



**HAL**  
open science

# Asservissement visuel d'un bras robotique en l'absence d'informations géométriques

Nicolas Andreff

► **To cite this version:**

Nicolas Andreff. Asservissement visuel d'un bras robotique en l'absence d'informations géométriques. Actes des Journées ORASIS GDR-PRC Communication Homme-Machine, May 1996, Clermont-Ferrand, France. inria-00590061

**HAL Id: inria-00590061**

**<https://inria.hal.science/inria-00590061>**

Submitted on 5 May 2011

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Asservissement visuel d'un bras robotique en l'absence d'informations géométriques

Nicolas Andreff

Equipe BIP

INRIA Rhône-Alpes/GRAVIR-IMAG

ZIRST, 655 avenue de l'Europe, 38330 Montbonnot Saint Martin

e-mail : andreff@imag.fr

Mai 1996

## Abstract

Cet article étend les résultats classiques d'asservissement visuel au cas non calibré. Ces extensions sont l'utilisation de paramètres physiques de caméra approximativement connus, l'hypothèse que la structure de la scène observée est inconnue et que l'on est capable de donner une estimation de la norme de la translation pince/caméra.

La méthode utilisée pour la commande du robot est la régulation par fonction de tâche, la tâche à effectuer étant le déplacement d'une caméra vis-à-vis d'un objet cible.

Cette méthode utilise le Jacobien de l'image qui dépend de la distance entre la caméra et la cible ainsi que de la structure euclidienne de cette dernière. Jusqu'à présent, seules étaient utilisées des approximations de ce Jacobien.

Le principal apport du travail effectué est l'inclusion du calcul du Jacobien par reconstruction euclidienne et de la transformation pince/caméra dans la structure de contrôle.

La méthode adoptée pour la reconstruction est une méthode itérative par approximations affines du modèle projectif de caméra.

**Mots-clés** — robotique, vision par ordinateur, asservissement visuel, fonctions de tâche, reconstruction 3D, coordination main/œil.

## 1 Introduction

Le problème présenté ici, plus détaillé dans [And95], consiste à déplacer une caméra non calibrée, montée sur un bras robotique à 6 degrés de liberté, jusqu'en une position où l'image obtenue correspond à une *image cible*. La clé de ce problème réside en la connaissance du Jacobien de l'image, élément principal de la loi de commande. Le Jacobien dépend des paramètres physiques de la caméra (*paramètres intrinsèques*), de la transformation rigide reliant la scène observée et la caméra (*géométrie*) et de la transformation rigide liant la caméra et l'outil terminal du robot (*transformation pince/caméra*).

Les paramètres intrinsèques sont inconnus puisque l'on utilise une caméra non calibrée. Il est, de plus, impossible, dans le cas le plus général, d'avoir une estimation *a priori* concernant la géométrie et la transformation pince/caméra. On cherche donc une solution permettant l'obtention de ces informations puis leur utilisation dans une boucle de commande. Cette solution doit, de plus, impliquer le moins de contraintes possible.

Il existe de nombreux algorithmes de reconstruction tridimensionnelle de scènes. Cependant, la plupart se basent sur des calculs d'optimisation non-linéaires, ce qui engendre des temps de calcul trop importants pour pouvoir inclure de telles méthodes dans une boucle de commande.

Il faut donc trouver un moyen d'estimer rapidement le Jacobien. Plusieurs travaux ont été menés dans cette optique, mais ils sont soit complexes, soit limités par trop de contraintes.

On citera [SP94] qui offre une solution peu intuitive d'estimation du Jacobien et de la commande à partir d'un filtre de Kalman généralisé, [YA94] qui se limite à des mouvements plans, [CAD95] qui travaille sur des objets plans et [SBC94] qui utilise la notion de foyer d'expansion de l'image.

La solution que je propose ici consiste à intégrer dans une loi de commande exprimée selon le formalisme des fonctions de tâches, une méthode de reconstruction tridimensionnelle par approximation affine du modèle projectif de la caméra [CH94], déjà utilisée avec succès en asservissement visuel [HCD94], et une méthode d'estimation de la transformation caméra/pince basée sur cette approximation affine [Mor95].

Les contraintes imposées par cette solution sont les suivantes : connaissance très approximative des paramètres intrinsèques (données constructeur, par exemple), marquage d'un minimum de 4 points non coplanaires quelconques de l'objet cible par des gommettes (pour la simplification du suivi) et connaissance de l'ordre de grandeur de la norme de la translation pince/caméra.

Après avoir défini la loi de commande, exhibant le Jacobien, nous rappellerons la forme générale de celui-ci en faisant apparaître tous les paramètres dont il dépend. Puis, nous aborderons succinctement les questions de reconstruction sous la forme de pointeurs vers les travaux originaux. Nous présenterons ensuite la méthode utilisée pour le suivi des points et leur mise en correspondance. Alors, il sera temps de donner l'algorithme d'asservissement visuel, base de cet article. Une courte discussion s'ensuivra concernant la mise à jour du Jacobien. Cette discussion sera naturellement suivie d'une courte conclusion.

## 2 Commande

### 2.1 Commande par fonction de tâche

L'asservissement visuel correspond à la minimisation de l'erreur entre l'image courante ( $s$ ) et l'image cible ( $s^*$ ). Cette minimisation est un cas particulier de la commande par fonction de tâche [ECR92, SLBE91, Cha90] : dans le cadre du problème présenté, la minimisation est effectuée en régulant à zéro la fonction de tâche suivante :

$$e = J^+(s(\bar{r}, t) - s^*(t))$$

où  $\bar{r}$ , élément du groupe des déplacements, représente la position du robot et  $J^+$  est la matrice pseudo-inverse du Jacobien de l'image.

On désire que la fonction de tâche se comporte comme un système découplé du premier ordre, à savoir  $\dot{e} = -\lambda e$ . Cela signifie que l'on souhaite que le système converge de manière exponentielle vers son état final.

De plus, sous certaines conditions concernant le robot (connaissance parfaite du jacobien inverse et commande par la vitesse articulaire), on peut utiliser le torseur cinématique  $T_c$  comme pseudo-commande. L'expression de ce torseur est alors dans ce cas simplifié :

$$T_c = -\lambda e$$

### 2.2 Forme générale du Jacobien

Il est possible d'exprimer le Jacobien,  $J$ , sous une forme générale permettant son calcul dans un repère quelconque [Esp93].

Soient  $R_p$ , le repère associé à la pince (ou plus généralement, à l'outil terminal du robot), dans lequel on souhaite exprimer le Jacobien, et  $R_c$ , le repère associé à la caméra. Ces deux repères peuvent être rigidement liés (caméra fixée sur l'effecteur du robot) ou en mouvement relatif (observation du mouvement du robot avec une caméra fixe).

Notons  $\bar{X}^T = (XYZ)^T$  les coordonnées d'un point  $\bar{M}$  dans  $R_p$  et  $s = (u, v)^T$  les coordonnées de sa projection dans le repère image. On a alors la relation suivante :  $\lambda (s, 1)^T = P (\bar{X}, 1)^T$ , où  $P$  est la matrice

de projection du repère pince dans le repère image. Cette matrice peut s'écrire comme composition de changements de repère :

$$P = A \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} Y$$

où

- $A$  est la matrice de changement de repère rétine→image. Elle contient les 5 paramètres intrinsèques de la caméra et est égale à :

$$A = \begin{pmatrix} A_{11} & \alpha \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} -\alpha_u & \alpha_u \cot \theta & u_0 \\ 0 & -\frac{\alpha_v}{\sin \theta} & v_0 \\ 0 & 0 & 1 \end{pmatrix}$$

Ces paramètres sont les facteurs d'échelle sur chaque axe ( $\alpha_u$  et  $\alpha_v$ ), les coordonnées dans le repère de la rétine de l'intersection des axes ( $u_0$  et  $v_0$ ) et l'angle qu'ils forment ( $\theta$ )<sup>1</sup>.

- $Y$  est la matrice de changement de repère pince→caméra. Elle est composée d'une rotation  $R$  et d'une translation  $t$  :

$$Y = \begin{pmatrix} R & t \\ 0 & 1 \end{pmatrix}$$

Notant  $\bar{X}' \triangleq (x, y, z)^T$ , les coordonnées de  $\bar{M}$  dans le repère caméra, on a, de plus, la relation :  $(\bar{X}', 1)^T = Y (\bar{X}, 1)^T$ . Introduisons, à présent,  $J_c$ , le Jacobien exprimé dans le repère caméra. Il s'écrit :

$$J_c(\bar{X}') = \begin{pmatrix} -1/z & 0 & y/z^2 & xy/z^2 & -(1+x^2/z^2) & y/z \\ 0 & -1/z & y/z^2 & 1+y^2/z^2 & -xy/z^2 & -x/z \end{pmatrix}$$

Enfin, en utilisant la matrice  $\Theta$  suivante :

$$\Theta = \begin{pmatrix} R & -RA_s(-R^T t) \\ 0 & R \end{pmatrix},$$

on peut écrire le Jacobien exprimé dans le repère pince :

$$J = A_{11} J_c(\bar{X}') \Theta.$$

Cette expression fait clairement apparaître les paramètres dont dépend le Jacobien : paramètres intrinsèques de la caméra, la géométrie de la scène et la transformation pince/caméra.

### 3 Reconstruction euclidienne et transformation pince/caméra

Nous ne détaillerons pas ces étapes, bien mieux présentées par leurs auteurs.

La méthode de reconstruction utilisée ici est celle présentée dans [CH94]. Il s'agit d'une méthode itérative dont chaque itération comporte une reconstruction affine et une reconstruction euclidienne par approximation paraperspective. L'approximation paraperspective est une approximation au premier ordre du modèle projectif de caméra qui permet de linéariser les équations de reconstruction et donc d'accélérer les calculs. Elle dépend de la géométrie.

Le but des itérations de la méthode est d'affiner l'approximation. En effet, la reconstruction affine est effectuée à partir de l'approximation courante dont la précision influe sur la qualité de reconstruction affine. Puis, la reconstruction euclidienne est lancée en se basant sur les résultats affines. On peut alors améliorer l'approximation paraperspective et par suite, la reconstruction euclidienne.

Cette méthode permet l'estimation, au signe et à un facteur d'échelle près, de la géométrie à partir des projections dans au moins 3 images, d'un minimum de 4 points non coplanaires (*points d'intérêt*) pris dans la scène et de la valeur, même très approximative, des paramètres intrinsèques.

<sup>1</sup>Généralement très proche de  $\Pi/2$  et, par conséquent, approximé par cette valeur.

Quant au calcul de la transformation pince/caméra[Mor95], il est effectué sous les mêmes hypothèses auxquelles il faut ajouter la détermination de 2 déplacements indépendants du robot qui séparent les 3 premières images. Cependant, il ne fournit pas la norme de la partie translationnelle de cette transformation (*i.e.* la distance pince/caméra). Les conséquences sur le Jacobien d'une erreur sur cette norme ne sont pas triviales puisque cette norme intervient dans le calcul de celui-ci au niveau de la matrice  $\Theta$ . Cela impose donc de connaître, à défaut de mieux et, espérons-le, de manière provisoire, un ordre de grandeur de cette distance.

Le signe et le facteur d'échelle, laissés indéterminés, peuvent être obtenus grâce au mouvement du robot. En effet, le Jacobien dépend linéairement de ces deux paramètres et d'après notre loi de commande, le torseur de commande aussi. Pour retrouver les bonnes valeurs, il n'y a plus qu'à comparer le mouvement effectif du robot lors des 2 premiers déplacements et le mouvement qu'aurait eu le robot s'il avait suivi la loi de commande sur ce chemin.

## 4 Suivi des points

Les deux méthodes précédentes nécessitent le suivi des points d'intérêt et de la mise en correspondance de leurs projections successives. Pour accélérer la localisation de ces points dans l'image, deux dispositions ont été prises. La première, purement matérielle, consiste à marquer ces points de pastilles blanches, afin de simplifier la détection bas-niveau. La seconde est d'ordre logiciel : on prédit la position de ces points grâce à la valeur estimée du Jacobien. Cette prédiction est obtenue en intégrant l'équation qui relie image et mouvement (torseur cinématique) du robot :

$$\dot{s} = J \cdot \tau.$$

En supposant le Jacobien et le torseur cinématique constants sur l'intervalle de temps  $\Delta t$  séparant deux prises d'images successives et en les approximant respectivement par  $\hat{J}$ , Jacobien calculé, et  $T_c$ , torseur de commande, on obtient l'équation de prédiction :

$$s_{pred}(t + \Delta t) = s(t) + \hat{J} \cdot T_c \cdot \Delta t.$$

Il est désormais possible d'effectuer une détection efficace en recherchant les points d'intérêt dans un voisinage de la valeur prédite. Ce voisinage doit être suffisamment petit pour ne pas prendre un point d'intérêt pour un autre. Cependant, un voisinage trop petit peut se révéler néfaste dans le cas où la prédiction s'avère erronée (perturbations, temps d'application du torseur de commande trop long (ou trop court), etc.) car alors, le point à détecter peut se retrouver hors du voisinage.

La solution suivante a donc été mise au point :

1. Prédire la prochaine position des points d'intérêt.
2. Fixer la taille des fenêtres de prédiction (10 pixels de côté)
3. Déterminer une petite fenêtre autour de chacune des positions prédites.
4. Pour chaque fenêtre, détecter le point d'intérêt (centre d'une tâche claire).
5. Si l'une des détections a échoué et si les fenêtres sont plus petites que l'image complète, alors augmenter la taille des fenêtres (+10 pixels) et retourner en 3.

A la suite de cette boucle, deux cas se présentent : soit la détection a bien fonctionné (*i.e.* on obtient autant de points distincts que de points d'intérêt), soit certains points obtenus sont doubles. Il y a alors occlusion d'un point et il faut mettre en œuvre un traitement d'exception (actuellement, arrêt du système).

## 5 Algorithme

On peut désormais écrire l'algorithme d'asservissement visuel :

- Initialisation
  - Prendre une image en position initiale ( $s_1$ )
  - Effectuer une reconstruction affine avec cette image et l'image cible
  - Déplacer le robot et prendre une deuxième image ( $s_2$ )
  - Déplacer le robot et prendre une troisième image ( $s_3$ )
  - Effectuer une reconstruction euclidienne par approximation affine sur les 4 images disponibles
  - Estimer la transformation pince/caméra ( $Y$ ) à l'aide des 3 dernières images et les 2 déplacements du robot
  - Calculer le Jacobien dans le repère du robot et sa pseudo-inverse
- Boucle de commande

**Pour  $j > 3$  faire :**

  - Evaluer la fonction de tâche :  $e = J^+(s_j - s^*)$
  - Si  $|e| \leq \text{seuil}$  alors STOP
  - Appliquer le torseur de commande  $T_c = -\lambda e$  au robot
  - Prendre une nouvelle image  $s_{j+1}$  en s'aidant de la prédiction

**Fin faire**

Cet algorithme a été mis en œuvre sur les robots SCEMI (bras articulés à 6 degrés de liberté) du LIFIA en utilisant les bibliothèques de programmes du projet MOVI.

## 6 Stratégies de mise à jour

On remarquera qu'il n'est pas question de mise à jour de la valeur du Jacobien, ni de celle de la transformation pince/caméra. Cette mise à jour peut être faite selon plusieurs stratégies.

La première consiste à faire une nouvelle estimation à chaque itération de la boucle de commande en utilisant toutes les images prises depuis le début. Mais c'est une solution gourmande en mémoire. On peut alors se limiter aux  $n$  dernières images (solution choisie (avec  $n = 5$ ) lors de l'implémentation), mais cela s'est avéré être une mauvaise idée. En effet, si cette solution fonctionne bien sur une séquence préenregistrée dont les images présentent des différences notables, elle ne fonctionne pas en pratique car alors, les  $n$  dernières images ne diffèrent pas suffisamment les unes des autres puisque le robot converge vers l'image cible. Partant, les qualités de reconstruction et d'estimation du Jacobien diminuent. On perd, du reste, le bénéfice des 2 mouvements indépendants du départ.

Il faut donc trouver des stratégies plus appropriées. Deux voies se présentent : une stratégie de mise à jour conditionnelle et une stratégie de modification en parallèle. La stratégie conditionnelle consiste à définir une condition logique qui, lorsqu'elle est satisfaite, déclenche un nouveau calcul avant de relancer une nouvelle itération de la boucle de commande. L'inconvénient d'une telle solution est qu'elle impose de nouvelles contraintes sur le temps de calcul.

En revanche, le calcul en parallèle semble plus attractif. En effet, il y a moins de contraintes de type temps-réel : au lieu de devoir effectuer les calculs en une itération de la boucle de commande, on dispose de quelques itérations supplémentaires. Un autre avantage est que l'on peut dans ce cas mettre en place un traitement de type filtrage.

## 7 Conclusion

La réponse à la discussion précédente reste un problème ouvert. On s'attachera aussi à améliorer le calcul de la transformation pince/caméra. Ces problèmes résolus, on pourra alors envisager d'intégrer des méthodes plus efficaces de détection et de mise en correspondance des points d'intérêt et se passer ainsi du marquage de l'objet cible par des gommettes.

**Remerciements** — Je tiens à remercier Bernard Espiau et Radu Horaud pour l'aide précieuse qu'ils m'ont apportée au cours de mon projet de DEA, base de cet article.

## References

- [And95] Nicolas Andreff. Asservissement visuel et calibration faible. Dea, INPG, Juin 1995.
- [CAD95] C. Colombo, B. Allota, and P. Dario. Affine Visual Servoing: A Framework for Relative Positioning with a Robot. In *Proc. IEEE International Conference on Robotics and Automation*, 1995.
- [CH94] S. Christy and R. Horaud. Euclidean Shape and Motion from Multiple Perspective Views by Affine Iterations. Rapport de Recherche 2421, INRIA, Décembre 1994.
- [Cha90] F. Chaumette. *La relation vision-commande : théorie et application à des tâches robotiques*. Thèse, Université de Rennes I, 1990.
- [ECR92] B. Espiau, F. Chaumette, and P. Rives. A New Approach To Visual Servoing in Robotics. *IEEE Trans. on Robotics and Automation*, 8(3), June 1992.
- [Esp93] B. Espiau. Effect of Camera Calibration Errors on Visual Servoing in Robotics. In *Third International Symposium on Experimental Robotics*, October 1993. Kyoto, Japan.
- [HCD94] R. Horaud, S. Christy, and F. Dornaika. Object Pose : The Link between Weak Perspective, Para Perspective and Full Perspective. Rapport de Recherche 2356, INRIA, September 1994.
- [Mor95] H. Mortimore. Reconstruction Euclidienne d'une Scène Tridimensionnelle avec une Caméra Affine non Calibrée Montée sur un Bras de Robot. DEA, INPG, 1995.
- [SBC94] V. Sundaeswaran, P. Bouthemy, and F. Chaumette. Visual servoing using dynamic image parameters. Rapport de Recherche 2336, INRIA, Août 1994.
- [SLBE91] C. Samson, M. Le Borgne, and B. Espiau. *Robot Control : The Task Function Approach*. Clarendon Press, Oxford, 1991.
- [SP94] S. Soatto and P. Perona. Structure-Independent Visual Motion Control on the Essential Manifold. In *Preprints of the Fourth IFAC Symposium on Robot Control*, September 1994. Capri, Italy.
- [YA94] B.H. Yoshimi and P.K. Allen. Active, Uncalibrated Visual Servoing. In *icra*, pages 156–161, 1994.