenus par cette technique.

Ces appariements peuvent être utilisés afin d'affiner le calibrage associé aux différentes prises de vue. Nous allons également étudier les solutions possibles afin

d'accroître la précision de l'opération de fusion des éléments (qui repose actuellement sur la qualité des points saisis par l'utilisateur). Enfin, nous allons étudier les voies d'amélioration de la qualité du modèle du bâtiment

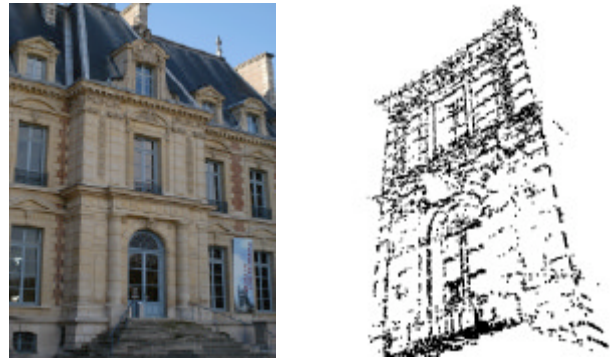


Figure 10. A gauche se trouve une des images d'un élément de la façade du bâtiment, à droite une image du nuage de points reconstruit suite à l'appariement 2D/2D.

6. Conclusion

Nous avons présenté une méthode originale de reconstruction 3D à partir d'images non-calibrées. Le modèle utilisé est constitué de points 3D contraints au moyen de règles géométriques simples et intuitives. Afin d'estimer les dimensions de ce modèle à partir des informations saisies dans les images, un algorithme d'ajustement de faisceaux, ne requérant pas d'initialisation des poses et capable d'optimiser des modèles contraints, a été développé et évalué. Enfin, la méthode a été employée avec succès sur un exemple réel et un modèle multi-échelle du château de Sceaux a été obtenu. Le résultat final est un outil simple d'emploi qui offre la flexibilité des approches par primitives bas-niveaux et la modularité des approches par primitives complexes. Les prochains travaux vont s'appuyer sur l'appariement de points homologues présenté à la fin de cet article afin d'améliorer la qualité globale de la reconstruction.

Références

- [1] A. Bartoli and P. Sturm. *Constrained structure and motion from multiple uncalibrated views of a piecewise planar scene*, International Journal of Computer Vision 52(1), 45-64, 2003.
- [2] R. Cipolla and D.P. Robertson. *3D Models of architectural scenes from uncalibrated images and vanishing points*. In Proc. IAPR 10th International Conference on Image Analysis and Processing, pages 824-829, Venice, 1999.
- [3] S. Cornou, M. Dhome and P. Sayd. *Bundle adjustment: a fast method with weak initialization*. BMVC 2002, Cardiff.

- [4] L.S. Davis and D.F. Dementhon. *Model-based object pose in 25 lines of code*. International Journal of Computer Vision, 15(2):123–141, 1995.
- [5] P. E. Debevec, C. J. Taylor, and J. Malik. *Modeling and Rendering Architecture from Photographs: A Hybrid Geometry and Image-Based Approach*. In SIGGRAPH 96, August 1996, New Orleans.
- [6] O. Faugeras, Q.T. Luong and T. Papadopoulos. *The geometry of Multiple Images: the laws that govern the formation of multiple images of a scene and some of their applications*. ISBN 0-262-06220-8, The MIT Press, 2001.
- [7] E. Grossmann and J. Santos-Victor. *Maximum likelihood 3D reconstruction from one or more images under geometric constraints*. BMVC 2002, Cardiff.
- [8] R. Hartley and A. Zisserman. *Multiple View Geometry*. Cambridge university press, 2000.
- [9] F. Lang, and W. Förstner. *3D-city modeling with a digital one-eye stereo system*, In Proceedings of the XVIII ISPRS-Congress, 1996, Vienna, Austria.
- [10] M. Brown and D. G. Lowe, *Invariant features from interest point groups*, British Machine Vision Conference, BMVC 2002, pp. 656-665, Cardiff.
- [11] E. Malis, A. Bartoli. *Euclidean Bundle Adjustment Independent on Camera Intrinsic Parameters*. Research report n°4377. INRIA – ISSN 0249-6399 - December 2001.
- [12] J. Matas, O. Chum, M. Urban and T. Pajdla, *Robust Wide Baseline Stereo from Maximally Stable Extremal Regions*, British Machine Vision Conference, BMVC 2002, Cardiff.
- [13] K. Mikolajczyk and C. Schmid. *An affine invariant interest point detector*. In European Conference on Computer Vision, vol. 1, 128–142, 2002.
- [14] D.D. Morris and T. Kanade. *Image-consistent surface triangulation*. Computer Vision and Pattern Recognition (CVPR 2000), IEEE Computer Society, Vol. 1, June, 2000, pp. 332-338.
- [15] M. Pollefeys. *Tutorial on 3D modelling from images*, in conjunction with ECCV 2000, 26 June 2000, Ireland, Dublin.
- [16] C. Rother, *Multi-View Reconstruction and Camera Recovery using a Real or Virtual Reference Plane*, Doctoral Dissertation, ISBN 91-7283-422-6, KTH, Stockholm, Sweden, Januari, 2003.
- [17] F. Schaffalitzky and A. Zisserman. *Multi-view matching for unordered image sets, or “How do I organise my holiday snaps?”*, 7th ECCV, 2002, Copenhagen.
- [18] Steedly, Essa. *Propagation of Innovative Information in Non-Linear Least-Squares Structure from Motion*, In Proceedings of International Conference on Computer Vision (ICCV 2001), Vancouver BC, Canada, July 2001.
- [19] B. Triggs, P. McLauchlan, R. Hartley and A. Fitzgibbon. *Bundle adjustment – A modern synthesis*. In W. Triggs, A. Zisserman and R. Szeliski, editors, Vision algorithms: Theory and practice, LNCS, pages 298-375. Springer Verlag, 2000.
- [20] T. Werner and A. Zisserman. *New Techniques for Automated Architecture Reconstruction from Photographs*. Proc. 7th European Conference on Computer Vision, Copenhagen, Denmark, 2002.
- [21] M. Wilczkowiak, P. Sturm, E. Boyer. *The Analysis of Ambiguous Solutions in Linear Systems and its Application to Computer Vision*. In Proceedings of the 14th BMVC, Norwich, UK, Vol I, pp. 53–62, September 2003.
- [22] Z. Zhang and Y. Shan. *Incremental Motion Estimation Through Local Bundle Adjustment*. Technical Report MSR-TR-01-54, Microsoft Research, Microsoft Corporation One Microsoft Way, Redmond, WA 98052, May 2001.