



**HAL**  
open science

# Comment calibrer extrinsèquement des caméras à champs non-recouvrants? Application pour un robot mobile

Pierre Lébraly, Omar Ait Aider, Eric Royer, Michel Dhome

## ► To cite this version:

Pierre Lébraly, Omar Ait Aider, Eric Royer, Michel Dhome. Comment calibrer extrinsèquement des caméras à champs non-recouvrants? Application pour un robot mobile. ORASIS - Congrès des jeunes chercheurs en vision par ordinateur, INRIA Grenoble Rhône-Alpes, Jun 2011, Praz-sur-Arly, France. inria-00595746

**HAL Id: inria-00595746**

**<https://inria.hal.science/inria-00595746>**

Submitted on 25 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.

# Comment calibrer extrinsèquement des caméras à champs non-recouvrants ? Application pour un robot mobile

Pierre Lébraly

Omar Ait-Aider

Eric Royer

Michel Dhome

Clermont Université, Université Blaise Pascal, LASMEA, BP 10448, F-63000 CLERMONT-FERRAND  
CNRS, UMR 6602, LASMEA, F-63177 AUBIERE

prénom.nom@lasmea.univ-bpclermont.fr

## Résumé

*Les systèmes multi-caméras sont de plus en plus utilisés en robotique mobile. Il est souvent nécessaire que l'étalonnage extrinsèque (poses relatives des caméras) soit précis. Pour cela, on utilise généralement des appariements entre différentes vues, ce qui est impossible à réaliser si les champs de vue des caméras sont disjoints.*

*Dans cet article, nous exposons une méthode simple et flexible pour étalonner extrinsèquement un système multi-caméras dont les champs de vue sont disjoints. Le but est de calibrer des caméras embarquées sur un véhicule pour des applications de navigation en milieu urbain. Les caméras observent donc des régions différentes à un instant donné. La procédure d'étalonnage consiste à manœuvrer le véhicule pendant que chaque caméra, intrinsèquement calibrée au préalable, observe une scène statique. Les principales contributions sont l'étude des mouvements singuliers, et un ajustement de faisceaux spécifique qui affine les scènes, les poses du système multi-caméras, et calibre extrinsèquement les caméras. Nous étudions comment traiter les mouvements singuliers, comme les mouvements plans. La méthode proposée est validée avec des données synthétiques et réelles. Traduction depuis l'anglais de l'article [19].*

## Mots Clef

Étalonnage, multi-caméras, champs non-recouvrants, mouvements singuliers, ajustement de faisceaux, robot mobile.

## Abstract

*Multi-camera systems are more and more used in vision-based robotics. An accurate extrinsic calibration (camera relative poses) is usually required. In most of cases, this task is done by matching features through different views of the same scene. However, if the camera fields of view do not overlap, such a matching procedure is not feasible anymore.*

*This article deals with a simple and flexible extrinsic calibration method, for non-overlapping camera rig. The aim is the calibration of non-overlapping cameras embedded on a vehicle, for visual navigation purpose in urban envi-*

*ronment. The cameras do not see the same area at the same time. The calibration procedure consists in manoeuvring the vehicle while each camera observes a static scene. Previously, the camera were intrinsically calibrated. The main contributions are a study of the singular motions and a specific bundle adjustment which both reconstructs the scene and calibrates the cameras. Solutions to handle the singular configurations, such as planar motions, are exposed. The proposed approach has been validated with synthetic and real data. This article is translated from [19].*

## Keywords

Calibration, camera rig, non-overlapping fields of view, singular motion, bundle adjustment, mobile robot.

## 1 Introduction

De nombreuses méthodes de navigation monoculaire utilisant des amers naturels ont été proposées récemment. Mais l'utilisation d'une seule caméra n'est pas suffisamment robuste aux conditions d'illumination extérieures; notamment aux problèmes de surexposition. Afin de résoudre ce problème, il est possible d'embarquer plusieurs caméras sur le véhicule : par exemple, l'une dirigée vers l'avant et une autre caméra dirigée vers l'arrière du véhicule. De cette manière, si l'une des caméras est éblouie par le soleil, une autre caméra pourra fournir des données pertinentes. Un autre avantage d'un système multi-caméras est qu'il fournit des informations redondantes, et rend donc la perception plus robuste. Mais afin de pouvoir utiliser les algorithmes de localisation, il faut préalablement étalonner le système multi-caméras. Peu d'approches ont été proposées pour calibrer des caméras à champs non-recouvrants.

Dans [22], l'auteur propose d'utiliser une caméra mobile supplémentaire se déplaçant à proximité des caméras à calibrer (restant statiques). L'étalonnage est obtenu à un facteur d'échelle près, à l'aide d'une reconstruction 3D.

Une autre approche consiste à utiliser un miroir plan qui crée un recouvrement entre les champs de vue des caméras. L'étalonnage extrinsèque du système multi-caméra est réalisé à l'aide d'une mire d'étalonnage dont la

géométrie est connue [15] ou inconnue [17]. Des caméras statiques à champs disjoints peuvent être également être calibrées en suivant un objet mobile. L'information manquante sur la trajectoire de l'objet est estimée dans les zones non observables [2, 25]. De manière similaire, Lamprecht *et al* [16] calibrent un système multi-caméras en mouvement. Un objet doit être suivi successivement dans chaque caméra et un *a priori* sur la vitesse du système est nécessaire. Une autre approche [7] consiste à utiliser la contrainte de rigidité entre les différents capteurs du système multi-caméras en mouvement. L'évolution temporelle de la pose de chaque caméra est déterminée. Puis, on déduit la pose relative entre les différentes caméras (paramètres extrinsèques), qui ne varie pas au cours du temps. Des travaux supplémentaires ont été proposés par Dai *et al* [4] pour moyennner des matrices de rotation, afin d'étalonner extrinsèquement des caméras.

Une fois les paramètres extrinsèques estimés, l'estimation minimale de la pose est possible [20], pour localiser le système multi-caméras [3, 9, 12, 13, 26], avec un filtre de Kalman étendu [11, 24] ou des algorithmes de reconstruction à partir du mouvement (structure from motion) [26, 28, 29]. Certains de ces articles déclarent utiliser un étalonnage extrinsèque manuel. Cependant, des mesures approximatives des angles ou des distances (avec un mètre) peuvent être trop imprécises.

Dans cet article, nous proposons une méthode flexible pour obtenir un étalonnage extrinsèque précis d'un ensemble mobile de caméras rigidement liées. La flexibilité est liée à la simplicité de mise en œuvre : pas besoin de miroir, ni d'*a priori* sur la vitesse du système multi-caméras, ni de parfaitement connaître au préalable la géométrie d'une mire.

Cet article présente une amélioration de l'algorithme proposé par Esquivel *et al* [7], en étendant l'algorithme classique d'ajustement de faisceaux décrit par Triggs *et al* [30] au cas de plusieurs caméras rigidement liées (avec des champs éventuellement disjoints) dans l'optique d'un étalonnage extrinsèque. Or, [4] et [7] considèrent tous deux que les trajectoires estimées séparément pour chaque caméra sont suffisamment précises pour obtenir des paramètres extrinsèques globalement cohérents. Nous ne faisons pas cette hypothèse. Afin d'augmenter la précision, les trajectoires calculées dans l'étape d'initialisation sont également raffinées dans la dernière étape.

Après une formalisation du problème (Partie 2), nous présenterons une initialisation linéaire des paramètres extrinsèques dans le cas de mouvements quelconques ou singuliers (Partie 3). La partie 4 est dédiée à l'optimisation non-linéaire qui affine les paramètres extrinsèques, la scène et les poses du système multi-caméras. Enfin (Partie 5), des résultats avec des données synthétiques et réelles valideront notre approche.

## 2 Formalisation du problème

Soient  $N_{cam} \geq 2$  caméras rigidement liées dont les paramètres intrinsèques sont connus, et avec des champs de vue non-recouvrants. Les caméras  $C_i$  sont synchronisées et effectuent  $K$  mouvements. Ainsi,  $K + 1$  est le nombre de poses du système multi-caméras.  $C_i^k$  représente la caméra  $n^o i$  à l'instant  $k$ . Chaque pose de la caméra  $C_i^k$  est exprimée relativement à sa première pose à l'instant  $k = 0$ . Soit  $T_i^k$  la transformation homogène passant du repère de  $C_i$  de l'instant 0 à l'instant  $k$ , quelque soit  $k \in \llbracket 0..K \rrbracket$  et  $i \in \llbracket 1..N_{cam} \rrbracket$  (cf Figure 1). De même,  $\Delta T_i$  est la transformation homogène passant du repère de  $C_1$  à celui de  $C_i$ . Chaque transformation homogène  $T$  est exprimée à l'aide de la rotation  $R$  et de la translation  $\mathbf{t}$  telles que :

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

Lorsque le système multi-caméras se déplace, chaque caméra  $C_i$  observe une scène statique  $S_i$  de points 3D.  $s_i^k$  sont les points détectés dans le plan image de  $C_i^k$ , correspondants aux points de la scène  $S_i$ . L'étalonnage (cf

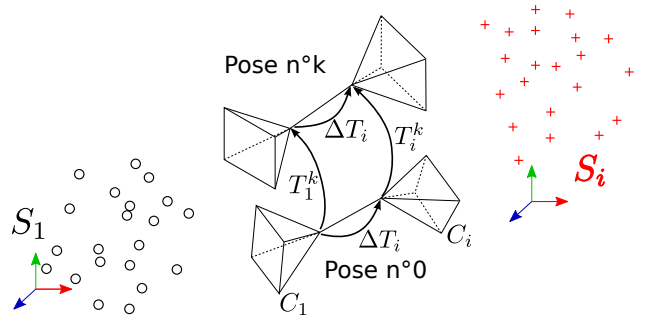


FIGURE 1 – Système multi-caméras à champs non-recouvrants se déplaçant dans un environnement statique.

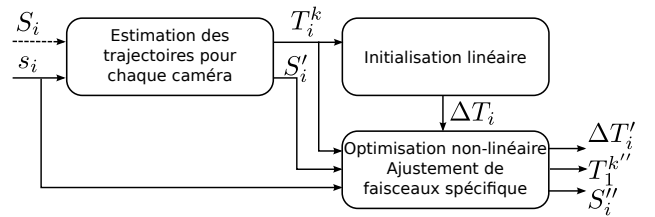


FIGURE 2 – Vue d'ensemble de l'étalonnage extrinsèque. Le symbole apostrophe distingue les estimations successives.

Figure 2) consiste à tout d'abord calculer les poses  $T_i^k$  de chaque caméra (cf ci-dessous : estimation des trajectoires). Ensuite, les paramètres extrinsèques  $\Delta T_i$  sont initialisés linéairement (Partie 3), puis affinés en même temps que les scènes et les poses du système multi-caméras grâce à un ajustement de faisceaux spécifique (Partie 4).

**Estimation des trajectoires** Premièrement, les trajectoires de chaque caméra sont calculées indépendamment. Pour cela, si la scène est de géométrie inconnue, un algorithme de structure from motion est utilisé (comme décrit dans [10]). *A contrario*, si la géométrie de la scène est approximativement connue, alors les poses de chaque caméra sont initialisées par la méthode décrite par Dementhon [5]. Ensuite, pour chaque caméra  $C_i$ , un ajustement de faisceaux classique [30] optimise à la fois les poses des caméras  $T_i^k$  et la scène  $S_i$ . Par la suite, nous supposons qu'une mesure de distance entre 2 points est disponible pour chaque scène  $S_i$ . Ainsi, l'ensemble du système a le même facteur d'échelle, et les facteurs d'échelle relatifs  $\delta_i$  entre les scènes  $S_1$  et  $S_i$  sont fixés à 1. Si ces mesures ne sont pas disponibles, les  $\delta_i$  peuvent être ajoutés aux paramètres à retrouver lors de l'initialisation.

### 3 Initialisation linéaire

#### 3.1 Mouvements 3D : cas général

Il est possible d'estimer linéairement les poses relatives  $\Delta T_i$  des caméras (paramètres extrinsèques) à partir des trajectoires  $T_1^k$  et  $T_i^k$ . L'hypothèse de rigidité entre les caméras et un changement de base permettent d'obtenir :

$$\forall i \in \llbracket 1..N_{cam} \rrbracket, \forall k \in \llbracket 1..K \rrbracket, T_1^k \Delta T_i = \Delta T_i T_i^k \quad (2)$$

Ces équations sont de la forme  $AX = XB$ , où  $X$  est la matrice inconnue. Ce sont les mêmes équations que l'on retrouve pour les problèmes d'étalonnage bras-œil [6], dont les solutions représentent les rotations soit par des quaternions unitaires [7], soit par des matrices orthogonales  $3 \times 3$  [1]. Rappelons tout d'abord la propriété (3), où  $A$ ,  $B$  et  $C$  sont des matrices,  $\otimes$  représente le produit de Kronecker et  $vec$  renvoie un vecteur colonne en concaténant la transposée des lignes d'une matrice. Le lecteur peut se référer à [27, §1.8, §2.11] pour plus de détails.

$$vec(ABC) = (A \otimes C^T)vec(B) \quad (3)$$

A l'aide de la définition (1) et de la propriété (3), l'équation (2) se scinde en deux parties :

$$\forall i \in \llbracket 1..N_{cam} \rrbracket, \forall k \in \llbracket 1..K \rrbracket,$$

$$(2) \Leftrightarrow \begin{cases} R_1^k \Delta R_i = \Delta R_i R_i^k & (4a) \\ R_1^k \Delta \mathbf{t}_i + \mathbf{t}_1^k = \Delta R_i \mathbf{t}_i^k + \Delta \mathbf{t}_i & (4b) \end{cases}$$

$$\Leftrightarrow \begin{cases} (I_9 - R_1^k \otimes R_i^k)vec(\Delta R_i) = 0_{9 \times 1} & (5a) \\ (I_3 - R_1^k)\Delta \mathbf{t}_i = \mathbf{t}_1^k - \Delta R_i \mathbf{t}_i^k & (5b) \end{cases}$$

Où  $I_n$  est la matrice identité de taille  $n$ .

Comme suggéré dans [1], nous avons choisi la représentation matricielle des rotations (ce qui permet d'exprimer de nouvelles équations dans la partie 3.2) et une solution en deux étapes : on estime premièrement les rotations  $\Delta R_i$ , puis les translations  $\Delta \mathbf{t}_i$ . Sous l'hypothèse de mouvements 3D quelconques, l'équation (2) admet une solution unique. Dans un premier temps, pour chaque caméra  $C_i$ ,

la rotation  $\Delta R_i$  est estimée à l'aide de l'équation (6) :  $\forall i \in \llbracket 2..N_{cam} \rrbracket$ ,

$$(5a) \Rightarrow \underbrace{\begin{pmatrix} I_9 - R_1^1 \otimes R_i^1 \\ \vdots \\ I_9 - R_1^k \otimes R_i^k \\ \vdots \\ I_9 - R_1^K \otimes R_i^K \end{pmatrix}}_{=L_i} vec(\Delta R_i) = 0_{9K \times 1} \quad (6)$$

Dans le cas général (mouvements 3D du système multicaméras), le rang de  $L_i$  vaut 8. Soit  $v_i$  un vecteur du noyau de  $L_i$  et soit  $V_i$  la matrice  $3 \times 3$  telle que  $v_i = vec(V_i)$ , on a alors :

$$\Delta R_i = V_i sign(det(V_i)) |det(V_i)|^{-\frac{1}{3}} \quad (7)$$

Dans un deuxième temps, les translations  $\Delta \mathbf{t}_i$  sont estimées grâce à l'équation de rang plein (8), obtenue pour chaque mouvement  $k$  grâce à (5b) :  $\forall i \in \llbracket 2..N_{cam} \rrbracket$ ,

$$\begin{pmatrix} I_3 - R_1^1 \\ \vdots \\ I_3 - R_1^k \\ \vdots \\ I_3 - R_1^K \end{pmatrix} \Delta \mathbf{t}_i = \begin{pmatrix} \mathbf{t}_1^1 - \Delta R_i \mathbf{t}_i^1 \\ \vdots \\ \mathbf{t}_1^k - \Delta R_i \mathbf{t}_i^k \\ \vdots \\ \mathbf{t}_1^K - \Delta R_i \mathbf{t}_i^K \end{pmatrix} \quad (8)$$

#### 3.2 Mouvements singuliers

**Vue d'ensemble.** Cette partie met en exergue les mouvements critiques pour lesquels les équations (6) et (8) deviennent singulières et ne peuvent plus être utilisées telles quelles. Premièrement, les translations pures sont des mouvements singuliers et ont déjà été étudiées par Esquivel [7]. Cependant, il existe d'autres singularités. Comme ce problème d'étalonnage extrinsèque peut être formulé avec les mêmes équations que l'étalonnage bras-œil, nous pouvons déduire les cas singuliers à partir de travaux antérieurs [1, 8]. Kim and Chung [11] analysent les cas singuliers pour un sujet similaire : la reconstruction 3D et l'estimation du mouvement à partir d'une paire stéréo calibrée sans correspondance stéréo. Par conséquent, les cas singuliers ont lieu lorsque les axes  $\mathbf{n}_1^k$  des rotations  $R_1^k$  sont parallèles pour tout  $k \in \llbracket 1..K \rrbracket$ . Autrement dit, les mouvements singuliers sont l'orbite de la première pose (pour  $k = 0$ ) sous l'action du groupe  $G$ , où  $G$  est composé des vissages d'axes colinéaires.

Les mouvements singuliers sont soit des rotations et des vissages autour d'un même axe (lorsque les axes  $\mathbf{n}_1^k$  sont confondus), soit des translations pures (selon un même axe ou des axes différents), soit des vissages avec des axes parallèles (lorsque les axes  $\mathbf{n}_1^k$  sont différents). Pour tous ces mouvements singuliers, l'étalonnage est partiel : seulement certains paramètres peuvent être estimés. La table 1 énumère le nombre de degrés de liberté observables pour les paramètres extrinsèques.

Les mouvements plans peuvent être vus comme plusieurs vissages dont les axes  $\mathbf{n}_1^k$  sont parallèles, et dont les translations selon ces axes sont nulles. Comme illustré dans la Table 1 et la Figure 3, l'une des dimensions de la translation extrinsèque  $\Delta \mathbf{t}_i$  n'est pas observable selon la normale au plan  $\mathbf{n}$ . On peut seulement affirmer que  $\Delta \mathbf{t}_i$  est de la forme :

$$\Delta \mathbf{t}_i = \mathbf{t}_{i\perp} + \alpha_i \mathbf{n} \quad (9)$$

où  $\alpha_i$  est un réel quelconque et  $\mathbf{t}_{i\perp}$  est la projection de  $\Delta \mathbf{t}_i$  dans le plan du mouvement.

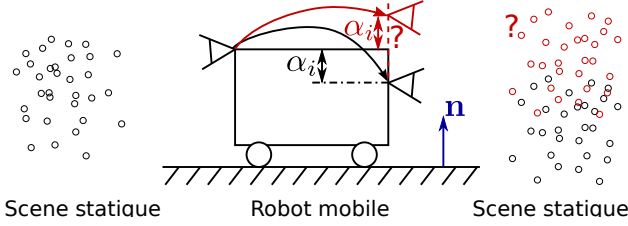


FIGURE 3 – Mouvement plan (cas singulier) : la hauteur relative  $\alpha_i$  entre les caméras (par rapport au plan du mouvement) n'est pas observable. Les observations ne changent pas entre le système réel (en noir) et estimé (en rouge).

**Solutions pour traiter les cas singuliers.** Cette partie traitera le cas des mouvements plans, qui est le cas singulier le plus fréquent pour les robots mobiles. Une estimation initiale des paramètres extrinsèques est proposée, puis nous montrons une méthode pratique pour initialiser la hauteur relative des caméras. Enfin, l'ajustement de faisceaux spécifique est appliqué (§4).

**Estimation de la rotation :** Ce cas ne peut pas être résolu ni par la méthode d'Esquivel *et al* pour des mouvements quelconques [7, §4.1], ni par notre initialisation linéaire (§3.1), où la première étape d'estimation de la rotation (6) échoue car le système linéaire n'est pas de rang plein. En effet, comme  $R_1^k$  et  $R_i^k$  ont les mêmes valeurs propres, les matrices  $L_i$  sont au plus de rang 6.

Par conséquent, lors de mouvements plans, de nouvelles contraintes doivent être ajoutées à l'initialisation précédente (6) afin de pouvoir estimer les rotations  $\Delta R_i$ . Sans perte de généralité, on choisit comme premier mouvement  $T_1^1$  un mouvement avec une rotation et une translation non nuls. Ainsi, en notant  $\mathbf{t}_i^k = (I_3 - R_i^k)\mathbf{t}_i^1 - (I_3 - R_i^1)\mathbf{t}_i^k$  et en supposant que  $\mathbf{t}_i^k \neq \mathbf{0}_{3 \times 1}$ , on résout le système de rang plein suivant :  $\forall i \in \llbracket 2..N_{cam} \rrbracket, \forall k \in \llbracket 2..K \rrbracket$ ,

$$\begin{pmatrix} I_3 \otimes \mathbf{t}_i^k \\ I_3 \otimes (\mathbf{n}_i^k)^\top \\ I_3 \otimes (\mathbf{t}_i^k \times \mathbf{n}_i^k)^\top \end{pmatrix} \text{vec}(\Delta R_i) = \begin{pmatrix} \mathbf{t}_i^k \\ \mathbf{n}_i^k \\ \mathbf{t}_i^k \times \mathbf{n}_i^k \end{pmatrix} \quad (10)$$

**Démonstration :** *Primo*, comme  $R_1^1$  et  $R_1^k$  commutent car leurs axes de rotation sont parallèles, on a :  $\forall k \in \llbracket 1..K \rrbracket$ ,

$$(I_3 - R_1^k)(I_3 - R_1^1) - (I_3 - R_1^1)(I_3 - R_1^k) = \mathbf{0}_{3 \times 3} \quad (11)$$

*Secundo*, avec l'équation (11), en partant des équations (5b) pour  $k = 1$  et  $k \neq 1$ , on peut simplifier l'expression  $(I_3 - R_1^k)(5b)_{k=1} - (I_3 - R_1^1)(5b)_{k \neq 1}$  en (12) :

$$(5b) \& (11) \Rightarrow \begin{aligned} & (I_3 - R_1^k)(\mathbf{t}_1^1 - \Delta R_i \mathbf{t}_i^1) \\ & - (I_3 - R_1^1)(\mathbf{t}_1^k - \Delta R_i \mathbf{t}_i^k) = \mathbf{0}_{9 \times 1} \quad (12) \end{aligned}$$

$$(12) \Leftrightarrow \begin{aligned} & (I_3 - R_1^k)\Delta R_i \mathbf{t}_i^1 - (I_3 - R_1^1)\Delta R_i \mathbf{t}_i^k \\ & = (I_3 - R_1^k)\mathbf{t}_1^1 - (I_3 - R_1^1)\mathbf{t}_1^k \quad (13) \end{aligned}$$

$$(13) \& (4a) \Leftrightarrow \begin{aligned} & \Delta R_i(I_3 - R_i^k)\mathbf{t}_i^1 - \Delta R_i(I_3 - R_i^1)\mathbf{t}_i^k \\ & = (I_3 - R_1^k)\mathbf{t}_1^1 - (I_3 - R_1^1)\mathbf{t}_1^k \quad (14) \end{aligned}$$

$$\Leftrightarrow \Delta R_i \mathbf{t}_i^k = \mathbf{t}_1^k \quad (15)$$

D'autre part, les axes  $\mathbf{n}_i^k$  des rotations  $R_i^k$  vérifient :

$$\forall i \in \llbracket 1..N_{cam} \rrbracket, \forall k \in \llbracket 1..K \rrbracket, \Delta R_i \mathbf{n}_i^k = \mathbf{n}_1^k \quad (16)$$

De plus,  $\mathbf{t}_i^k$  et  $\mathbf{n}_i^k$  sont linéairement indépendants car le mouvement est plan. Par conséquent, les vecteurs  $\mathbf{t}_i^k$ ,  $\mathbf{n}_i^k$  et le produit vectoriel  $\mathbf{t}_i^k \times \mathbf{n}_i^k$ , forment une famille linéairement indépendante. Finalement, en utilisant la propriété (3), on obtient le système de rang plein (10).  $\square$

En pratique, les systèmes (6) et (10) sont concaténés pour estimer  $\Delta R_i$ . De plus, comme les conditions de planéité ne sont pas parfaites en présence de bruit, les équations (11) sont quasiment vérifiées. Il est donc préférable d'orthogonaliser les matrices de rotation estimées via une SVD.

**Estimation de la translation :** Pour les mouvements plans, la translation demeure partiellement estimable. Ainsi, après l'initialisation, les hauteurs relatives des caméras  $\alpha_i$  restent inconnues. Au lieu d'utiliser des valeurs aberrantes, on les initialise à une éventuelle valeur approximativement connue *a priori* ou 0 m sinon. En pratique, au moins une pose non-coplanaire suffit à s'affranchir de la singularité. Par exemple, si le système multi-caméras est embarqué, il suffirait d'avoir un dos-d'âne sur la trajectoire du robot mobile.

Une autre solution consiste à déplacer le système multi-caméras, afin qu'au moins un point de la scène  $S$  soit vu par chaque caméra à différents instants. Pendant la phase d'acquisition, le système multi-caméras est déplacé afin que les scènes observées soient permutées (*i.e.* la scène observée par l'une des caméras à un instant donné, peut ensuite être vue par une autre caméra.). Avec des correspondances connues entre les images, les hauteurs relatives  $\alpha_i$  sont calculables. En résumé, pour un mouvement plan, il est possible d'estimer à la fois les rotations  $\Delta R_i$  et les translations  $\Delta \mathbf{t}_i$ . Nous allons montrer comment il est possible de compléter l'initialisation précédente (§3.1) avec de nouvelles contraintes (obtenues par la permutation des scènes). Que le mouvement soit plan ou non, ces contraintes restent valables.  $T_{i \rightarrow \sigma(i)}^k$  est la transformation homogène entre la caméra  $C_i$  à l'instant 0 et la caméra  $C_{\sigma(i)}$  à l'instant  $k$ , avec la permutation  $\sigma$ . Pour  $N_{cam} > 2$ , on obtient un système d'équations couplées. Pour  $N_{cam} = 2$ ,  $\sigma(1) = 2$  et

Mouvements	Axes de rotation	$\Delta R_i$	$\Delta \mathbf{t}_i$
1) Rotations et vissages selon un axe $\mathbf{a}$	confondus ( $\mathbf{a} = \mathbf{n}_1^k$ )	2	2
2) Translations pures selon un axe $\mathbf{a}$ (ou plusieurs axes)	pas de rotation	2 (3)	0
3) Mouvements plans et vissages selon différents axes	différents et parallèles	3	2
4) 3D (cas général)	différents	3	3

TABLE 1 – Nombre de degrés de liberté observables des paramètres extrinsèques avec  $K$  mouvements du système multi-caméras. Notons que pour le mouvement 1), si l’axe  $\mathbf{a}$  passe par tous les centres optiques, alors  $\Delta \mathbf{t}_i$  n’est pas observable. Pour les mouvements 1) et 2), si on applique un vissage selon  $\mathbf{a}$  à  $S_i$  et aux paramètres extrinsèques, alors on ne change pas les observations.

$\sigma(2) = 1$ , on obtient les équations suivantes :

$$T_{1 \rightarrow 2}^k (\Delta T_2)^{-1} = \Delta T_2 T_{2 \rightarrow 1}^k \quad (17)$$

$$\Leftrightarrow \begin{cases} R_{1 \rightarrow 2}^k (\Delta R_2)^\top = \Delta R_2 R_{2 \rightarrow 1}^k & (18a) \\ -\Delta R_2 R_{2 \rightarrow 1}^k + \mathbf{t}_{1 \rightarrow 2}^k = \Delta R_2 \mathbf{t}_{2 \rightarrow 1}^k + \Delta \mathbf{t}_2 & (18b) \end{cases}$$

$$\Leftrightarrow \begin{cases} (I_9 - (R_{1 \rightarrow 2}^k \otimes R_{2 \rightarrow 1}^k) \Phi_9) \text{vec}(\Delta R_2) = \mathbf{0}_{9 \times 1} & (19a) \\ (I_3 + \Delta R_2 R_{2 \rightarrow 1}^k) \Delta \mathbf{t}_2 = \mathbf{t}_{1 \rightarrow 2}^k - \Delta R_2 \mathbf{t}_{2 \rightarrow 1}^k & (19b) \end{cases}$$

où  $\Phi_9$  est la matrice de permutation telle que  $\text{vec}(M^\top) = \Phi_9 \text{vec}(M)$ .

Les équations (19a) peuvent être ajoutées aux équations (6), mais cela n’augmente pas suffisamment le rang du système dans le cas de mouvements plans (contrairement à l’équation (10)). Finalement, les équations (8) et (19b) sont concaténées pour estimer linéairement la translation  $\Delta \mathbf{t}_2$  en totalité.

## 4 Ajustement de faisceaux spécifique

La dernière étape consiste à affiner à la fois les paramètres extrinsèques  $\Delta T_i$ , les scènes 3D et les poses du système multi-caméras  $T_1^k$ . Cette étape n’a pas été proposée ni par [4], ni par [7]. Notons que les similitudes avec l’étalonnage bras-œil ne sont plus valables dans cette partie.

Même si les  $N_{cam}$  ajustements de faisceaux précédents sont des estimateurs du maximum de vraisemblance pour chaque caméra, ce n’est pas le cas pour un système multi-caméras. En effet, lorsque les ajustements de faisceaux sont considérées indépendamment les uns des autres, l’ensemble du système est sur-paramétré. Par exemple, Triggs *et al*[30] propose un ajustement de faisceaux pour une ou plusieurs caméras se déplaçant indépendamment les unes des autres. Cependant, toutes les poses de chaque caméras, exprimées dans un repère global, sont optimisées<sup>1</sup>. Dans notre cas, les caméras sont rigidement liées. Comme proposé par King [14] pour une paire stéréo, une paramétrisation minimale est préférable : seules les poses de la caméra maître  $C_1$  sont exprimées dans un repère global ( $6(K+1)$  paramètres pour  $\mathbf{t}_1^k$  et  $R_1^k$ ). Les poses des autres caméras sont exprimées relativement à la caméra maître ( $6(N_{cam}-1)$  paramètres pour  $\Delta \mathbf{t}_i$  et  $\Delta R_i$ , représentées localement par des angles d’Euler).

1. Les paramètres intrinsèques peuvent également être optimisés pour des techniques d’auto-étalonnage

Si le repère global était défini comme celui de la première pose de la caméra maître, la paramétrisation la plus éparse serait atteinte. Mais en pratique, des minima locaux peuvent être engendrés par cette représentation (pour plus de détails sur les libertés de jauge, le lecteur peut se référer à [30]). Ainsi, aucune pose de caméra n’est arbitrairement choisie comme le repère monde.

L’algorithme est initialisé avec l’estimation linéaire des paramètres extrinsèques, et l’union des points de toutes les scènes exprimés dans le même repère. Soit  $M$  le nombre total de points 3D. Au final,  $6(K+1) + 6(N_{cam}-1) + 3M$  paramètres sont optimisés par l’algorithme de Levenberg Marquardt en minimisant les erreurs de reprojection.

Sous l’hypothèse d’un bruit gaussien des  $s_i^k$ , l’algorithme proposé est un estimateur de maximum de vraisemblance. Les occultations de la scène sont supportées. De plus, chaque point 3D est potentiellement observable par n’importe quelle caméra (à différents instants), donc les permutations des scènes sont également possibles.

## 5 Validation

### 5.1 Résultats avec des données synthétiques

Le protocole expérimental est le suivant, pour  $N_{cam} = 2$  caméras. 10 poses du système multi-caméras sont générées avec un mouvement 3D sans demi-tour. Les points de la scène  $S_i$  sont créés devant chaque caméra  $C_i$ . Cette vérité terrain sert à calculer les points images 2D  $s_i^k$ . Puis, un bruit additif gaussien d’écart type  $\sigma$  leur est appliqué. Les mesures de distance entre des points 3D (utilisées pour fixer le facteur d’échelle) sont également soumises à un bruit additif gaussien d’écart type 0.5 mm. Enfin, l’algorithme est testé à partir des données bruitées. Pour cette expérience, la transformation  $\Delta T_2$  correspond à la translation (0.1m 0.1m -2m) et à la rotation déterminée par les angles de roulis, tangage, lacet de  $(-179^\circ, -4^\circ, 171^\circ)$ .

Pour chaque valeur de  $\sigma$ , 10 mesures ont été faites. La précision de l’étalonnage est mesurable à la fois par la norme  $\|dT\|$  de la différence entre la translation réelle et estimée (et aussi par  $\|dT\|/\|\Delta T_2\|$  exprimé comme un pourcentage de la distance séparant les caméras), et par l’erreur angulaire  $dR$ . Plus précisément :

$$dR = d(\hat{R}, R^*) = \arccos \left( \frac{\text{trace}(\hat{R}^\top R^*) - 1}{2} \right) \quad (20)$$

où  $R^*$  est la rotation parfaite et  $\hat{R}$  est la rotation estimée. La Figure 4 montre les performances de l’algorithme, avec une scène de 11 points 3D (d’environ 1.5 m par 0.5 m) pour  $S_1$  et  $S_2$  à une distance d’environ 1 m de chaque caméra  $C_i$ . Dans le cas d’un mouvement 3D, l’algorithme proposé dans [7] donne les mêmes résultats que notre initialisation linéaire. L’ajustement de faisceaux spécifique améliore les deux méthodes précédentes (l’estimation de la translation et de la rotation sont environ respectivement quatre et trois fois plus précises).

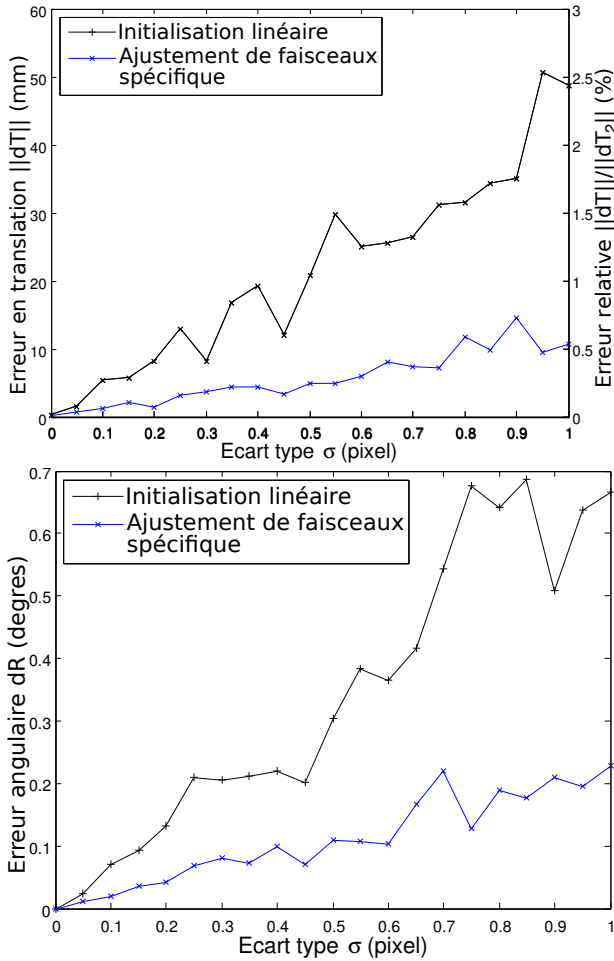


FIGURE 4 – Résultats avec des données synthétiques : erreur d’étalonnage extrinsèque en fonction du niveau de bruit. Avec des données réelles, l’écart type est en dessous de 0.1 pixel.

## 5.2 Résultats avec des données réelles

Dans un premier temps, la précision de l’étalonnage proposé est démontrée avec une paire stéréo avec des champs recouvrant. Puis, la faisabilité de l’approche est illustrée à l’aide de caméras à champs non-recouvrants embarquées sur un véhicule. Pour les amers visuel de la scène, on utilise des cibles circulaires similaires à [21], développées dans [17, §VII]. Une cible est composée d’un disque noir

et d’un code-barres circulaire (cf Figure 5). Elles sont détectées avec une précision sous-pixellique. Le détecteur trouve automatiquement le centre de la cible et son label. Par conséquent, les correspondances entre les images sont effectuées automatiquement. La résolution des images est 1600x1200. La distance focale des caméras est de 5.6 mm et la taille d’un pixel est de 4.4  $\mu m$ .

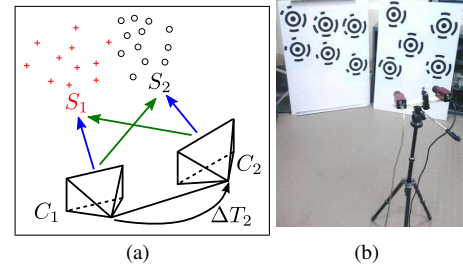


FIGURE 5 – Étalonnage d’une paire stéréo avec l’hypothèse de champs de vue disjoints, pour des caméras à champs recouvrants.

**Paire stéréo.** Dans ce paragraphe, une paire stéréo à champs recouvrant est calibrée avec l’hypothèse de champs non-recouvrants : chaque caméra n’observe qu’une seule scène à la fois (cf flèches bleues de la Figure 5a, ou flèches vertes en cas de permutation).  $S_1$  et  $S_2$  sont respectivement sur le panneau de gauche et de droite (cf Figure 5b). Les résultats de notre algorithme sont comparés à un algorithme classique d’étalonnage, dont les champs de vue sont recouvrants, en appliquant des mouvements à la paire stéréo (cf Table 2). Pour chaque cas, on prend 15 images. La distance séparant les caméras est d’environ 22 cm. Tous les écarts types des erreurs de reprojection valent environ 0.03 *pix*.

Champs	Recouvrants		Disjoints	
	Plan	3D	non	oui
Mouvements	n/a	non	non	oui
Scènes permutes ?	n/a	non	non	oui
$\ dT\ $ (mm)	0.14	0.41	60.39	0.08
$dR(^{\circ})$	0.008	0.011	0.038	0.011

TABLE 2 – Comparaison de la précision des étalonnages extrinsèques en fonction de l’hypothèse de champs de vue recouvrants, du mouvement et de la permutation de scènes. La référence est l’étalonnage classique de la paire stéréo avec des champs recouvrants et en lui appliquant un mouvement 3D.

Les résultats de la Table 2 montrent que notre étalonnage avec des champs non-recouvrants est tout aussi précis qu’un algorithme classique dont les champs sont recouvrants (cf colonne 3D). Lorsque les mouvements sont quasiment plans, et si les scènes ne sont pas permutes (cf 4<sup>me</sup> colonne), alors la singularité induit une baisse de la précision. En effet, la translation extrinsèque n’est alors pas

totallement observable (cf Figure 3). *A contrario*, lorsque les scènes sont permutées, on obtient la même précision qu'un algorithme classique (cf dernière colonne) : moins de 0.5 mm pour la translation (soit  $\|dT\|/\|\Delta T_2\| < 0.2\%$ ) et environ  $0.01^\circ$  pour la rotation.

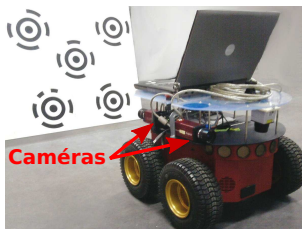


FIGURE 6 – Étalonage d'une paire stéréo avec des caméras embarquées à champs non-recouvrants.

**Système mutli-caméras embarqué.** On embarque deux caméras sur un robot mobile : l'une à l'avant, l'autre à l'arrière (la distance les séparant est d'environ 22 cm, cf Figure 6). Selon Pless [23], il s'agit de la meilleure disposition pour l'estimation de mouvement avec deux caméras. Les principaux avantages sont la symétrie du véhicule et la robustesse face aux conditions extérieures d'illumination, pour des applications de localisation.

Le robot manœuvre sur un sol plat entre deux scènes. Pendant l'acquisition, un demi-tour est effectué afin de permuter les scènes observées. Comme les occultations de la scène sont supportées, on ajoute des prises de vue où les scènes sont vues de côté, pour améliorer la reconstruction des scènes 3D. L'algorithme renvoie les paramètres extrinsèques. L'écart type des erreurs de reprojection est d'environ  $0.09 \text{ pix}$ . On remarque que si l'équation (19b) n'était pas ajoutée pour initialiser la translation extrinsèque, on aurait une erreur d'environ  $1.6 \text{ m}$  selon la normale au plan du mouvement.

Pour valider l'étalonnage, nous considérons une paire d'images stéréo qui n'a pas été utilisée lors de l'étalonnage. En utilisant l'estimation précédente des points 3D et des paramètres extrinsèques, la pose du système multi-caméra est calculée. Les erreurs de reprojection sont centrées et ont un écart type d'environ  $0.05 \text{ pix}$  (même ordre de grandeur que précédemment).

## 6 Conclusion

Une méthode flexible pour l'étalonnage extrinsèque d'un système multi-caméras a été proposée et validée aussi bien avec des données synthétiques que des données réelles. Tout d'abord, les paramètres extrinsèques sont initialisés linéairement à partir des trajectoires de chaque caméra. Puis, l'ajustement de faisceaux spécifique, qui est un estimateur de maximum de vraisemblance, affine à la fois les points 3D, les poses du système multi-caméras et les paramètres extrinsèques. Les mouvements singuliers ont été mis en exergue : pour le cas du mouvement plan, la permutation des scènes permet d'obtenir un étalonnage com-

plet. De plus, les résultats montrent que notre étalonnage, sous l'hypothèse de champs de vue disjoints, est tout aussi précis qu'un étalonnage classique dont les champs de vue sont recouvrants.

Dans nos prochains travaux, les cibles circulaires seront remplacées par des points d'intérêt afin de trouver un compromis entre souplesse de mise en œuvre et précision. Pour avoir une implémentation la plus rapide possible, nous examinerons analytiquement la jacobienne de l'erreur de projection en fonction des paramètres optimisés [18].

## Remerciements

Ce travail a été financé par les Fonds Européen de Développement Régional (FEDER), la région Auvergne et Clermont Communauté dans le cadre du projet VIPA (Véhicule Individuel Public Autonome).

## Références

- [1] N. Andreff, R. Horaud, and B. Espiau. Robot hand-eye calibration using structure-from-motion. *The International Journal of Robotics Research*, 20(3) :228, 2001.
- [2] N. Anjum, M. Taj, and A. Cavallaro. Relative position estimation of non-overlapping cameras. In *IEEE International Conference on Acoustics, Speech and Signal Processing, 2007. ICASSP 2007*, volume 2, 2007.
- [3] B. Clipp, J.H. Kim, J.M. Frahm, M. Pollefeys, and R.I. Hartley. Robust 6dof motion estimation for non-overlapping, multi-camera systems. In *IEEE Workshop on Applications of Computer Vision, 2008. WACV 2008*, pages 1–8, 2008.
- [4] Y. Dai, J. Trunpf, H. Li, N. Barnes, and R. Hartley. Rotation Averaging with Application to Camera-Rig Calibration. *Computer Vision—ACCV 2009*, pages 335–346, 2010.
- [5] Daniel F. Dementhon and Larry S. Davis. Model-based object pose in 25 lines of code. *International Journal of Computer Vision*, 15 :123–141, 1995.
- [6] M. Dhome. *Visual Perception Through Video Imagery*. Wiley-ISTE, 2009.
- [7] S. Esquível, F. Woelk, and R. Koch. Calibration of a Multi-camera Rig from Non-overlapping Views. In *Pattern Recognition : 29th DAGM Symposium, Heidelberg, Germany, September 12-14, 2007, Proceedings*, page 82. Springer-Verlag New York Inc, 2007.
- [8] I. Fassi and G. Legnani. Hand to sensor calibration : A geometrical interpretation of the matrix equation  $AX = XB$ . *Journal of Robotic Systems*, 22(9) :497, 2005.
- [9] J.M. Frahm, K. Koser, and R. Koch. Pose Estimation for Multi-camera Systems. In *Pattern recognition : 26th DAGM Symposium, Tübingen, Germany, August 30-1 September 2004 : proceedings*, page 286. Springer-Verlag New York Inc, 2004.



- [10] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004.
- [11] J.H. Kim and M.J. Chung. Absolute motion and structure from stereo image sequences without stereo correspondence and analysis of degenerate cases. *Pattern Recognition*, 39(9) :1649–1661, 2006.
- [12] J.H. Kim, R. Hartley, J.M. Frahm, and M. Pollefeys. Visual Odometry for Non-Overlapping Views Using Second-Order Cone Programming. *Lecture Notes in Computer Science*, 4844 :353, 2007.
- [13] Jun-Sik Kim, Myung Hwangbo, and T. Kanade. Motion estimation using multiple non-overlapping cameras for small unmanned aerial vehicles. In *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pages 3076–3081, 19-23 2008.
- [14] BA King. Optimisation of Bundle Adjustments for Stereo Photography. *International Archives Of Photogrammetry And Remote Sensing*, 29 :168–168, 1993.
- [15] R.K. Kumar, A. Ilie, J.-M. Frahm, and M. Pollefeys. Simple calibration of non-overlapping cameras with a mirror. *IEEE Conference on Computer Vision and Pattern Recognition, 2008. CVPR 2008*, pages 1–7.
- [16] B. Lamprecht, S. Rass, S. Fuchs, and K. Kyamakyia. Extrinsic camera calibration for an on-board two-camera system without overlapping field of view. In *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE*, pages 265–270, sept. 2007.
- [17] Pierre Lébraly, Clément Deymier, Omar Ait-Aider, Eric Royer, and Michel Dhôme. Flexible Extrinsic Calibration of Non-Overlapping Cameras Using a Planar Mirror : Application to Vision-Based Robotics. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2010*.
- [18] Pierre Lébraly, Eric Royer, Omar Ait-Aider, Deymier Clément, and Michel Dhôme. Fast Calibration of Embedded Non-Overlapping Cameras. In *International Conference on Robotics and Automation, ICRA 2011*.
- [19] Pierre Lébraly, Eric Royer, Omar Ait-Aider, and Michel Dhôme. Calibration of non-overlapping cameras - application to vision-based robotics. In *Proceedings of the British Machine Vision Conference*, pages 10.1–10.12. BMVA Press, 2010. doi :10.5244/C.24.10.
- [20] Hongdong Li, Richard I. Hartley, and Jae-Hak Kim. A linear approach to motion estimation using generalized camera models. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2008*. IEEE Computer Society, 2008.
- [21] Diego López de Ipi na, Paulo R. S. Mendonça, and Andy Hopper. Trip : A low-cost vision-based location system for ubiquitous computing. *Personal Ubiquitous Comput.*, 6(3) :206–219, 2002.
- [22] Etienne Mouragnon. *Reconstruction 3D et localisation simultanée de caméras mobiles : une approche temps-réel par ajustement de faisceaux local*. PhD thesis, Ecole Doctorale Sciences Pour l’Ingénieur de Clermont-Ferrand, 2007.
- [23] R. Pless. Using many cameras as one. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2003. Proceedings*, pages 587–593, 2003.
- [24] ME Ragab, KH Wong, JZ Chen, and MMY Chang. EKF Based Pose Estimation using Two Back-to-Back Stereo Pairs. In *IEEE International Conference on Image Processing, 2007. ICIP 2007*, volume 6, 2007.
- [25] A. Rahimi, B. Dunagan, and T. Darrell. Simultaneous calibration and tracking with a network of non-overlapping sensors. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 1, pages I–187 – I–194 Vol.1, 27 2004.
- [26] T. Sato, S. Ikeda, and N. Yokoya. Extrinsic camera parameter recovery from multiple image sequences captured by an omni-directional multi-camera system. *Computer Vision-ECCV 2004*, pages 326–340, 2004.
- [27] W.H. Steeb and T.K. Shi. *Matrix calculus and Kronecker product with applications and C++ programs*. World Scientific Pub Co Inc, 1997.
- [28] H. Stewenius and K. Åström. Structure and motion problems for multiple rigidly moving cameras. *European Conference on Computer Vision, ECCV 2004*, pages 252–263, 2004.
- [29] Henrik Stewenius, Magnus Oskarsson, and Kalle Åström. Reconstruction from planar motion image sequences with applications for autonomous vehicles. In *Scandinavian Conf. on Image Analysis*, pages 609–618, 2005.
- [30] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. Bundle adjustment—a modern synthesis. *Vision algorithms : theory and practice*, pages 153–177, 2000.