



**HAL**  
open science

# Réalisation et calibration d'un système expérimental de vision composé d'une caméra mobile embarquée sur un robot-manipulateur

François Chaumette, Patrick Rives

## ► To cite this version:

François Chaumette, Patrick Rives. Réalisation et calibration d'un système expérimental de vision composé d'une caméra mobile embarquée sur un robot-manipulateur. [Rapport de recherche] RR-0994, INRIA. 1989. inria-00075565

**HAL Id: inria-00075565**

**<https://inria.hal.science/inria-00075565>**

Submitted on 24 May 2006

**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.

# INRIA

UNITE DE RECHERCHE  
INRIA-RENNES

Institut National  
de Recherche  
en Informatique  
et en Automatique

Domaine de Voluceau  
Rocquencourt  
BP 105  
78153 Le Chesnay Cedex  
France  
Tel. (1) 39 63 55 11

## Rapports de Recherche

N° 994

*Programme 6  
Robotique, image et vision.*

### REALISATION ET CALIBRATION D'UN SYSTEME EXPERIMENTAL DE VISION COMPOSE D'UNE CAMERA MOBILE EMBARQUEE SUR UN ROBOT-MANIPULATEUR

François CHAUMETTE  
Patrick RIVES

Mars 1989



507

# IRISA

INSTITUT DE RECHERCHE EN INFORMATIQUE  
ET SYSTÈMES ALÉATOIRES

Campus Universitaire de Beaulieu  
35042 - RENNES CÉDEX  
FRANCE  
Téléphone : 99 36 20 00  
Télex : UNIRISA 950 473 F  
Télécopie : 99 38 38 32

**REALISATION ET CALIBRATION D'UN SYSTEME  
EXPERIMENTAL DE VISION COMPOSE D'UNE CAMERA MOBILE  
EMBARQUEE SUR UN ROBOT-MANIPULATEUR**

**DESIGN AND CALIBRATION OF AN EXPERIMENTAL VISION  
SYSTEM BASED ON A MOBILE CAMERA MOUNTED ON A  
ROBOT-ARM**

François CHAUMETTE et Patrick RIVES

IRISA / INRIA-Rennes  
Campus de Beaulieu  
35042 Rennes Cedex

Publication Interne n° 454 - Février 1989 - 36 Pages

**Resumé :** L'exploitation des informations fournies par une caméra montée sur l'organe terminal d'un robot manipulateur nécessite, au préalable, une phase de calibration de l'ensemble du système. Ce problème de calibration peut se scinder en trois problèmes d'identification :

- L'identification des zéros articulaires et des divers paramètres du robot afin de pouvoir utiliser les transformateurs de coordonnées nécessaires à la commande.
- L'identification des paramètres intrinsèques de la caméra afin de connaître la transformation repère scène-repère caméra.
- L'identification de la liaison rigide (position et orientation) liant le repère caméra au repère de l'effecteur sur lequel elle se trouve fixée.

On montre dans ce rapport comment l'utilisation des informations fournies par la caméra peut apporter une aide précieuse à la résolution du premier type de problème. Concernant l'identification du modèle de la caméra, elle est faite en deux temps. Tout d'abord, la résolution d'un système linéaire permet d'obtenir des résultats approchés des paramètres; malheureusement, ces résultats sont biaisés en présence de bruits. Dans un deuxième temps, ces résultats sont utilisés comme estimées initiales en entrée d'une méthode de minimisation non linéaire robuste. Le troisième problème d'identification est résolu en utilisant des déplacements du manipulateur.

**Abstract :** In order to exploit the information given by a camera mounted on the end effector of a robot-arm, one requires a calibration phase. This calibration problem can be subdivided as follows into three identification problems :

- The identification of the reference position and the different parameters of the robot, in order to be able to use the coordinate transformers which are necessary for the robot control.
- The identification of the intrinsic parameters of the camera in order to compute the transformation from the object coordinate system to the camera coordinate system.
- The identification of the rigid body (position and orientation) linking the camera coordinate system to the effector coordinate system.

In this report, one shows how the information given by the camera can greatly help in solving the first subproblem. The identification problem of the camera model is twofold. First, the solving of a system of linear equation yields approximate values for the parameters; however, these are corrupted in the presence of noise. They are thus used as initial estimates in a robust non-linear minimization. The third problem is solved using at least three displacements of the robot.

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Notations . . . . .	3
1.2	Description et méthodologie . . . . .	4
1.2.1	Description . . . . .	4
1.2.2	Méthodologie . . . . .	6
<b>2</b>	<b>Réglage des zéros angulaires d'un robot à l'aide d'une caméra embarquée</b>	<b>8</b>
2.1	Description . . . . .	8
2.2	Réglage de $q_{05}$ . . . . .	10
2.3	Réglage de $q_{02}$ et $q_{03}$ . . . . .	11
2.4	Réglage de $q_{04}$ . . . . .	12
2.5	Résultats expérimentaux . . . . .	13
<b>3</b>	<b>Modélisation et calibration de la caméra</b>	<b>14</b>
3.1	Mise en équations . . . . .	14
3.2	Résolution par un système linéaire – Simulation . . . . .	17
3.3	Résolution par une méthode non linéaire – Simulation . . . . .	20
3.4	Résultats expérimentaux . . . . .	22
3.5	Simplifications . . . . .	24
<b>4</b>	<b>Identification de la matrice de passage entre le poignet du robot et la caméra</b>	<b>26</b>
4.1	Mise en équations . . . . .	26
4.2	Résolution – Discussion . . . . .	28
<b>5</b>	<b>Validation et Conclusion</b>	<b>30</b>
5.1	Validation des résultats . . . . .	30
5.2	Conclusion . . . . .	31

# Chapitre 1

## Introduction

L'objet du banc expérimental que nous allons décrire dans ce rapport est l'étude de la vision en mouvement à partir d'une caméra mobile. Les mouvements de la caméra, calculés soit à partir d'une trajectoire prédéfinie (Boucle ouverte), soit à partir des informations fournies par la caméra elle-même (Boucle fermée), sont effectués par un robot manipulateur.

Il est donc indispensable de pouvoir exprimer les mouvements de la caméra en fonction des mouvements du manipulateur et d'assurer la précision des positions obtenues. Par ailleurs, il est nécessaire d'identifier les paramètres caractéristiques de la caméra afin de pouvoir exploiter les informations qu'elle fournit au sein de tâches robotiques.

### 1.1 Notations

Soient  $R_i$  et  $R_j$  deux repères orthonormés  $(O_i, \vec{x}_i, \vec{y}_i, \vec{z}_i)$  et  $(O_j, \vec{x}_j, \vec{y}_j, \vec{z}_j)$ , on représente l'orientation et la position de  $R_j$  par rapport à  $R_i$  par la matrice homogène  $M_i^j$  :

$$M_i^j = \begin{pmatrix} R_i^j & T_i^j \\ 0 & 1 \end{pmatrix} \quad (1.1)$$

$T_i^j$  est le vecteur de translation correspondant aux coordonnées de  $O_j$  dans  $R_i$  et  $R_i^j$  la matrice de rotation entre  $(\vec{x}_i, \vec{y}_i, \vec{z}_i)$  et  $(\vec{x}_j, \vec{y}_j, \vec{z}_j)$  (voir figure 1.1).

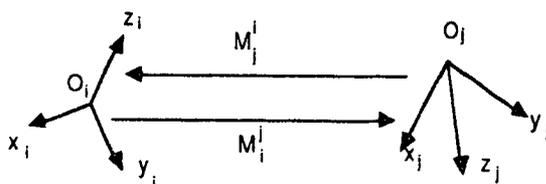


Figure 1.1 : Changement de repère

On rappelle que :

$$M_i^j . M_j^k = M_i^k \quad (1.2)$$

et

$$M_j^i = M_i^{j^{-1}} = \begin{pmatrix} R_i^{j^T} & -R_i^{j^T} . T_i^j \\ 0 & 1 \end{pmatrix} \quad (1.3)$$

Dans la suite de ce rapport, les matrices homogènes seront écrites sous la forme réduite :

$$M_i^j = \begin{pmatrix} R_i^j & T_i^j \end{pmatrix} \quad (1.4)$$

De plus, les coefficients d'une matrice de rotation R seront notés  $R_{[ij]}$  et une matrice M dépendant du temps sera notée  $M_{(t)}$  à l'instant t.

Enfin, nous aurons besoin de la représentation des rotations sous leur forme matricielle, avec les propriétés bien connues d'orthonormalité des lignes et colonnes d'une matrice de rotation, mais également sous leur forme  $(U, \theta)$  où U est l'axe normé de la rotation et  $\theta$  l'angle de rotation autour de cet axe. Le passage d'une représentation à l'autre est immédiate :

- Soit  $(U = (U_x, U_y, U_z)^T, \theta)$  la représentation d'une rotation et R la matrice de rotation correspondante, on a :

$$R = \cos \theta . I_3 + (1 - \cos \theta) . UU^T + \sin \theta . As(U) \quad (1.5)$$

avec  $I_3$  la matrice identité et  $As(U)$  la matrice antisymétrique :

$$As(U) = \begin{pmatrix} 0 & -U_z & U_y \\ U_z & 0 & -U_x \\ -U_y & U_x & 0 \end{pmatrix}$$

- Réciproquement, on a :

$$\cos \theta = \frac{1}{2} (R_{[11]} . R_{[22]} . R_{[33]} - 1) \quad (1.6)$$

et

$$\sin \theta . As(U) = \frac{1}{2} (R - R^T) \quad (1.7)$$

En se fixant  $\theta$  positif, U est déterminé de manière unique si  $\sin \theta \neq 0$ .

## 1.2 Description et méthodologie

### 1.2.1 Description

Le système expérimental est constitué par un robot manipulateur A.I.D. à six degrés de liberté sur le poignet duquel est fixée une caméra CCD MICAM. Au robot manipulateur est associé un processeur de commande en position (ROBOSOFT) et à la caméra est associé un processeur d'acquisition et de traitement d'images (EDIXIA). Le tout est piloté à partir d'une station SUN sous système UNIX (voir figures 1.2 et 1.3 ).

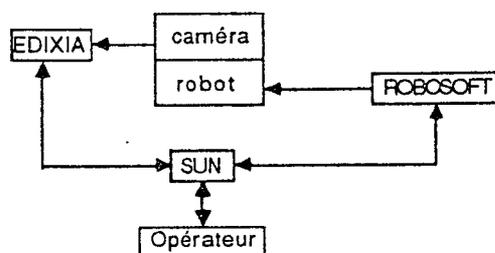


Figure 1.2 : Système expérimental

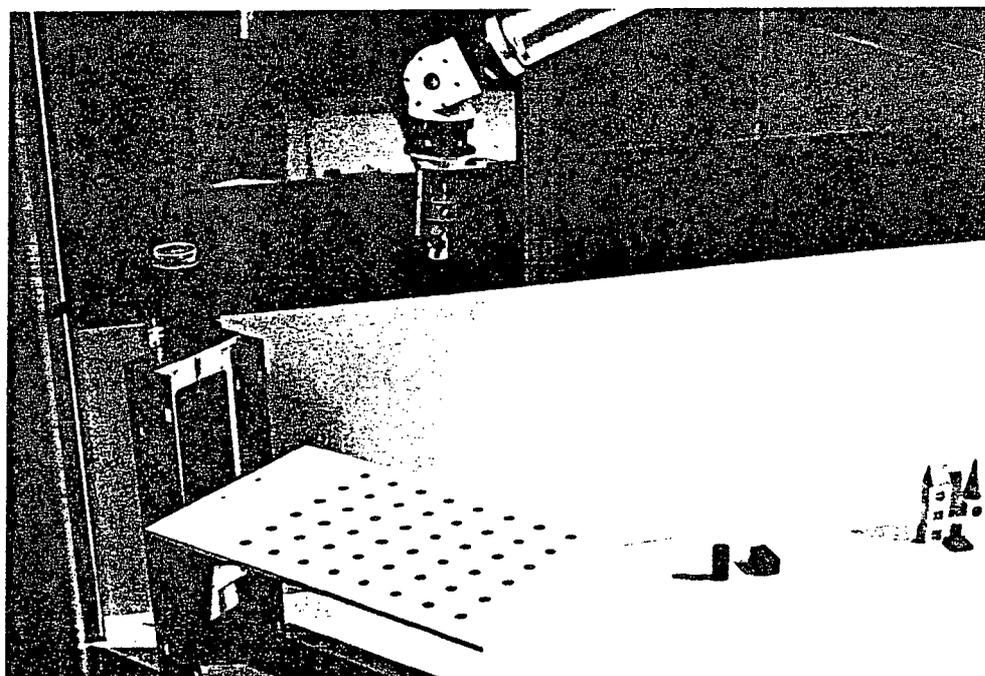


Figure 1.3 : Le poignet du robot, la caméra et la mire utilisée pour la calibration

## 1.2.2 Méthodologie

Les différents repères mis en jeu sont illustrés sur la figure 1.4 :

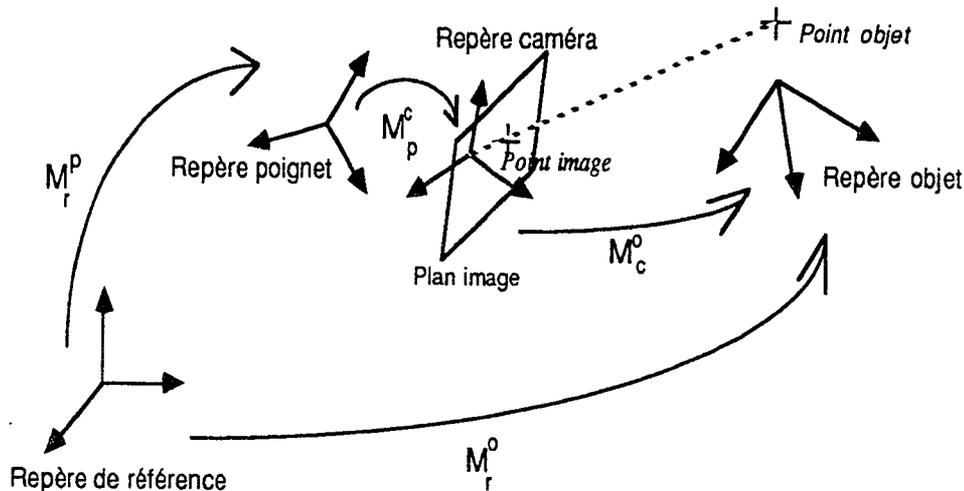


Figure 1.4 : Les différents repères utilisés

Pour pouvoir contrôler les mouvements de la caméra, il est nécessaire de connaître à chaque instant la matrice  $M_r^c$  représentant la position et l'orientation de la caméra par rapport à un repère de référence associé au robot. Cette matrice résulte de la composition de deux matrices :  $M_r^p$ , correspondant à la transformation repère de référence-repère de l'organe terminal, et  $M_p^c$  représentant la transformation repère de l'organe terminal-repère caméra.

- Le calcul de  $M_r^p$  constitue l'identification du modèle géométrique du robot, c'est à dire la relation entre les coordonnées articulaires et la matrice  $M_r^p$ .
- Le calcul de  $M_p^c$  se fait par le biais de l'introduction d'un repère supplémentaire fixe : le repère objet dans lequel la scène vue par la caméra est référencée. On a alors :

$$M_r^o = M_r^p \cdot M_p^c \cdot M_c^o = cte \quad (1.8)$$

Le calcul de  $M_c^o$  sous-entend le choix d'un modèle de la caméra. Une identification préalable des paramètres du modèle choisi est donc nécessaire. Ce problème classique, la calibration, permet de lier un point (3D), visible par la caméra, à sa position dans l'image (2D).

Enfin, comme il est impossible de connaître précisément  $M_r^o$  (aucune hypothèse sur la position de la scène par rapport au repère de référence n'est formulée), on utilisera des déplacements du robot et des calibrations à chaque position correspondante de la caméra pour calculer  $M_p^c$  :

$$M_{r(1)}^p \cdot M_p^c \cdot M_{c(1)}^o = M_{r(2)}^p \cdot M_p^c \cdot M_{c(2)}^o = \dots \quad (1.9)$$

Il est à remarquer que l'identification du robot, des paramètres du modèle de la caméra et de la matrice  $M_p^c$  n'est à effectuer qu'une seule fois. Aussi les différentes méthodes employées ont-elles été choisies pour leur précision et leur robustesse.

## Chapitre 2

# Réglage des zéros angulaires d'un robot à l'aide d'une caméra embarquée

### 2.1 Description

Le robot manipulateur dont nous disposons est un A.I.D. V5 à six degrés de liberté, à articulations rotoïdes et concourantes. Les différents repères de la chaîne articulée et leur orientation de référence sont représentés sur la figure 2.1 (les origines des repères  $R_4$ ,  $R_5$  et  $R_6$  sont confondues, formant une liaison rotule). Par convention, l'axe  $z$  de chaque repère coïncide avec l'axe de rotation de l'articulation correspondante.

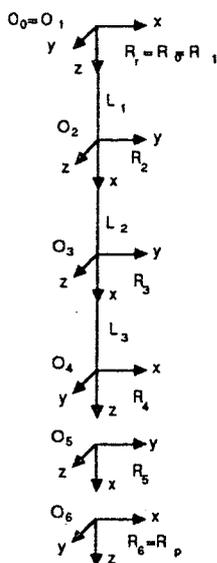


Figure 2.1 : Configuration de référence

A la configuration initiale représentée sur la figure 2.1 correspondent les valeurs des six coordonnées articulaires  $q_{0i}$ . Il est primordial de repérer exactement ces valeurs  $q_{0i}$  par rapport à une position connue du robot (par exemple les butées mécaniques) afin de pouvoir utiliser un transformateur de coordonnées donnant la position et l'orientation du poignet ( $R_p$ ) par rapport à un repère fixe de référence ( $R_r$ ). En effet, on a, pour une position quelconque  $q_i$  :

$$M_r^p = M_0^6 = M_0^1 \cdot M_1^2 \cdot M_2^3 \cdot M_3^4 \cdot M_4^5 \cdot M_5^6 \quad (2.1)$$

avec :

$$M_0^1 = \begin{pmatrix} \cos(q_1 - q_{01}) & -\sin(q_1 - q_{01}) & 0 & 0 \\ \sin(q_1 - q_{01}) & \cos(q_1 - q_{01}) & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$M_1^2 = \begin{pmatrix} -\sin(q_2 - q_{02}) & \cos(q_2 - q_{02}) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \cos(q_2 - q_{02}) & \sin(q_2 - q_{02}) & 0 & L_1 \end{pmatrix}$$

$$M_2^3 = \begin{pmatrix} \cos(q_3 - q_{03}) & -\sin(q_3 - q_{03}) & 0 & L_2 \\ \sin(q_3 - q_{03}) & \cos(q_3 - q_{03}) & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$M_3^4 = \begin{pmatrix} 0 & 0 & 1 & L_3 \\ \cos(q_4 - q_{04}) & -\sin(q_4 - q_{04}) & 0 & 0 \\ \sin(q_4 - q_{04}) & \cos(q_4 - q_{04}) & 0 & 0 \end{pmatrix}$$

$$M_4^5 = \begin{pmatrix} \sin(q_5 - q_{05}) & \cos(q_5 - q_{05}) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \cos(q_5 - q_{05}) & -\sin(q_5 - q_{05}) & 0 & 0 \end{pmatrix}$$

$$M_5^6 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ \cos(q_6 - q_{06}) & -\sin(q_6 - q_{06}) & 0 & 0 \\ \sin(q_6 - q_{06}) & \cos(q_6 - q_{06}) & 0 & 0 \end{pmatrix}$$

Les longueurs  $L_i$  des trois segments du manipulateur ayant déjà été mesurées avec une précision de l'ordre du dixième de millimètre lors d'expérimentations précédentes [BRU 87], seule l'identification des valeurs initiales  $q_{0i}$  des coordonnées articulaires sera détaillée ici. On remarque, de plus, que la valeur  $q_{01}$  dépend uniquement du choix du repère de référence et que la valeur  $q_{06}$  peut être choisie quelconque : en effet, la matrice de passage entre le repère  $R_6$  et le repère de la caméra qui, seul, nous intéresse, sera identifiée ultérieurement et un éventuel décalage de  $q_{06}$  sera donc compensé lors de cette identification.

Le problème se ramène donc à l'identification précise des valeurs  $q_{02}$  à  $q_{05}$  correspondant aux articulations à l'intérieur de la chaîne cinématique du manipulateur.

La méthode de réglage utilise les propriétés des singularités du manipulateur et la possibilité d'accéder à des mesures d'erreurs très faibles par le biais de la caméra. En effet, à une singularité correspond un ensemble de combinaisons des coordonnées articulaires amenant une position et une orientation constantes du poignet. Les images acquises par la caméra sont donc identiques lorsque le vecteur de coordonnées articulaires appartient à cet ensemble de combinaisons.

## 2.2 Réglage de $q_{05}$

- Pour une position initiale  $q_1, q_2, q_3, q_4, q_5, q_6$  donnant  $M_{0(1)}^6$  et pour une position finale  $q_1, q_2, q_3, q_4 + \delta, q_5, q_6 - \delta$  donnant  $M_{0(2)}^6$ , avec  $q_1, q_2, q_3, q_4, q_5, q_6$  et  $\delta (\neq 0)$  quelconques, on montre que :

$$M_{0(1)}^6 = M_{0(2)}^6 \text{ si et seulement si } q_5 = q_{05} \text{ ou } q_5 = q_{05} + \pi \quad (2.2)$$

- En effet, on a évidemment  $M_{0(1)}^3 = M_{0(2)}^3$  et  $T_{0(1)}^6 = T_{0(2)}^6$  puisque  $q_1, q_2$  et  $q_3$  sont constants.

De plus, en notant :

$$S_i = \sin(q_i - q_{0i}), \quad S_{i+\delta} = \sin(q_i - q_{0i} + \delta), \quad S_{i+j} = \sin(q_i - q_{0i} + q_j - q_{0j})$$

$$C_i = \cos(q_i - q_{0i}), \quad C_{i+\delta} = \cos(q_i - q_{0i} + \delta), \quad C_{i+j} = \cos(q_i - q_{0i} + q_j - q_{0j})$$

on obtient :

$$R_{3(1)}^6 = \begin{pmatrix} -S_5 C_6 & S_5 S_6 & C_5 \\ C_4 C_5 C_6 - S_4 S_6 & -C_4 C_5 S_6 - S_4 C_6 & S_5 C_4 \\ S_4 C_5 C_6 + C_4 S_6 & -S_4 C_5 S_6 + C_4 C_6 & S_5 S_4 \end{pmatrix}$$

et

$$R_{3(2)}^6 = \begin{pmatrix} -S_5 C_{6-\delta} & S_5 S_{6-\delta} & C_5 \\ C_{4+\delta} C_5 C_{6-\delta} - S_{4+\delta} S_{6-\delta} & -C_{4+\delta} C_5 S_{6-\delta} - S_{4+\delta} C_{6-\delta} & S_5 C_{4+\delta} \\ S_{4+\delta} C_5 C_{6-\delta} + C_{4+\delta} S_{6-\delta} & -S_{4+\delta} C_5 S_{6-\delta} + C_{4+\delta} C_{6-\delta} & S_5 S_{4+\delta} \end{pmatrix}$$

Pour avoir  $R_{3(1)[11]}^6 = R_{3(2)[11]}^6$ , il faut et il suffit que  $S_5 = 0$ .

– Réciproquement, si  $S_5 = 0$ , alors :

\* si  $C_5 = 1$

$$\begin{aligned} R_{3(2)}^6 &= \begin{pmatrix} 0 & 0 & 1 \\ C_{4+\delta}C_{6-\delta} - S_{4+\delta}S_{6-\delta} & -C_{4+\delta}S_{6-\delta} - S_{4+\delta}C_{6-\delta} & 0 \\ S_{4+\delta}C_{6-\delta} + C_{4+\delta}S_{6-\delta} & -S_{4+\delta}S_{6-\delta} + C_{4+\delta}C_{6-\delta} & 0 \end{pmatrix} \\ &= \begin{pmatrix} 0 & 0 & 1 \\ C_{4+6} & -S_{4+6} & 0 \\ S_{4+6} & C_{4+6} & 0 \end{pmatrix} \\ &= R_{3(1)}^6 \end{aligned}$$

\* De même, si  $C_5 = -1$ , on retrouve  $R_{3(1)}^6 = R_{3(2)}^6$ .

En comparant les images aux positions initiale et finale, on peut régler  $q_5$  jusqu'à obtenir deux images identiques. On a alors  $q_5 = q_{05}$ . En pratique, il suffit de comparer les deux images sur seulement quatre points non-coplanaires dans la scène pour savoir si les deux images sont égales (la scène est évidemment fixe). Par exemple, les points considérés peuvent être obtenus en calculant le centre de gravité de la projection sur l'image de quatre disques.

- Remarque : on ne peut confondre les valeurs  $q_{05}$  et  $q_{05} + \pi$  car  $R_5$  doit conserver son orientation définie par la position de référence.

## 2.3 Réglage de $q_{02}$ et $q_{03}$

Ayant identifié  $q_{05}$ , une démarche analogue permet de régler les zéros des articulations 2 et 3 en amenant ces articulations dans la singularité correspondant à la configuration où le bras est entièrement étendu.

Comme précédemment, on peut démontrer :

- Pour une position initiale  $q_1, q_2, q_3, q_4, q_{05}, q_6$  donnant  $M_{0(1)}^6$  et pour une position finale  $q_1 + \delta, q_2, q_3, q_4, q_{05}, q_6 - \delta$  donnant  $M_{0(2)}^6$ , avec  $q_1, q_2, q_3, q_4, q_6$  et  $\delta (\neq 0)$  quelconques, on a :

$$M_{0(1)}^6 = M_{0(2)}^6 \text{ si et seulement si } \begin{cases} q_2 = q_{02} \text{ ou } q_2 = q_{02} + \pi \\ q_3 = q_{03} \text{ ou } q_3 = q_{03} + \pi \end{cases} \quad (2.3)$$

Il faut remarquer qu'il est impossible de dissocier le réglage des articulations 2 et 3. La méthodologie adoptée consiste à régler  $q_3$  au mieux, ce qui fournit  $\tilde{q}_3$ , puis  $q_2$  en prenant  $q_3 = \tilde{q}_3$  pour obtenir  $\tilde{q}_2$  et à réitérer le processus jusqu'à obtenir deux images identiques aux positions initiale et finale.

## 2.4 Réglage de $q_{04}$

Connaissant maintenant  $q_{02}, q_{03}$  et  $q_{05}$ , il est possible de positionner le robot tel que :

$$\begin{cases} O_6 \in (O_0, \vec{z}_0) \\ \vec{z}_0 = \vec{z}_6 \\ O_0O_6 = h < L_1 + L_2 + L_3 \end{cases} \quad (2.4)$$

Il faut (voir figure 2.2):

$$\begin{cases} L_2 \cos q'_2 + L_3 \cos q'_2 + q'_3 = h \\ L_2 \sin q'_2 + L_3 \sin q'_2 + q'_3 = 0 \\ q'_5 = -(q'_2 + q'_3) \end{cases} \quad (2.5)$$

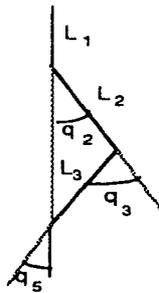


Figure 2.2 : Poignet vertical

On démontre :

- Pour une position initiale  $q_1, q_2, q_3, q_4, q_5, q_6$  donnant  $M_{0(1)}^6$  et pour une position finale  $q_1 + \delta, q_2, q_3, q_4, q_5, q_6 - \delta$  donnant  $M_{0(2)}^6$ , avec  $q_2, q_3, q_4$  et  $q_5$  vérifiant les conditions ci-dessus et  $q_1, q_2, \delta (\neq 0)$  quelconques, on a :

$$M_{0(1)}^6 = M_{0(2)}^6 \text{ si et seulement si } q_4 = q_{04} \text{ ou } q_4 = q_{04} + \pi \quad (2.6)$$

On peut ainsi régler  $q_4$  de la même manière que  $q_5$  pour obtenir  $q_{04}$ .

## 2.5 Résultats expérimentaux

Le réglage de  $q_{05}$  a donné des résultats excellents : pas de différence entre les deux images. Par contre, il a été impossible d'obtenir parfaitement  $q_{02}$  et  $q_{03}$  : des erreurs d'environ un pixel séparent les deux positions de chaque point considéré dans l'image. Le réglage de l'articulation 4 a donné des résultats aussi précis, ce qui laisse supposer une bonne mesure des longueurs des segments du robot et une bonne valeur de  $q_{04}$ . Les erreurs constatées sont certainement dues au fait que les articulations du robot ne sont pas parfaitement concourantes, par défaut de construction.

Des vérifications sur la précision des valeurs obtenues peuvent être effectuées en utilisant un transformateur de coordonnées inverse qui donne l'ensemble des solutions des positions articulaires à partir d'une matrice  $M_0^6$  voulue. En comparant les images correspondant à chaque configuration, on constate des erreurs de 3 pixels au plus séparant les positions d'un même point dans les deux images acquises. La faible valeur de ces erreurs (1 pixel  $\sim 20 \times 20 \mu m^2$ ) correspondant aux cas de figure les plus défavorables atteste la bonne précision des résultats obtenus.

- Remarque : Lors de l'exécution d'un déplacement, on s'efforcera, pour garantir une excellente précision, d'éviter ces reconfigurations : en effet, les coordonnées articulaires varient alors de manière discontinue d'une position à une autre.

Une caméra, de par sa grande précision, est un excellent moyen de calibration d'un robot et pourrait être utilisée par les constructeurs eux-mêmes pour évaluer la précision de leurs produits. Les résultats obtenus dans notre cas particulier montrent la validité de l'approche. Toutefois, une réflexion plus approfondie est en cours, visant à proposer une méthodologie et une métrologie adaptables à une large classe de manipulateurs de géométries différentes.

# Chapitre 3

## Modélisation et calibration de la caméra

La caméra est modélisée de façon très classique par une projection perspective. L'identification des paramètres de ce modèle permet d'établir la correspondance entre la position d'un point dans l'image numérisée, exprimée en pixel, et sa position réelle dans la scène.

Enfin, pour améliorer la précision de ce modèle, nous introduisons un coefficient de distorsion radiale permettant de prendre en compte les défauts éventuels de l'objectif.

Pour plus de détails sur les principales méthodes de calibration, le lecteur se reportera à l'article de Tsai [TSA 87a].

### 3.1 Mise en équations

L'origine du repère de la caméra est située au centre de projection  $C$ . L'axe  $(C, \vec{x}_c)$  est orienté comme les abscisses du plan image et l'axe  $(C, \vec{y}_c)$  comme les ordonnées. La distance minimale du centre de projection au plan image, appelée distance focale, est notée  $f$ . L'intersection de l'axe  $(C, \vec{z}_c)$  avec le plan image n'est pas supposée être au centre de l'image, mais a pour coordonnées dans l'image numérisée  $(X_c, Y_c)$  (voir figure 3.1).

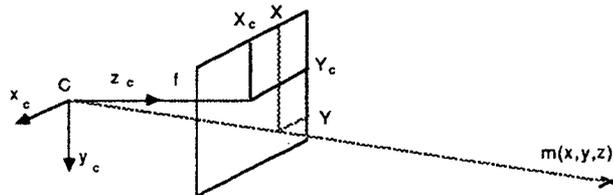


Figure 3.1 : Projection perspective

- Remarque : la seule hypothèse faite sur l'image est sa planéité et l'orthogonalité de ses axes.

Dans le repère choisi, la projection perspective d'un point  $m(x, y, z)$  sur le plan image est  $M(X, Y)$  (voir figure 3.2) avec :

$$X = f \cdot \frac{x}{z}, \quad Y = f \cdot \frac{y}{z} \quad (3.1)$$

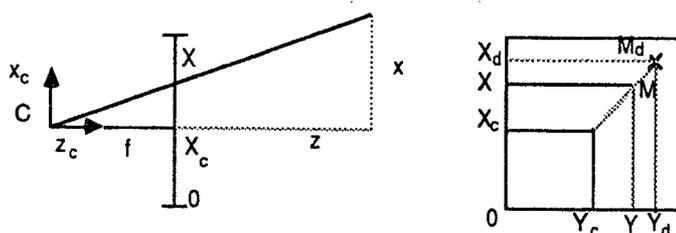


Figure 3.2 : Position d'un point dans l'image

Exprimons maintenant M par sa position  $M_p(X_p, Y_p)$  dans l'image numérisée (en pixel) :

$$X_p = X_c + \frac{X}{l_x}, \quad Y_p = Y_c + \frac{Y}{l_y} \quad (3.2)$$

où  $l_x$  et  $l_y$  sont des coefficients proportionnels à la taille d'un pixel élémentaire.

Soit  $K_d$  le coefficient de distorsion radiale [BRO 71], alors la position du point  $M_p$  réellement observé dans l'image est  $M_d(X_d, Y_d)$  avec :

$$\begin{cases} X_d = X_p + K_d \cdot (X_p - X_c) \cdot [(X_p - X_c)^2 + (Y_p - Y_c)^2] \\ Y_d = Y_p + K_d \cdot (Y_p - Y_c) \cdot [(X_p - X_c)^2 + (Y_p - Y_c)^2] \end{cases} \quad (3.3)$$

A partir de ces équations, on peut écrire le modèle complet de la caméra :

$$\begin{cases} X_d = F_x \cdot \frac{x}{z} + X_c + K_d \cdot F_x \cdot \frac{x}{z} \cdot (F_x^2 \cdot \frac{x^2}{z^2} + F_y^2 \cdot \frac{y^2}{z^2}) \\ Y_d = F_y \cdot \frac{y}{z} + Y_c + K_d \cdot F_y \cdot \frac{y}{z} \cdot (F_x^2 \cdot \frac{x^2}{z^2} + F_y^2 \cdot \frac{y^2}{z^2}) \end{cases} \quad (3.4)$$

où  $F_x = f/l_x$  et  $F_y = f/l_y$ .

Les paramètres à identifier, appelés paramètres intrinsèques de la caméra, sont alors :  $X_c, Y_c, F_x, F_y$  et  $K_d$ .

Dans la majorité des applications, il est inutile de dissocier  $f, l_x$  et  $l_y$  pour utiliser le modèle de la caméra. Si l'on a tout de même besoin explicitement de ces valeurs, fournies en général par le constructeur, il suffit d'en supposer une exacte et d'en déduire les autres.

Les équations (3.4) ne sont pas directement exploitables car l'on ne connaît pas en pratique les coordonnées  $(x, y, z)$  de  $m$  dans le repère de la caméra. Par contre, si l'on se fixe un repère objet dans lequel on connaît les coordonnées  $(x_o, y_o, z_o)$  de  $m$ , il est possible d'identifier les paramètres intrinsèques de la caméra, de même que l'orientation et la position du repère objet par rapport au repère de la caméra, soit la matrice  $M_c^o$ . Les paramètres représentant cette matrice sont appelés paramètres extrinsèques.

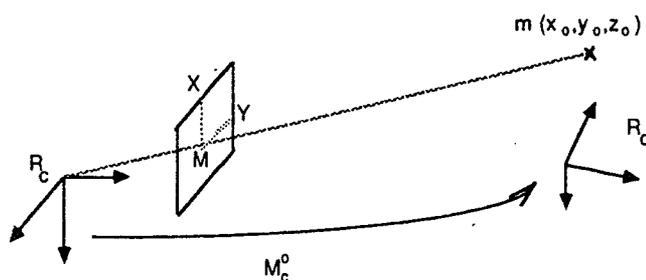


Figure 3.3 : Passage du repère caméra au repère objet

Soit  $m$  un point de coordonnées  $(x, y, z)$  dans  $R_c$  et  $(x_o, y_o, z_o)$  dans  $R_o$ , on a bien entendu (voir figure 3.3) :

$$\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = M_c^o \cdot \begin{pmatrix} x_o \\ y_o \\ z_o \\ 1 \end{pmatrix} \quad (3.5)$$

avec

$$M_c^o = \begin{pmatrix} R_c^o & T_c^o \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & X_o \\ a_{21} & a_{22} & a_{23} & Y_o \\ a_{31} & a_{32} & a_{33} & Z_o \end{pmatrix}$$

En combinant les équations (3.4) et (3.5), on obtient deux équations reliant les coordonnées d'un point dans le repère objet et les coordonnées de son image dans l'image numérisée.

## 3.2 Résolution par un système linéaire – Simulation

Dans un premier temps, nous supposons le coefficient de distorsion  $K_d$  nul. Cela va nous permettre de formuler le problème en termes d'un système linéaire. Le modèle de la caméra peut alors s'écrire sous la forme d'une matrice homogène  $P$  et on obtient :

$$X_p = \frac{X}{\omega}, \quad Y_p = \frac{Y}{\omega}, \quad f = \frac{Z}{\omega} \quad (3.6)$$

avec

$$\begin{pmatrix} X \\ Y \\ Z \\ \omega \end{pmatrix} = P \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 1/l_x & 0 & X_c/f \\ 0 & 1/l_y & Y_c/f \\ 0 & 0 & 1 \\ 0 & 0 & 1/f \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

Les équations (3.4) se réécrivent :

$$\begin{cases} X_p \cdot z - X_c \cdot z - F_x \cdot x = 0 \\ Y_p \cdot z - Y_c \cdot z - F_y \cdot y = 0 \end{cases} \quad (3.7)$$

et on en déduit, en utilisant (3.5) :

$$\begin{aligned} & a_{31}X_p x_o + a_{32}X_p y_o + a_{33}X_p z_o + X_p Z_o \\ & -a_{31}X_c x_o - a_{32}X_c y_o - a_{33}X_c z_o - X_c Z_o \\ & -F_x a_{11}x_o - F_x a_{12}y_o - F_x a_{13}z_o + F_x X_o = 0 \end{aligned} \quad (3.8)$$

$$\begin{aligned} & a_{31}Y_p x_o + a_{32}Y_p y_o + a_{33}Y_p z_o + Y_p Z_o \\ & -a_{31}Y_c x_o - a_{32}Y_c y_o - a_{33}Y_c z_o - Y_c Z_o \\ & -F_y a_{21}x_o - F_y a_{22}y_o - F_y a_{23}z_o - F_y Y_o = 0 \end{aligned}$$

On obtient le système linéaire à douze inconnues  $I = (I_1 \cdots I_{12})^T$  :

$$\begin{pmatrix} X_p x_o & X_p y_o & X_p z_o & -x_o & -y_o & -z_o & 0 & 0 & 0 & -1 & 0 & X_p \\ Y_p x_o & Y_p y_o & Y_p z_o & 0 & 0 & 0 & -x_o & -y_o & -z_o & 0 & -1 & Y_p \end{pmatrix} I = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

avec :

$$\begin{aligned} I_1 &= a_{31} & I_2 &= a_{32} & I_3 &= a_{33} \\ I_4 &= a_{31}X_c + a_{11}F_x & I_5 &= a_{32}X_c + a_{12}F_x & I_6 &= a_{33}X_c + a_{13}F_x \\ I_7 &= a_{31}Y_c + a_{21}F_y & I_8 &= a_{32}Y_c + a_{22}F_y & I_9 &= a_{33}Y_c + a_{23}F_y \\ I_{10} &= X_c Z_o + F_x X_o & I_{11} &= Y_c Z_o + F_y Y_o & I_{12} &= Z_o \end{aligned} \quad (3.9)$$

Six points non-coplanaires sont suffisants pour résoudre ce système, mais, pour améliorer la précision des résultats, il est nécessaire d'en utiliser davantage : on a alors un système surdimensionné à  $(2 \times N \text{ points})$  équations si  $N > 6$ .

Une résolution directe est impossible (solution triviale nulle), aussi utilise-t-on une contrainte liée aux propriétés d'orthonormalité des matrices de rotation, à savoir :

$$a_{31}^2 + a_{32}^2 + a_{33}^2 = 1 \quad (3.10)$$

La méthode de Lagrange permet alors de résoudre ce système en termes de minimisation sous contrainte [FAU 87].

En effet, le système peut se réécrire sous la forme  $A.X_1 + B.X_2 = 0$  en posant  $X_1 = (a_{31}, a_{32}, a_{33})^T$ . On doit alors minimiser le critère :

$$C = \|A.X_1 + B.X_2\|^2 + \lambda (1 - \|X_1\|^2) \quad (3.11)$$

où  $\lambda$  est un multiplicateur de Lagrange.

En annulant les dérivées partielles de C, on obtient :

$$\begin{cases} \frac{1}{2} \frac{\partial C}{\partial X_1} = A^T A.X_1 + A^T B.X_2 - \lambda X_1 = 0 \\ \frac{1}{2} \frac{\partial C}{\partial X_2} = B^T A.X_1 + B^T B.X_2 = 0 \end{cases} \quad (3.12)$$

d'où

$$\begin{cases} X_2 = -(B^T B)^{-1} B^T A.X_1 \\ E.X_1 = \lambda X_1 \text{ avec } E = A^T A - A^T B (B^T B)^{-1} B^T A \end{cases} \quad (3.13)$$

De plus, en utilisant (3.12), on voit que le critère C a pour valeur  $\lambda$  si  $X_1$  est un vecteur propre unitaire de E (matrice symétrique et positive) correspondant à la valeur propre  $\lambda$ .

$X_1$  est donc le vecteur propre unitaire correspondant à la plus petite valeur propre de E. On obtient ainsi  $X_1$  et  $X_2$  au signe près, et, pour remonter à I, il suffit de remarquer que  $I_{12}$  est positif (le repère objet est choisi arbitrairement devant la caméra).

Grâce aux propriétés des matrices de rotation et aux relations (3.9), on en déduit les paramètres intrinsèques et extrinsèques :

$$\begin{cases} X_c = I_1.I_4 + I_2.I_5 + I_3.I_6 & Y_c = I_1.I_7 + I_2.I_8 + I_3.I_9 \\ F_x = I_4^2 + I_5^2 + I_6^2 - X_c^2 & F_y = I_7^2 + I_8^2 + I_9^2 - Y_c^2 \\ a_{31} = I_1 & a_{32} = I_2 & a_{33} = I_3 \\ a_{11} = I_4/F_x - a_{31}X_c & a_{12} = I_5/F_x - a_{32}X_c & a_{13} = I_6/F_x - a_{33}X_c \\ a_{21} = I_7/F_y - a_{31}Y_c & a_{22} = I_8/F_y - a_{32}Y_c & a_{23} = I_9/F_y - a_{33}Y_c \\ X_0 = (I_{10} - X_c I_{12})/F_x & Y_0 = (I_{11} - Y_c I_{12})/F_y & Z_0 = I_{12} \end{cases} \quad (3.14)$$

Un programme de simulation a été développé pour tester cette méthode. Il consiste à construire les points images à partir d'un modèle donné et de points objets choisis; à brouter les points images obtenus et, enfin, à retrouver ce modèle.

Les résultats sont excellents pour des simulations exactes (bruit nul). Par contre, en présence d'un bruit blanc uniforme, même peu important, les paramètres retrouvés sont biaisés. Dans le tableau 3.1 sont regroupés les résultats (moyenne  $\bar{m}$ , écart-type  $\sigma$ , minimum et maximum) de 100 simulations pour des bruits blancs d'amplitude maximale  $\varepsilon = 0.5, 1, 1.5$  et 2 pixels. La rotation entre le repère caméra et le repère scène est représentée par le vecteur  $T = (T_x, T_y, T_z)^T$  qui s'obtient par  $T = \theta.U$  à partir de la représentation  $(U, \theta)$  de cette rotation. L'intérêt d'utiliser cette représentation sera vu ultérieurement.

$\varepsilon = 0.5$ pixel					$\varepsilon = 1.0$ pixel			
	$\bar{m}$	$\sigma$	min	max	$\bar{m}$	$\sigma$	min	max
$X_c$	121.080	0.55	119.86	122.43	121.160	0.95	118.87	123.64
$Y_c$	141.161	0.64	138.74	142.48	141.557	1.40	137.46	144.71
$F_x$	399.545	0.63	398.01	401.12	397.950	1.36	394.86	401.29
$F_y$	499.464	0.77	497.71	501.58	497.435	1.71	493.04	502.68
$X_o$	-100.100	0.68	-101.71	-98.64	-100.242	1.21	-103.33	-97.04
$Y_o$	-120.180	0.64	-121.52	-117.76	-120.629	1.41	-123.93	-116.52
$Z_o$	499.282	0.95	496.85	501.81	496.821	2.14	492.09	502.60
$T_x$	14.917	0.07	14.64	15.09	14.929	0.15	14.47	15.29
$T_y$	16.906	0.07	16.73	17.11	16.910	0.13	16.56	17.19
$T_z$	31.367	0.02	31.30	31.41	31.365	0.04	31.28	31.49
$\varepsilon = 1.5$ pixel					$\varepsilon = 2.0$ pixel			
	$\bar{m}$	$\sigma$	min	max	$\bar{m}$	$\sigma$	min	max
$X_c$	121.061	1.80	116.72	124.93	121.694	1.89	116.74	126.86
$Y_c$	142.395	2.09	137.60	148.92	143.657	2.69	136.70	149.47
$F_x$	395.222	2.01	390.81	400.58	391.779	2.64	385.39	400.06
$F_y$	494.141	2.53	487.70	501.23	489.953	3.24	481.56	498.95
$X_o$	-100.117	2.19	-104.62	-94.61	-101.042	2.32	-107.38	-94.60
$Y_o$	-121.568	2.11	-128.07	-116.80	-122.946	2.72	-128.47	-115.76
$Z_o$	492.747	3.22	486.11	499.92	487.276	4.10	477.79	499.09
$T_x$	14.981	0.26	14.31	15.79	15.042	0.31	14.32	15.84
$T_y$	16.902	0.22	16.46	17.43	16.841	0.26	16.31	17.53
$T_z$	31.374	0.05	31.22	31.51	31.382	0.08	31.19	31.58

Tableau 3.1 : Simulation de la méthode de Lagrange

- Remarque : Pour les différentes simulations, les paramètres intrinsèques et extrinsèques du modèle à identifier sont :

$$\begin{aligned}
 X_c &= 121 \text{ pixels} & Y_c &= 141 \text{ pixels} \\
 F_x &= 400 & F_y &= 500 \\
 K_d &= 0 \\
 X_o &= -100 \text{ mm} & Y_o &= -120 \text{ mm} & Z_o &= 500 \text{ mm} \\
 T_x &= 14.912 & T_y &= 16.920 & T_z &= 31.362
 \end{aligned}$$

En comparant les valeurs de ce modèle avec les paramètres calculés par la méthode décrite ci-dessus, on constate un biais important sur les valeurs  $F_x$ ,  $F_y$  et  $Z_o$ , bien que le coefficient de distorsion soit choisi nul dans le modèle de référence.

Dans le cas de l'identification d'une caméra réelle, les imperfections des optiques et du positionnement de l'élément photosensible dans la caméra entraînent inévitablement des erreurs sur la mesure des coordonnées-image. Aussi ne peut-on se satisfaire de cette méthode donnant des résultats biaisés en présence de bruit. Toutefois, ces résultats peuvent servir d'estimées initiales à des méthodes plus performantes.

### 3.3 Résolution par une méthode non linéaire – Simulation

Parmi les méthodes d'estimations classiquement utilisées, certains auteurs ont fait appel, pour résoudre ce problème, à une méthode de filtrage étendu de Kalman. L'inconvénient de cette technique réside dans le fait qu'elle nécessite la linéarisation du modèle et qu'elle ne permet pas d'estimer le coefficient de distorsion radiale. Nous avons choisi, plus naturellement, une méthode de minimisation non linéaire [SOB 74].

Le critère à minimiser est :

$$C_2 = \sum_{i=1}^N [(X_{di} - X_{mi})^2 + (Y_{di} - Y_{mi})^2] \quad (3.15)$$

où  $(X_{mi}, Y_{mi})$  est la mesure du point  $i$  dans l'image numérisée et où  $(X_{di}, Y_{di})$  représente son estimée par le modèle calculé à partir des équations (3.4) et (3.5), fonction des paramètres intrinsèques et extrinsèques de la caméra.

Cette méthode permet en outre d'identifier le coefficient de distorsion radiale  $K_d$  dont l'estimée initiale est évidemment choisie nulle.

La rotation est représentée par le vecteur  $T = \theta.U$  pour avoir une représentation minimale des rotations, condition nécessaire pour employer une méthode de minimisation non linéaire sans contrainte classique (dans notre cas, une méthode de type Gauss-Newton adaptée aux fonctions composées de sommes de carrés [NAG 84]).

Cette méthode a été testée en simulation et nous avons constaté qu'elle donnait des résultats tout à fait satisfaisants et sans biais (voir tableau 3.2, résultats correspondant au même modèle que précédemment).

	$\epsilon = 0.5$ pixel				$\epsilon = 1.0$ pixel			
	$\bar{m}$	$\sigma$	min	max	$\bar{m}$	$\sigma$	min	max
$X_c$	121.016	0.49	119.96	122.43	121.016	0.85	119.06	123.13
$Y_c$	140.998	0.62	138.61	142.30	140.927	1.35	136.86	144.03
$F_x$	400.095	0.61	398.55	401.58	400.063	1.26	396.93	402.93
$F_y$	500.135	0.76	498.39	502.12	500.037	1.58	495.99	504.57
$K_d \cdot 10^6$	-0.001	0.02	-0.06	0.06	-0.003	0.05	-0.16	0.13
$X_o$	-100.006	0.58	-101.69	-98.75	-100.030	1.02	-102.53	-97.55
$Y_o$	-119.995	0.63	-121.33	-117.59	-119.927	1.36	-123.14	-115.82
$Z_o$	500.126	0.92	497.71	502.50	500.080	1.99	495.80	505.12
$T_x$	14.911	0.07	14.64	15.07	14.899	0.15	14.46	15.25
$T_y$	16.913	0.06	16.75	17.06	16.924	0.12	16.54	17.23
$T_z$	31.365	0.02	31.30	31.41	31.360	0.03	31.26	31.47
	$\epsilon = 1.5$ pixel				$\epsilon = 2.0$ pixel			
	$\bar{m}$	$\sigma$	min	max	$\bar{m}$	$\sigma$	min	max
$X_c$	120.723	1.73	116.08	125.07	121.139	1.86	116.43	125.31
$Y_c$	140.967	2.03	136.12	146.88	140.994	2.68	134.70	147.64
$F_x$	399.901	2.03	395.18	405.14	400.135	2.67	394.57	409.47
$F_y$	499.907	2.56	493.96	506.95	500.232	3.32	492.89	511.12
$K_d \cdot 10^6$	0.002	0.07	-0.17	0.21	0.013	0.10	-0.18	0.29
$X_o$	-99.628	2.06	-104.84	-93.96	-100.197	2.24	-105.52	-94.18
$Y_o$	-119.970	2.08	-125.87	-115.13	-119.977	2.72	-126.45	-113.40
$Z_o$	499.978	3.29	492.85	507.92	500.201	4.13	490.31	514.32
$T_x$	14.917	0.24	14.27	15.65	14.914	0.31	14.28	15.75
$T_y$	16.940	0.21	16.46	17.50	16.907	0.24	16.44	17.56
$T_z$	31.360	0.05	31.20	31.50	31.360	0.08	31.16	31.55

Tableau 3.2 : Simulation de la méthode non linéaire

### 3.4 Résultats expérimentaux

Pour l'expérimentation en site réel, nous avons choisi les mêmes points objets qu'en simulation, à savoir une grille rectangulaire de  $6 \times 8$  disques que l'on peut déplacer très précisément (au centième de millimètre) par rapport à l'axe  $\vec{z}_o$  du repère objet (voir figure 3.4). Cinq positions équidistantes ont été choisies sur un intervalle de 20 cm, de manière à avoir un ensemble de 240 points de mesure lorsque tous les points sont visibles par la caméra.

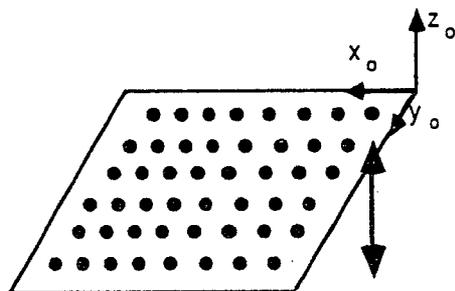


Figure 3.4 : Mire utilisée

Les points objets sont situés au centre de ces disques et la position des points images correspondants est obtenue en calculant le centre de gravité de leur projection sur le plan image (voir figure 3.5).

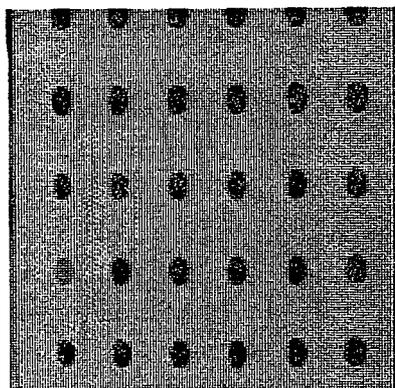


Figure 3.5 : Caméra 1

On obtient les résultats suivants :

$$\begin{aligned} X_c &= 120 \text{ pixels} & Y_c &= 153 \text{ pixels} \\ F_x &= 387 & F_y &= 578 \\ K_d &= -4 e - 7 \end{aligned}$$

- Remarques :

1. La valeur du critère  $C_2$  après minimisation (30.5) correspond à une erreur moyenne de l'ordre du quart de pixel entre chaque point mesuré et sa reconstruction par le modèle. ce qui confirme la validité du choix du modèle.
2. D'après les équations (3.3), la distorsion est d'environ 1 pixel pour les points situés à une distance de 150 pixels de  $(X_c, Y_c)$ , c'est à dire qu'elle est quasiment négligeable, comme on pouvait l'espérer en observant les images de la mire (voir figure 3.5) non ou infiniment peu déformées.
3. D'après les données fournies par le constructeur, la matrice de la caméra est constituée de 208 lignes et 288 colonnes, la taille d'un pixel élémentaire est de  $28 \times 15 \mu m^2$  et la distance focale est de 9 millimètres. De plus, les images numérisées sont de taille  $256 \times 256$  pixels. On en déduit :

$$F_x \sim \frac{9}{28 \cdot 10^{-3}} \cdot \frac{256}{208} = 396 \quad \text{et} \quad F_y \sim \frac{9}{15 \cdot 10^{-3}} = 600$$

Ces valeurs sont bien du même ordre de grandeur que celle obtenues par identification du modèle de la caméra. La différence importante entre ces valeurs et celles obtenues par identification confirme la faible fiabilité des données fournies par un constructeur.

4. Vu la taille des images et les valeurs de  $X_c$  et  $Y_c$  obtenues, on constate un décalage important entre le centre de l'image et la projection  $(X_c, Y_c)$  du centre optique sur le plan image. L'hypothèse de décentrage est donc essentielle pour obtenir un modèle correct.

Les mêmes expérimentations ont été effectuées sur une autre caméra où l'on observe une distorsion importante (voir figure 3.6).

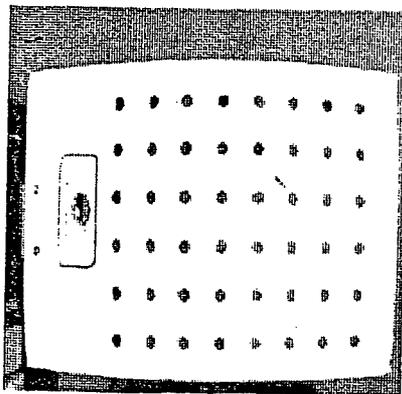


Figure 3.6 : Caméra 2

On a obtenu les résultats suivants :

$$\begin{aligned} X_c &= 151 \text{ pixels} & Y_c &= 146 \text{ pixels} \\ F_x &= 208 & F_y &= 296 \\ K_d &= -3.5 e - 6 \end{aligned}$$

Comme prévu, le coefficient de distorsion radiale est beaucoup plus important pour cette caméra. Ici encore, l'erreur moyenne entre les points mesurés et les points reconstruits est de l'ordre du quart de pixel pour la méthode non linéaire. Si l'on n'utilise que la méthode de Lagrange, l'erreur moyenne vaut environ trois quarts de pixel .

### 3.5 Simplifications

Une fois les paramètres intrinsèques déterminés, on peut effectuer la même démarche pour identifier uniquement la matrice  $M_c^o$ . Les simplifications donnent le nouveau système :

$$A.X_1 + B.X_2 = 0 \tag{3.16}$$

avec

$$X_1 = (a_{31}, a_{32}, a_{33})^T$$

$$X_2 = (a_{11}, a_{12}, a_{13}, a_{21}, a_{22}, a_{23}, X_o, Y_o, Z_o)^T$$

$$A = \begin{pmatrix} (X_p - X_c)x_o & (X_p - X_c)y_o & (X_p - X_c)z_o \\ (Y_p - Y_c)x_o & (Y_p - Y_c)y_o & (Y_p - Y_c)z_o \end{pmatrix}$$

$$B = \begin{pmatrix} -F_x x_o & -F_x y_o & -F_x z_o & 0 & 0 & 0 & -F_x & 0 & X_p - X_c \\ 0 & 0 & 0 & -F_y x_o & -F_y y_o & -F_y z_o & 0 & -F_y & Y_p - Y_c \end{pmatrix}$$

La résolution de ce système par la méthode de Lagrange, en utilisant la même contrainte sur  $X_1$ , donne également des résultats biaisés. Aussi emploie-t-on la même méthode de minimisation non linéaire avec seulement les six paramètres extrinsèques  $X_o, Y_o, Z_o, T_x, T_y$  et  $T_z$  à identifier (voir tableau 3.3).

On peut vérifier lors des expérimentations réelles la précision des résultats obtenus en comparant les positions des points dans l'image numérisée et les positions des points reconstruits par le modèle de la caméra. On constate que quelques points ont parfois une erreur supérieure à 1 pixel : cela est du à des erreurs de mesure mais surtout aux défauts inévitables de l'objectif.

En moyenne, l'erreur est de l'ordre du quart de pixel, ce qui permet d'affirmer que les paramètres du modèle de la caméra ont été correctement identifiés.

méthode de Lagrange					méthode non linéaire			
$\epsilon = 0.5$ pixel								
	$\bar{m}$	$\sigma$	min	max	$\bar{m}$	$\sigma$	min	max
$X_o$	-99.858	0.17	-100.44	-99.35	-99.997	0.07	-100.15	-99.81
$Y_o$	-119.848	0.18	-120.49	-119.44	-119.989	0.04	-120.09	-119.89
$Z_o$	499.122	0.97	496.98	502.63	499.980	0.13	499.61	500.44
$T_x$	14.926	0.05	14.79	15.04	14.915	0.01	14.85	14.95
$T_y$	16.921	0.04	16.80	17.02	16.915	0.01	16.86	16.94
$T_z$	31.362	0.01	31.33	31.39	31.364	0.01	31.33	31.38
$\epsilon = 1.0$ pixel								
	$\bar{m}$	$\sigma$	min	max	$\bar{m}$	$\sigma$	min	max
$X_o$	-99.514	0.34	-100.70	-98.81	-99.993	0.11	-100.21	-99.66
$Y_o$	-119.467	0.37	-120.68	-118.52	-119.998	0.09	-120.23	-119.71
$Z_o$	496.798	1.91	492.25	503.03	500.002	0.23	499.24	500.69
$T_x$	14.952	0.09	14.76	15.16	14.911	0.03	14.80	14.99
$T_y$	16.942	0.09	16.68	17.17	16.918	0.04	16.80	16.99
$T_z$	31.355	0.02	31.30	31.41	31.363	0.01	31.31	31.41
$\epsilon = 1.5$ pixel								
	$\bar{m}$	$\sigma$	min	max	$\bar{m}$	$\sigma$	min	max
$X_o$	-98.874	0.57	-100.80	-97.25	-99.983	0.22	-100.63	-99.40
$Y_o$	-118.815	0.56	-120.50	-117.21	-120.010	0.14	-120.39	-119.71
$Z_o$	492.869	3.35	481.44	501.86	499.965	0.39	499.08	500.85
$T_x$	15.016	0.15	14.64	15.36	14.910	0.04	14.78	15.06
$T_y$	16.981	0.12	16.68	17.31	16.915	0.06	16.72	17.07
$T_z$	31.346	0.04	31.25	31.42	31.364	0.04	31.27	31.45
$\epsilon = 2.0$ pixel								
	$\bar{m}$	$\sigma$	min	max	$\bar{m}$	$\sigma$	min	max
$X_o$	-97.957	0.67	-99.71	-96.37	-99.962	0.26	-100.61	-99.23
$Y_o$	-117.813	0.69	-119.37	-116.19	-120.018	0.18	-120.50	-119.47
$Z_o$	486.788	3.80	478.17	497.09	500.025	0.48	498.92	501.39
$T_x$	15.087	0.21	14.22	15.63	14.905	0.07	14.68	15.08
$T_y$	17.012	0.13	16.76	17.43	16.908	0.08	16.68	17.12
$T_z$	31.328	0.05	31.16	31.45	31.366	0.04	31.24	31.50

Tableau 3.3 : Simulation des deux méthodes de calibration

# Chapitre 4

## Identification de la matrice de passage entre le poignet du robot et la caméra

### 4.1 Mise en équations

Dans le premier chapitre de ce rapport, nous avons montré qu'il est possible de régler précisément les zéros angulaires du robot afin de connaître la position et l'orientation du poignet par rapport à un repère de référence, fixe.

Nous allons maintenant identifier la matrice  $M_p^c$ , dernier maillon de la chaîne cinématique du système robot-caméra, représentant la position et l'orientation de la caméra (et plus précisément de son centre optique) par rapport au poignet.

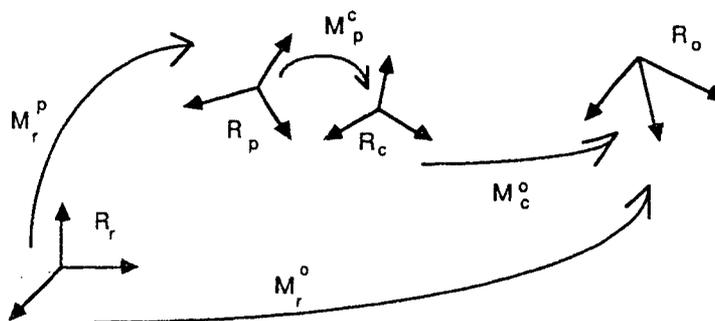


Figure 4.1 :

On a (voir figure 4.1)

$$M_r^o = M_r^p \cdot M_p^c \cdot M_c^o \quad (4.1)$$

ce qui donne :

$$M_p^c = M_r^{p-1} \cdot M_r^o \cdot M_c^{o-1} \quad (4.2)$$

Si  $M_r^p$  et  $M_c^o$  peuvent être connues ( $M_r^p$  en utilisant le transformateur de coordonnées,  $M_c^o$  par calibration), il est, par contre, impossible de connaître précisément la matrice  $M_r^o$  bien que celle-ci soit choisie constante. Aussi allons-nous utiliser des mouvements du poignet, et donc de la caméra, pour déterminer  $M_p^c$ .

Pour des positions distinctes  $i$  et  $j$ , on a :

$$M_r^o = M_{r(i)}^p \cdot M_p^c \cdot M_{c(i)}^o = M_{r(j)}^p \cdot M_p^c \cdot M_{c(j)}^o \quad (4.3)$$

donc

$$M_{r(j)}^{p-1} \cdot M_{r(i)}^p \cdot M_p^c = M_p^c \cdot M_{c(j)}^o \cdot M_{c(i)}^{o-1} \quad (4.4)$$

En décomposant ces matrices homogènes en leur partie rotation et translation, on obtient :

$$R_{r(j)}^{p-1} \cdot R_{r(i)}^p \cdot R_p^c = R_p^c \cdot R_{c(j)}^o \cdot R_{c(i)}^{o-1} \quad (4.5)$$

$$(R_{r(j)}^{p-1} \cdot R_{r(i)}^p - I_3) \cdot T_p^c = R_p^c \cdot (T_{c(j)}^o - R_{c(j)}^o \cdot R_{c(i)}^{o-1} \cdot T_{c(i)}^o) - R_{r(j)}^{p-1} \cdot (T_{r(j)}^p - T_{r(i)}^p) \quad (4.6)$$

Notons :

$$\begin{aligned} R_{r(ij)}^p &= R_{r(j)}^{p-1} \cdot R_{r(i)}^p & T_{r(ij)}^p &= R_{r(j)}^{p-1} \cdot (T_{r(j)}^p - T_{r(i)}^p) \\ R_{c(ij)}^o &= R_{c(j)}^o \cdot R_{c(i)}^{o-1} & T_{c(ij)}^o &= T_{c(j)}^o - R_{c(j)}^o \cdot R_{c(i)}^{o-1} \cdot T_{c(i)}^o \end{aligned}$$

on a alors :

$$R_{r(ij)}^p \cdot R_p^c = R_p^c \cdot R_{c(ij)}^o \quad (4.7)$$

et

$$(R_{r(ij)}^p - I_3) \cdot T_p^c = R_p^c \cdot T_{c(ij)}^o - T_{r(ij)}^p \quad (4.8)$$

D'après (4.8), le calcul de  $T_p^c$  ne peut évidemment s'effectuer qu'après celui de  $R_p^c$ .

Par ailleurs, deux positions du robot sont insuffisantes pour résoudre les systèmes (4.7) et (4.8) [CHA 86] (pour (4.8) par exemple, on sait que  $R - I_3$  est de rang 2 quel que soit  $R \neq I_3$ ). Une troisième position du poignet, au moins, est donc nécessaire pour pouvoir résoudre ces systèmes.

## 4.2 Résolution – Discussion

La résolution du système (4.7) est loin d'être triviale [GAN 60] mais les propriétés des matrices de rotation permettent d'aboutir [SHI 87].

- Soit  $P = 2 \sin(\frac{\theta}{2}) \cdot U$  et  $P' = 2 \tan \frac{\theta}{2}$  deux représentations des rotations obtenues à partir de  $(U, \theta)$ , alors, la résolution du système (4.7) est équivalente à celle du système linéaire suivant [TSA 87b]:

$$As(P_{r(ij)}^p + P_{c(ij)}^o) \cdot P_p'^c = P_{c(ij)}^o - P_{r(ij)}^p \quad (4.9)$$

Comme  $As(U)$  est de rang 2, quelque soit  $U$ , il faut au moins une troisième orientation  $k$  du poignet (telle que deux des axes  $P_{r(ij)}^p, P_{r(ik)}^p$  et  $P_{r(kj)}^p$  au moins aient des directions différentes) pour assurer un système de rang plein.

De plus, pour augmenter la précision dans la résolution du système (4.8) donnant  $T_p^c$ , nous nous fixons  $T_{r(ij)}^p = 0$ , c'est à dire une position du poignet constante en translation : seules des rotations du poignet seront effectuées.

Enfin, pour atténuer les effets des erreurs de mesure, plus de trois orientations du poignet seront prises en compte pour résoudre, par la méthode des moindres carrés, les systèmes (4.7), puis (4.8).

On obtient les résultats suivants :

$$P_p'^c = \begin{pmatrix} -0.0622 \\ -0.0283 \\ 0.0011 \end{pmatrix} (\Rightarrow \theta = 1.98^\circ) \Rightarrow R_p^c = \begin{pmatrix} 0.9999 & -0.0003 & -0.0143 \\ 0.0008 & 0.9995 & 0.0314 \\ 0.0143 & -0.0314 & 0.9994 \end{pmatrix}$$

$$T_p^c = \begin{pmatrix} 0 \text{ mm} \\ -8 \text{ mm} \\ 189 \text{ mm} \end{pmatrix}$$

Plusieurs expérimentations ont été effectuées, donnant les mêmes résultats au millimètre près pour la translation et au dixième de degré près pour les angles d'Euler de la rotation.

- Remarque : il existe une autre méthode, théoriquement plus simple, pour calculer la rotation  $R_p^c$  entre le poignet et la caméra : supposons l'orientation du poignet constante, on a alors  $R_{r(ij)}^p = R_{c(ij)}^o = I_3$  et l'équation (4.8) se réécrit :

$$R_p^c \cdot (T_{c(j)}^o - T_{c(i)}^o) = T_{r(j)}^p - T_{r(i)}^p$$

soit

$$R_p^c \cdot \Delta T_c^o = \Delta T_r^p$$

Ainsi deux translations perpendiculaires du poignet permettent-elles de déterminer la matrice  $R_p^c$  : par exemple, la première colonne est donnée par  $\Delta T_c^o$  correspondant à  $\Delta T_r^p = (1 \ 0 \ 0)^T$ , la seconde colonne par  $\Delta T_c^o$  correspondant à  $\Delta T_r^p = (0 \ 1 \ 0)^T$  et la troisième par produit vectoriel des deux premières.

En pratique, cette méthode s'avère beaucoup moins précise que la précédente : elle ne permet aucune correction des erreurs de mesure, d'autant plus que les déplacements du robot sont moins précis en translation qu'en rotation. En effet, les positions de référence des axes 2 et 3 (qui donnent la translation du poignet) n'ont pu être obtenues parfaitement, comme on l'a vu dans la première partie de ce rapport.

# Chapitre 5

## Validation et Conclusion

### 5.1 Validation des résultats

La transformation repère de l'organe terminal- repère de la caméra a été rajoutée dans le transformateur de coordonnées afin de connaître la position et l'orientation du centre optique de la caméra par rapport à notre repère de référence en fonction des coordonnées articulaires du robot :  $(M_r^c = M_r^p \cdot M_p^c)$ .

Pour vérifier les résultats obtenus, nous avons considéré plusieurs positions de la caméra et effectué à chaque position correspondante une calibration donnant  $M_c^o$ , position et orientation du repère caméra au repère objet; on doit vérifier :

$$M_r^o = M_{r(i)}^c \cdot M_{c(i)}^o = M_{r(j)}^c \cdot M_{c(j)}^o = \dots \quad (5.1)$$

Nous obtenons (les translations sont données en millimètres):

$$M_{r(1)}^c = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1000 \end{pmatrix}, M_{c(1)}^o = \begin{pmatrix} 0.9992 & -0.0405 & -0.0043 & -156 \\ 0.0404 & 0.9986 & -0.0335 & -166 \\ 0.0057 & 0.0333 & 0.9994 & 609 \end{pmatrix}$$

$$\Rightarrow M_r^o = \begin{pmatrix} 0.9992 & -0.0404 & -0.0044 & -156 \\ 0.0403 & 0.9986 & -0.0335 & -166 \\ 0.0057 & 0.0333 & 0.9994 & 1609 \end{pmatrix}$$

$$M_{r(2)}^c = \begin{pmatrix} 1 & 0 & 0 & 50 \\ 0 & 1 & 0 & 50 \\ 0 & 0 & 1 & 1050 \end{pmatrix}, M_{c(2)}^o = \begin{pmatrix} 0.9990 & -0.0443 & -0.0023 & -204 \\ 0.0442 & 0.9985 & -0.0323 & -214 \\ 0.0037 & 0.0321 & 0.9995 & 559 \end{pmatrix}$$

$$\Rightarrow M_r^o = \begin{pmatrix} 0.9990 & -0.0443 & -0.0022 & -155 \\ 0.0442 & 0.9985 & -0.0322 & -165 \\ 0.0036 & 0.0320 & 0.9995 & 1609 \end{pmatrix}$$

$$M_{r(3)}^c = \begin{pmatrix} 0.9924 & 0.0869 & -0.0871 & 50 \\ -0.0944 & 0.9917 & -0.0867 & 50 \\ 0.0789 & 0.0943 & 0.9924 & 1050 \end{pmatrix}, M_{c(3)}^o = \begin{pmatrix} 0.9881 & -0.1330 & 0.0775 & -140 \\ 0.1282 & 0.9897 & 0.0640 & -177 \\ -0.852 & -0.0533 & 0.9949 & 592 \end{pmatrix}$$

;

$$\Rightarrow M_r^o = \begin{pmatrix} 0.9991 & -0.0414 & -0.0043 & -155 \\ 0.0412 & 0.9987 & -0.0302 & -163 \\ 0.0055 & 0.0300 & 0.9995 & 1609 \end{pmatrix}$$

La quasi-constance de la matrice  $M_r^o$  confirme la validité des méthodes employées, aussi bien l'identification du robot que la modélisation de la caméra et l'identification de la liaison entre la caméra et le poignet du robot.

## 5.2 Conclusion

L'exploitation dans une tâche robotique d'informations fournies par des capteurs extéroceptifs embarqués sur le robot nécessite d'une part l'identification des paramètres *intrinsèques* à ces capteurs (par exemple, focale et position du centre de projection pour des caméras), d'autre part la calibration des paramètres *extrinsèques* (localisation du repère capteur vis-à-vis du robot). Ces problèmes de calibration sont souvent mal conditionnés et nécessitent des méthodes d'identification robustes.

Concernant la calibration de la caméra (paramètres intrinsèques et identification du passage repère image - repère objet), notre étude a permis de déterminer que la résolution du problème par la méthode de Lagrange donnait des résultats biaisés pour des mesures bruitées dans l'image. Ces erreurs de mesure sont inévitables en pratique, aussi avons-nous mis en oeuvre une méthode non linéaire robuste permettant en outre d'identifier un modèle plus réaliste de la caméra prenant en compte la distorsion radiale. Connaissant ce modèle, il est alors possible d'extraire d'une image des informations exploitables dans une tâche robotique référencée vision.

Enfin, l'identification de la position et de l'orientation de la caméra par rapport au robot permet de connaître les différents mouvements effectués entre toute acquisition d'images. On accède ainsi à des informations de type mouvement dans l'image qui sont utilisables aussi bien pour la reconstruction 3D d'une scène quelconque que pour l'élaboration de lois de commande basés sur la vision. Au cours de cette identification, nous avons été confrontés au problème de l'exactitude des paramètres du modèle géométrique du robot lui-même, notamment de la précision des zéros articulaires utilisés dans le transformateur de coordonnées. Ces paramètres ont pu être identifiés avec succès grâce à l'utilisation de la caméra en tant qu'instrument de mesure. Cette démarche originale nous a permis d'engager une réflexion plus générale sur le problème de la calibration des robots à partir de systèmes optiques embarqués. Une méthodologie plus générale que celle évoquée dans ce rapport, est en cours de développement.

# Références

- [BRO 71] D.C. Brown : Close Range Camera Calibration.  
*Photogrammetric Engineering*. vol. 37, no.8, pp. 855-866, 1971.
- [BRU 87] J.- F. Brunet : Identification d'un système de vision monté sur l'organe terminal d'un manipulateur.  
*Rapport de Stage*. IRISA, Rennes, Juin 1987.
- [CHA 86] F. Chaumette : Couplage et Calibrage d'une caméra.  
*Rapport de Stage*. IRISA, Rennes, Septembre 1986.
- [FAU 87] O.D. Faugeras, G. Toscani : Camera Calibration for 3D Computer Vision.  
*Proceedings of International Workshop on Machine Vision and Machine Intelligence*. Tokyo, February 1987.
- [GAN 60] F.R. Gantmacher : Matrix Theory, vol. 1.  
*Chelsea Publishing Company*. New York, 1960.
- [NAG 84] Numerical Algorithms Group : Fortran Library Manual, vol. 3.  
*Mayfield House*. Oxford, 1984.
- [SHI 87] Y.C. Shiu, S. Ahmad : Finding the Mounting Position of a Sensor by Solving a Homogeneous Transform Equation  $AX=XB$ .  
*Proceedings of IEEE International Conference on Robotics and Automation*. Raleigh N.C., pp. 1666-1671, April 1987.
- [SOB 74] I. Sobel : On Calibrating Computer Controlled Cameras for Perceiving 3D Scenes.  
*Artificial Intelligence*. vol. 5, pp. 185-198, 1974.
- [TSA 87a] R.Y. Tsai : A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology using Off-the-Shelf TV Cameras and Lenses.  
*IEEE Journal of Robotics and Automation*. vol. RA-3, no.4, pp. 323-344, 1987.
- [TSA 87b] R.Y. Tsai, R. Lenz : A New Technique for Autonomous and Efficient 3D Robotics Hand-Eye Calibration.  
*Research Report RC13212 IBM T.J. Watson Research Center*. Yorktown Heights, NY 10598, October 1987.

## LISTE DES DERNIERES PUBLICATIONS INTERNES

- PI 449 **CONVOLUTION SYSTOLIQUE DE FONCTIONS ARITHMETIQUES**  
Patrice QUINTON, Yves ROBERT  
30 Pages, Janvier 1989.
- PI 450 **MISE EN OEUVRE D'ALGORITHMES NUMERIQUES SUR UN  
HYPERCUBE**  
Brigitte VITAL  
28 Pages, Janvier 1989.
- PI 451 **LANCER DE RAYON SUR DES ARCHITECTURES PARALLELES :  
UNE ETUDE DE PERFORMANCE**  
Thierry PRIOL, Kadi BOUATOUCH  
20 Pages, Janvier 1989.
- PI 452 **LANCER DE RAYON : APPROCHES PARALLELES**  
Didier BADOUEL, François BODIN, Thierry PRIOL  
16 Pages, Janvier 1989.
- PI 453 **UN COMPILATEUR ESTELLE MULTI-PROCESSEURS POUR L'EX-  
PERIMENTATION D'ALGORITHMES DISTRIBUES SUR MACHINES  
PARALLELES**  
Jean-Marc JEZEQUEL, Claude JARD  
54 Pages, Janvier 1989.
- PI 454 **REALISATION ET CALIBRATION D'UN SYSTEME EXPERIMENTAL  
DE VISION COMPOSE D'UNE CAMERA MOBILE EMBARQUEE SUR  
UN ROBOT-MANIPULATEUR**  
François CHAUMETTE, Patrick RIVES  
36 Pages, Février 1989.
- PI 455 **ARCHITECTURE SYSTOLIQUE POUR LA CORRECTION  
AUTOMATIQUE DE LIBELLE D'ADRESSE**  
Dominique LAVENIER, Jean-Luc SCHARBARG, Patrice FRISON  
22 Pages, Février 1989.

