

Modélisation 3D d'un objet par un capteur stéréo monté sur un manipulateur mobile

F. Trujillo-Romero

M. Devy

LAAS groupe Robotique et Intelligence Artificiel

LAAS, Av du Colonel Roche, 31077 Toulouse Cedex 04

{ftrujill, michel}@laas.fr

Résumé

Ce rapport montre des résultats obtenus sur la modélisation incrémentale d'un objet 3D de forme quelconque, en utilisant un capteur stéréo monté sur un bras manipulateur, lui-même monté sur un robot mobile. Une telle processus nécessite les étapes classiques de recalage entre vues acquises pendant le déplacement du capteur autour de l'objet à modéliser, puis de construction d'un maillage depuis le nuage de points 3D acquis depuis toutes les vues. Nous présentons surtout une technique alternative aux algorithmes de type Marching Cubes exploités pour construire le maillage. Cette technique exploite une paramétrisation sphérique des points 3D afin de construire le maillage d'une façon rapide et efficace, puisque c'est plus facile d'obtenir le maillage d'une sphère que d'un objet dont nous ne connaissons pas la forme.

Mots clefs

Modèle 3D, Maillage, Stéréovision, Recalage, Fusion.

1 Introduction

La modélisation d'objets 3D peut être traitée par de nombreuses méthodes, exploitant divers capteurs pour acquérir des points 3D sur la surface de l'objet. Les progrès technologiques très rapides en ce domaine, donnent accès à des capteurs 3D qui ont chaque fois une sensibilité et une résolution plus grandes que celles qui peuvent être obtenues avec une simple paire de caméras, utilisée dans un banc stéréo. Ces nouveaux capteurs se fondent sur la télémétrie-laser, sur un couple caméra-illuminateur ou sur la mesure du temps de vol (Swiss Ranger) ; ils permettent de modéliser des objets qui ont des surfaces non texturées. Mais ces techniques ont aussi quelques inconvénients ; ils peuvent être encombrants, chers et lents (télémétrie avec balayage deux axes), peuvent être trop fragiles pour être montés sur un manipulateur (Swiss Ranger), ou encore nécessiter un calibrage délicat (caméra avec illuminateur). Par ailleurs une meilleure résolution implique une grande quantité de points, ce qui rend plus complexes et lentes, les méthodes de construction de maillage. Or, nous participons au projet européen *Cogniron*, qui étudie le concept du robot as-

sistant de l'homme. Il a pour but entre autres, d'intégrer sur un robot, des fonctions permettant de modéliser un objet inconnu a priori, puis de le saisir pour le donner à un homme ; dans ce contexte, les méthodes de reconstruction 3D doivent être exécutées en ligne.

Il existe plusieurs travaux dans le domaine de la modélisation 3D. Nous nous sommes inspirés des travaux de Floater [1] qui paramétrise une maille 3D vers une maille 2D sans perte d'information. Bien que la paramétrisation de Floater, comme toutes paramétrisations, exploite une maille déjà construite, cette méthode de reconstruction permet de réduire ou augmenter la quantité de polygones. Gu et al. [2] ont décrit comment comprimer un maillage triangulaire en utilisant la méthode des images géométriques. La méthode que nous utilisons pour modéliser un objet, se base en partie sur la paramétrisation sphérique et sur les travaux de Gu et al. [3], qui ont comparé différentes méthodes pour effectuer une telle paramétrisation. Hormman et Greiner [4] proposent une alternative à la paramétrisation, en exploitant un plan à la place d'une sphère ; cette représentation présente certaines caractéristiques très intéressantes pour traiter de la reconstruction d'objets 3D en temps réel.

La section 2 présente plusieurs méthodes proposées pour la construction de maillage 3D. La section 3 décrit nos propres travaux sur la modélisation, tandis que la section 4 présente quelques résultats. Enfin la section 5 résume l'état actuel de nos travaux et donne quelques perspectives.

2 Problématique : la construction d'un maillage 3D

Une fois qu'un nuage de points est obtenu par recalage de plusieurs vues, il est très difficile de construire un maillage, et cela d'autant plus qu'il existe des contraintes temps réel. Plusieurs problèmes sont posés : (1) les relations topologiques entre points (voisinage) sont perdues après recalage et aggrégation des images de points 3D dans un nuage, (2) les points présentent un bruit non uniforme, (3) la résolution des points est variable dans le nuage et (4) l'objet à modéliser peut avoir une géométrie complexe (concavités, trous...) et peut comporter plusieurs parties non connexes.

Le maillage doit avoir une résolution variable, afin de lisser le bruit et de disposer de modèles multi-résolution.

Citons d'abord les techniques de décimation, exploitées pour réduire le nombre de triangles dans un maillage existant (Garland, Hoppe, Rossignac...). La décimation décrite par Schroeder et al. [5] permet de réduire le nombre de polygones, y compris sur des maillage de très grande taille. Malheureusement ces méthodes ne peuvent pas s'appliquer à un nuage de points 3D, non encore représenté par un maillage. Par conséquent nous devons employer une technique alternative comme celle mentionnée dans le travail développé par Gao et Lu [6].

D'autre part il existe plusieurs méthodes pour construire un maillage depuis un nuage de points. La plus célèbre est la méthode des *Marching Cubes* [7], développé par Lorensen et Cline. C'est un algorithme rapide mais qui a l'inconvénient de laisser les trous dans un maillage là où la résolution du nuage est trop faible. Il existe des améliorations et des variantes de cet algorithme. Citons les *Marching triangles* [8] et Les *Dual Marching Cubes* [9].

A. Restrepo [10] a montré que l'algorithme de *Ball Pivoting* développé par Bernardini et al. [11] est meilleur que *Marching Cubes* mais qu'il a l'inconvénient d'être très lent. Un autre algorithme très classique pour modéliser des objets 3D de forme quelconque ou déformable (en particulier, applications médicales : coeur... Voir un exemple dans [12]) exploite le concept de *Surface déformable*. Cet algorithme part d'un maillage initial, généralement généré par discrétisation d'une sphère, qui doit être déformé itérativement afin de suivre au mieux les points de mesure. La modélisation obtenue est souvent meilleure qu'avec d'autres méthodes, mais avec l'inconvénient que le réglage des paramètres est très délicat et très dépendant de la forme de l'objet à modéliser.

Nous pouvons finalement citer les techniques qui utilisent la triangulation Delaunay comme dans les travaux de Fang et Piegl [13]. Ils présentent un algorithme qui est rapide et qui fournit de bons résultats. C'est dans cette dernière catégorie que nous pouvons inclure notre travail, vu que c'est la plus simple à mettre en oeuvre et c'est aussi la plus rapide pour une exécution en temps réel.

3 Nos développements sur la modélisation 3D

3.1 Système utilisé

La figure 1 montre le robot utilisé pour valider notre travail. Ce robot est formé par une plateforme mobile Neobotix sur laquelle est monté un bras robotique PA10-6C développé par Mitsubishi. Ce bras a six degrés de liberté. L'organe terminal est doté d'un capteur d'effort et d'une pince (constituée de trois doigts) qui servira à prendre des objets à partir d'informations visuelles. Ces informations sont acquises depuis une paire stéréo exploitée dans notre travail, pour modéliser l'objet (supposé inconnu) avant de chercher une position de prise pour la pince, et de contrôler

la saisie. Le robot JIDO dispose d'autres capteurs montés sur un mât derrière le bras : nous n'utilisons pas ces capteurs pour l'heure.



Figure 1 – Bras Robotique PA10-6C : JIDO

Le processus de modélisation se divise en trois fonctions principales : l'acquisition des images, le recalage et la construction du maillage. Dans les sections suivantes nous développerons chacune de ces étapes, qui s'exécutent soit itérativement (recalage et mise-à-jour incrémentale du maillage après chaque acquisition) ou en séquence (acquisition de N images, recalage de ces images et fusion dans un nuage de points 3D, construction du maillage à partir de ce nuage).

3.2 Acquisition des images sur l'objet

La phase d'acquisition des images est démarrée quand le système détecte dans l'environnement, un objet à modéliser. Dans le scénario complet, cette détection nécessite d'abord que le robot localise une table, s'en approche, acquérir une image du plateau et y détecte la présence d'un objet. A partir des points 3D calculés depuis cette image, le centre de gravité P des points perçus sur l'objet est grossièrement localisé sur la table. Ce point P permet de calculer la position d'accostage de la plateforme autour de la table, et les positions du bras pour placer le capteur stéréo et acquérir des images complémentaires sur l'objet.

A partir d'une demi-sphère centrée sur le point P, nous calculons 6 positions différentes pour pouvoir modéliser l'objet. Nous pourrions prendre beaucoup plus d'images mais il est suffisant d'acquérir les vues principales de l'objet, et surtout, de cette manière, le système est plus rapide. On calcule ces positions du capteur stéréo sur la demi-sphère de sorte que les positions correspondantes de l'organe terminal du bras, soient atteignables : pour cette raison, on ne peut acquérir des images sur la partie arrière de l'objet. Dans une extension future, le robot JIDO pourra accoster la table en plusieurs positions afin de voir l'objet sous tous ses points de vue.

3.3 Recalage des images 3D dans un nuage de points 3D

Après chaque acquisition i , le module stéréo est activé pour générer une image I_i de points 3D, reconstruits dans le repère du capteur stéréo. Pour agréger cette image I_i dans le nuage de points 3D, exprimée dans le repère du robot

JIDO, nous devons calculer la transformation T_i entre le repère stéréo (position du point de vue courant sur l'objet) et le repère JIDO. Une estimée initiale T_i^0 de cette transformation T_i est obtenue en composant plusieurs transformations rigides (position du repère stéréo par rapport au repère Organe terminal, position de la base du bras sur JIDO) avec la transformation qui donne la position de l'organe terminal par rapport à la base du bras, estimée par les codeurs du bras.

Cette estimée T_i^0 n'est pas suffisamment précise pour garantir une bonne modélisation. Il faut appliquer la méthode *Iterative Closest Points* (ICP) [14] afin d'améliorer itérativement cette transformation, pour à la fin, obtenir un nuage de points cohérent. Ce processus ICP n'est pas trivial, car il faut choisir une stratégie (recalage incrémental de I_i avec I_{i-1} ou recalage de I_i avec le nuage en cours de formation), il faut conserver uniquement les bons points à apparier dans l'image I_i et les précédentes. Nous utilisons la méthode ICP implementée par A. Restrepo [10]. La convergence n'est pas garantie, mais nous avons vérifié que la précision finale est suffisante.

3.4 Construction du maillage

A partir du nuage de points obtenu après recalage, le maillage est construit en utilisant le processus de paramétrisation sphérique, qui consiste à projeter tous les points du nuage vers une sphère (en fait une demi-sphère vu que l'objet est posé sur une table). Il n'existait pas de méthode de ce type, car la plupart des travaux existants (voir les sections 1 et 2) se basent sur l'existence d'un maillage initial de triangles ou de polygones, qu'il convient ensuite de décimer.

Notre mise en oeuvre part de l'équation d'une sphère que possède un centre \vec{C} et un rayon R . Le centre de la sphère est dans le centre de gravité du nuage de points et le rayon est la distance qui existe du \vec{C} vers le point le plus éloigné de la nuage plus un petit accroissement ε . Avec ceci nous arrivons à enfermer complètement le nuage. Dans l'équation 1 on peut voir cette relation

$$R = \max(x, y, z) + \varepsilon \quad (1)$$

Une fois déterminé le centre et le rayon de la sphère on projette chaque un des points du nuage vers la sphère. Celui-ci, c'est le processus principal de notre travail dû au fait que s'il n'est pas bien calculé peuvent apparaître effets non souhaités. Comme dans projeter plusieurs points du nuage dans un seul point de la sphère. Il convient de clarifier que doit exister un seul point dans la sphère par un point dans le nuage 3D.

Ceci est obtenu de la manière suivante :

Soit une droite \vec{L} qui passe par le centre \vec{C} et par un point \vec{N} du nuage à triangulaire. On calcule la plus petite distance entre le point \vec{N} et la sphère.

Pour effectuer ce calcul il est nécessaire de trouver l'intersection de la droite \vec{L} et la sphère définie par \vec{C} et R .

Nous considérons d'abord la paramétrisation de la droite \vec{L} comme

$$\vec{L}(t) = \vec{N} + t\vec{D} \quad (2)$$

où

$$\vec{D} = \vec{N} - \vec{C} \quad (3)$$

et la paramétrisation de la sphère comme

$$|\vec{L} - \vec{C}|^2 = R^2 \quad (4)$$

En remplaçant l'équation 1 de la droite \vec{L} dans l'équation 4 nous allons obtenir la valeur de t . Lequel doit être positif pour l'intersection de la droite et la sphère.

Avec ceci nous calculons le point dans la sphère comme l'indique l'équation 1 et nous obtenons la projection de \vec{n} dans la sphère.

On effectue la triangulation en utilisant l'algorithme de convex hull. Grâce au fait que tous les points du nuage sont dans la sphère il est possible d'obtenir une surface uniforme.

Une fois triangulaire la sphère nous obtenons les relations des triangles formés pour pouvoir produire le modèle final de l'objet qui est reconstruit.

En bref, c'est l'équation 1 qui est exploitée pour projeter un point de l'objet sur la sphère, en faisant l'hypothèse que la droite reliant le centre de la sphère au point 3D, coupe la sphère en un seul point. Nous calculons cette intersection pour obtenir la paramétrisation sphérique souhaitée. Le maillage peut être construit facilement sur la sphère, et peut être décimé par une méthode classique (Hoppe ou Garland) pour obtenir un maillage avec une résolution adaptée aux traitements suivants (pour nous, recherche des positions de prise pour la pince) et à la complexité de la forme de l'objet.

4 Résultats

Cette section présente plusieurs résultats ainsi que la comparaison de notre méthode avec deux autres algorithmes de reconstruction 3D, *Ball Pivoting* [11] et *Marching Cubes* [7].

Les figures 2 à 5 montrent les résultats obtenus lorsque nous avons appliqué les différents algorithmes pour modéliser une tête de mannequin. Dans la figure 2 nous pouvons observer le résultat de la paramétrisation sphérique appliquée au nuage de points acquis sur le mannequin. Ensuite nous présentons la reconstruction du mannequin avec la paramétrisation sphérique en figure 3, avec *Marching Cubes* en figure 4 et *Ball pivoting* en figure 5.

Il existe beaucoup de différences entre les méthodes, sur la qualité du maillage obtenu et sur les temps d'exécution. Malgré la rapidité d'exécution de *Marching Cubes*, la qualité du modèle final n'est pas bon. En outre, notre algorithme a un temps d'exécution sensiblement équivalent à celui de *Marching cubes*, mais le résultat final est meilleur.

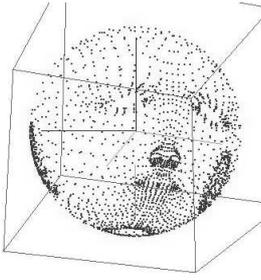


Figure 2 – Paramétrisation sphérique du mannequin

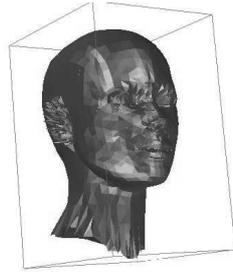


Figure 3 – Reconstruction par la paramétrisation sphérique

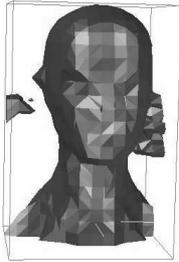


Figure 4 – Reconstruction par Marching cubes

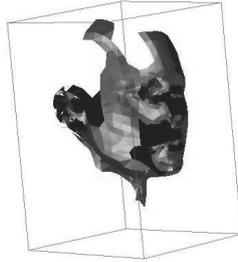


Figure 5 – Reconstruction par Ball-Pivoting

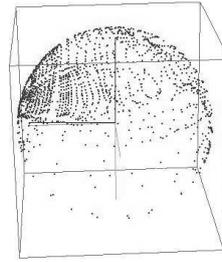


Figure 6 – Paramétrisation sphérique de la boîte

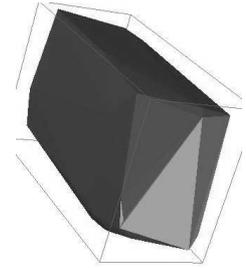


Figure 7 – Reconstruction par la paramétrisation sphérique

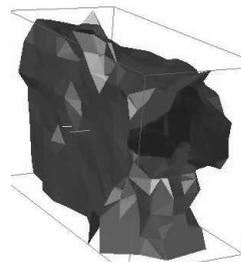


Figure 8 – Reconstruction par Marching cubes

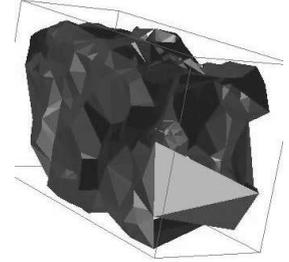


Figure 9 – Reconstruction par Ball-Pivoting

Les figures 6 à 9 présentent les résultats obtenus lorsque nous avons appliqué les différents algorithmes de reconstruction décrits dans la section précédente, à un objet réel, en ce cas une boîte parallélépipédique. La figure 6 montre le résultat de la paramétrisation sphérique du nuage de points. Nous présentons la reconstruction de cette boîte, obtenue avec l’algorithme que nous proposons en figure 7, avec *Marching cubes* en figure 8 et avec *Ball pivoting* en figure 9.

Objet	M.C.	B.P.	P.S.
Mannequin	3.82 s.	9.13 m.	5.87 s.
Boîte	2.51 s.	6.32 m.	3.74 s.

Tableau 1 – Temps de reconstruction

Le tableau 1 montre le temps d’exécution des algorithmes avec les deux modèles utilisés : nous pouvons voir que le compromis qualité/temps est meilleur dans notre algorithme qu’avec *Marching Cubes* ou *Ball Pivoting*. Ce dernier est très complexe, et nécessite des temps d’exécution incompatibles avec un traitement en temps réel.

5 Conclusions et perspectives

Nous avons présenté une méthode alternative pour reconstruire des objets 3D de forme libre à partir de nuages de points acquis par stéréovision. Cette méthode se fonde sur une paramétrisation sphérique. Cet algorithme est non seulement fonctionnel pour effectuer la triangulation d’un nuage de points 3D, mais il est en plus très rapide. La comparaison de cette méthode avec *Marching Cubes* et *Ball Pi-*

voting a montré que notre méthode présentait plus d’avantage que d’inconvénients.

Il convient de mentionner que les objets à modéliser doivent être convexes, ce qui est le cas de nombreux objets que nous utilisons tous les jours dans notre maison ou dans le travail. Néanmoins, une amélioration en cours est l’extension de cette méthode pour l’appliquer à des objets concaves. Cette mise en oeuvre sera d’une grande utilité pour pouvoir modéliser tout type d’objet.

A court terme, nous allons participer à une expérimentation collective, qui doit valider les capacités du robot JIDO pour : (1) détecter un objet inconnu dans son environnement, (2) le modéliser, (3) le saisir et (4) le tendre à un opérateur humain préalablement perçu.

A moyen terme, cette méthode sera exploitée pour créer une base de données d’objets 3D en exploitant pour cela la méthodologie d’images géométriques développée par Gu et al. [2], présentée aussi par Hoppe et al. [15] pour obtenir la représentation en 2D d’un objet 3D. Pour pouvoir effectuer la reconnaissance d’un objet par classification à partir de ses caractéristiques, il est nécessaire d’effectuer cette paramétrisation.

Remerciements

Cette recherche a été menée grâce à l’appui financier de l’organisme Mexicain CONACYT et a été réalisée dans le cadre du projet Européen COGNIRON .

Références

- [1] M. Floater. Parametrization and smooth approximation of surface triangulations. *CAGD*, 14(3) :231–

250, 1997.

- [2] X. GU, S. Gortler, et H. Hoppe. Geometry images. *ACM Transaction on Graphics*, 21(3) :355–361, 2002.
- [3] X. GU C. Gotsman et A. Sheffer. Fundamentals of spherical parametrization for 3d meshes. *ACM Transaction on Graphics*, 22(3) :358–363, July 2003.
- [4] K. Hormman et G. Greiner. Mips : An efficient global parametrization method. *Curve and surface Desing*, pages 153–162, 2000.
- [5] J. A. Zarge W. J. Schroeder et W. E. Lorensen. Decimation of triangle meshes. Dans *SIGGRAPH*, pages 65–70. ACM, 1992.
- [6] S. Gao et H.-Q. Lu. A fast algorithm for delaunay based surface reconstruction. *The 11-th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision*, February 2003.
- [7] W. E. Lorensen et H. E. Cline. Marching cubes : A high resolution 3d surface construction algorithm. Dans *SIGGRAPH '87 : Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 163–169, New York, NY, USA, July 1987. ACM Press.
- [8] S. Akkouche et E. Galin. Adaptive implicit surface polygonization using marching triangles. *Comput. Graph. Forum*, 20(2) :67–80, 2001.
- [9] G. M. Nielson. Dual marching cubes. Dans *VIS '04 : Proceedings of the conference on Visualization '04*, pages 489–496, Washington, DC, USA, 2004. IEEE Computer Society.
- [10] J.A. Restrepo Specht. *Modélisation d'objets 3D par construction incrémentale d'un maillage triangulaire, dans un contexte robotique*. Thèse de doctorat, LAAS, 7, Av Colonel Roche Toulouse France, Janvier 2005.
- [11] F. Bernardini, J. Mittleman, H. E. Rushmeier, C. T. Silva, et G. Taubin. The ball-pivoting algorithm for surface reconstruction. *IEEE Trans. Vis. Comput. Graph.*, 5(4) :349–359, 1999.
- [12] C. Xu et J. L. Prince. Gradient vector flow : A new external force for snakes. Dans *Conference on Computer Vision and Pattern Recognition (CVPR '97)*, pages 66–. IEEE Computer Society, 1997.
- [13] T. P. Fang et L. A. Piegl. Delaunay triangulation in three dimensions. *IEEE Computer Graphics and Applications*, 15(5) :62–69, Sept. 1995.
- [14] Y. Chen et G. G. Medioni. Object modelling by registration of multiple range images. *Image Vision Comput.*, 10(3) :145–155, 1992.
- [15] H. Hoppe. Overview of recent work on geometry images. Dans *Geometric Modeling and Processing*, page 12. IEEE Computer Society, 2004.