

# Reconstruction 3D d'objets sous-marins par asservissement visuel

V. BRANDOU<sup>1,2</sup>, E. MALIS<sup>1</sup>, P. RIVES<sup>1</sup>, A.G. ALLAIS<sup>2</sup>, M. PERRIER<sup>2</sup>

<sup>1</sup> INRIA Sophia-Antipolis

<sup>2</sup> IFREMER Centre de Méditerranée, La Seyne-sur-Mer

2004 route des lucioles  
BP 330 06902 Sophia-Antipolis  
Vincent.Brandou@sophia.inria.fr

## Résumé

*L'objectif est de proposer une méthodologie complète de reconstruction 3D d'objets sous-marins à des fins de mesures quantitatives. Afin de faciliter le calcul de la reconstruction 3D, nous utilisons un asservissement visuel à l'aide d'un système de stéréovision permettant d'obtenir différentes vues de l'objet à intervalles réguliers selon une trajectoire prédéfinie.*

## 1 Introduction

Nos recherches menées dans le cadre du projet européen EXOCET/D<sup>1</sup>, sont nées du besoin d'avoir des outils adaptés pour l'exploitation des images de scènes sous-marines, en appliquant les techniques de vision par ordinateur. Notre but est de développer une méthodologie pour effectuer une reconstruction 3D de structures sous-marines, permettant de réaliser des mesures quantitatives tridimensionnelles. Dans cette optique, nous avons mené des actions de recherche portant sur l'utilisation de techniques d'asservissement visuel afin d'améliorer la reconstruction d'objets 3D. La technique développée permet de réaliser des trajectoires optimales en vue de réaliser une reconstruction précise. Nous utilisons un asservissement visuel à l'aide d'un système de stéréovision monté au bout du bras instrumenté d'un robot. La géométrie de la tête stéréo contraint la trajectoire suivie par les caméras.

## 2 Trajectoires contraintes par la géométrie de la tête stéréo

L'asservissement visuel utilisé consiste à capturer une image de référence avec la première caméra, puis à commander la seconde caméra par rapport à cette image afin qu'elle converge vers la position d'où a été prise l'image de référence. L'angle et la distance entre les caméras contraignent la trajectoire décrite par la tête stéréo. Nous avons montré que fixer la géométrie entre les deux caméras, revient à déplacer l'origine du repère associé à

chacune des deux caméras sur la surface d'un cylindre. La démonstration consiste à intégrer des déplacements infinitésimaux pour obtenir un déplacement discret. Soit  $\mathbf{T} \in SE(3)$ <sup>2</sup> la matrice de transformation entre la position des deux caméras, où  $\mathbf{R}$  est la matrice de rotation ( $3 \times 3$ ), et  $\mathbf{t}$  le vecteur de translation ( $3 \times 1$ ).

$$\mathbf{T} = \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{pmatrix} \quad \mathbf{T} \in SE(3) \quad (1)$$

$\mathbf{T}$  peut toujours s'écrire :

$$\mathbf{T} = e^{\mathbf{A}} \quad \mathbf{A} \in se(3)^3 \quad (2)$$

avec

$$\mathbf{A} = \begin{pmatrix} [\boldsymbol{\omega}]_{\times} & \boldsymbol{\nu} \\ \mathbf{0} & 0 \end{pmatrix} \quad (3)$$

où  $\boldsymbol{\omega}$  est le vecteur de dimension ( $3 \times 1$ ) de vitesse de rotation, vérifiant  $\|\boldsymbol{\omega}\| < \pi$ , et  $\boldsymbol{\nu}$  est un vecteur de dimension ( $3 \times 1$ ) de vitesse de translation ;  $[\boldsymbol{\omega}]_{\times}$  est la matrice antisymétrique associée au vecteur  $\boldsymbol{\omega}$ .

On a :

$$\dot{\mathbf{x}}(t) = [\boldsymbol{\omega}]_{\times} \mathbf{x}(t) + \boldsymbol{\nu} \quad (4)$$

où  $\dot{\mathbf{x}}(t)$  représente un déplacement infinitésimal des coordonnées  $\mathbf{x}$  de l'origine du repère sur la trajectoire en fonction du temps  $t$  normalisé entre 0 et 1.

On a montré que l'intégrale de cette équation peut s'écrire sous la forme générale des équations paramétriques d'un cylindre :

$$\mathbf{x}(t) = \mathbf{x}_0 + \cos(\theta)\mathbf{u} + \sin(\theta)\mathbf{v} + \mathbf{w}t \quad (5)$$

avec  $\theta = \|\boldsymbol{\omega}t\|$ .

Les trois vecteurs  $\mathbf{u}$ ,  $\mathbf{v}$  et  $\mathbf{w}$  sont unitaires et forment un trièdre direct. Ils sont fonction, ainsi que  $\mathbf{x}_0$ , des différentes composantes de  $\boldsymbol{\nu}$  et  $\boldsymbol{\omega}$ . L'axe du cylindre défini par la géométrie de la tête stéréo, de vecteur unitaire  $\mathbf{w}$ , passe par le point  $\mathbf{x}_0$ .

A partir de ces résultats, on déduit quelle sera la trajectoire

<sup>1</sup>Référence FP6-GOCE-CT-2003-505342

<sup>2</sup>Groupe spécial euclidien de dimension trois

<sup>3</sup>Algèbre de Lie associé au groupe de Lie SE(3)

théorique suivie par les caméras en fonction de la géométrie de la tête stéréo. Par conséquent, il est possible de choisir la géométrie de la tête stéréo en fonction de la forme, du volume, et de l'orientation de l'objet sous-marin que l'on souhaite reconstruire.

### 3 Approche d'asservissement visuel retenue

Nous utilisons la méthode d'asservissement visuel proposée dans [2], car les paramètres intrinsèques des caméras constituant la paire stéréo sont différents.

On désire asservir la caméra gauche sur la position initiale de la caméra droite. En utilisant [1], on extrait des points d'intérêts dans les images des deux caméras de la paire stéréo. Une fois les  $n$  points appariés, on note  $\mathbf{p}_i^*$  les points provenant de la caméra droite, et  $\mathbf{p}_i$  les points provenant de la caméra gauche. Les points  $\mathbf{p}_i$  sont suivis dans l'image tout au long de l'asservissement par [3]. On construit les points  $\mathbf{q}_i^*$  et  $\mathbf{q}_i$ , appartenant à l'espace  $Q$  invariant aux paramètres intrinsèques des caméras [2], en projetant les coordonnées des points  $\mathbf{p}_i^* = (u_i^*, v_i^*, 1)$  et  $\mathbf{p}_i = (u_i, v_i, 1)$ . Ces nouveaux points sont alors contenus dans deux vecteurs :  $\mathbf{s}^* = (\mathbf{q}_1^*, \mathbf{q}_2^*, \dots, \mathbf{q}_n^*)$  et  $\mathbf{s} = (\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_n)$ . La caméra a convergé sur la position de référence lorsque  $\mathbf{s} = \mathbf{s}^*$ . La dérivée du vecteur  $\mathbf{s}$  s'écrit :

$$\dot{\mathbf{s}} = \mathbf{L}\mathbf{v} \quad (6)$$

où  $\mathbf{L}$  est la matrice d'interaction ( $3n \times 6$ ), et le vecteur  $\mathbf{v}$  ( $6 \times 1$ ) représente la vitesse Cartésienne de la caméra. La fonction de tâche calculée est :

$$\mathbf{e} = \hat{\mathbf{L}}^+(\mathbf{s} - \mathbf{s}^*) \quad (7)$$

où  $\hat{\mathbf{L}}^+$  est une approximation de la pseudo-inverse de  $\mathbf{L}$ . La commande utilisée est :

$$\mathbf{v} = -\lambda\mathbf{e}, \quad \text{où } \lambda > 0 \quad (8)$$

### 4 Expérimentation

Les expérimentations ont été menées à l'INRIA sur le robot Anis (Fig.1), et également à IFREMER sur le bras TAO (les deux bras ont 6 d.d.l.). Les Figures 1 représente une expérimentation réalisée en prenant un écart de 20 cm entre les caméras, et un angle de 15 degrés sur la caméra droite. Les images sont acquises à une fréquence de 25 images/sec. La cible 3D est représentée par trois cibles planes avec des vecteurs normaux différents, placées à environ 1,3 m des caméras. La Figure 2 représente la convergence exponentielle de la fonction de tâche vers zéro sur deux asservissements successifs. Les expérimentations menées ces derniers mois ont confirmé et validé nos résultats de simulation.

### 5 Conclusion

Cette nouvelle méthode permet l'acquisition d'images prises à intervalles réguliers le long d'une trajectoire pré-

définie, afin de faciliter le calcul de la reconstruction 3D d'un objet quelconque.

### 6 Remerciement

Je tiens à remercier tout particulièrement Selim BENHIMANE pour son aide sur les expérimentations.

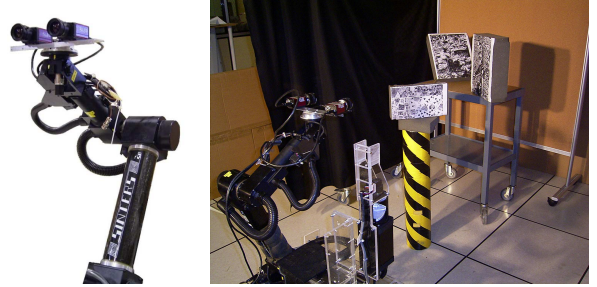


FIG. 1 – Expérimentation avec le robot Anis de l'INRIA

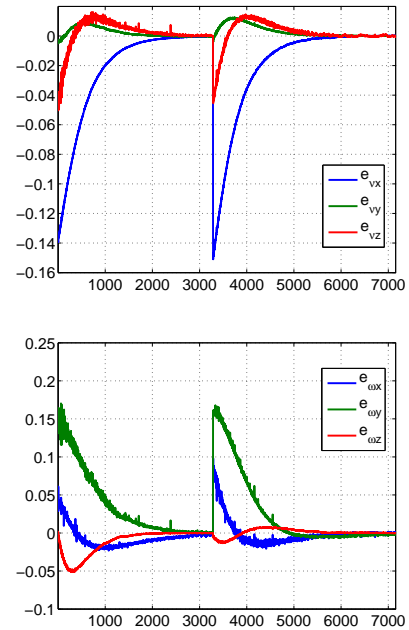


FIG. 2 – Fonction de tâche :  $\mathbf{e}_v$  et  $\mathbf{e}_\omega$

### Références

- [1] D.G. Lowe *Distinctive Image Features from Scale-Invariant Keypoints*, International Journal of Computer Vision, 60, 2 (2004), pp. 91-110.
- [2] E. Malis, *A Unified Approach to Model-based and Model-free Visual Servoing*, European Conference on Computer Vision, vol 4, pp. 433-447, Copenhagen, Denmark, May 2002.
- [3] S. Benhimane, E. Malis. *Real-time image-based tracking of planes using efficient second-order minimization*, IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 1, pp. 943-948, Sendai, Japan, October 2004.