

Calibration métrique faible et construction interactive de modèles 3D de scènes

Weak metric calibration and 3D interactive construction of scene models

Vannary Meas-Yedid, Jean-Philippe Tarel, André Gagalowicz
INRIA, Domaine de Voluceau, Rocquencourt
B.P. 105, 78153 Le Chesnay Cedex, France.

Email: vannary@bora.inria.fr, tarel@bora.inria.fr, ag@bora.inria.fr

Résumé

Dans le cadre d'un système interactif de construction de modèles tridimensionnels, à partir de différentes vues d'une scène réelle, nous avons développé une méthode de calibration simple, tant du point de vue des calculs que de celui des données à fournir. Cette méthode permet de combiner diverses primitives tridimensionnelles et d'utiliser, dans une même phase de calibration, des segments, des triangles, des rectangles, entre autres. Les données à fournir sont les tailles et les orientations dans un même repère des segments qui constituent les primitives. En revanche, il n'est pas nécessaire de connaître les positions relatives de ces primitives. Dans la pratique les orientations sont choisies orthogonales, ainsi de simples mesures de longueurs suffisent.

mots-clés: positionnement d'un robot, calibration, géométrie perspective, modélisation tridimensionnelle.

Abstract

In the framework of an interactive system to build 3D-model of a scene from its perspective view points, we have computed a simple calibration method. Our calibration method is simple with regards to computation and required data. This method allows different tridimensional primitives such as segments, triangles and rectangles to be combined and used in the same calibration. The data to be provided are absolute orientations and primitives' dimensions. We do not have to know relative position between primitives. In practice, if orientations are chosen to be orthogonal, then simple measurements of length are sufficient.

keywords: robot positioning, camera calibration, perspective geometry, tridimensional modelization.

1 Introduction

Il existe de nombreux liens entre le problème de positionnement d'un objet par rapport à une caméra, celui du calcul du mouvement de la caméra et de sa calibration. Ces trois problèmes ne peuvent donc pas être considérés comme indépendants. Par exemple, la calibration résout le problème du positionnement de la caméra par rapport à un objet particulier (la mire) tout en déterminant les paramètres intrinsèques de celle-ci. Autre exemple: si l'on détermine le positionnement d'une caméra à différents instants par rapport au même objet, il est simple d'obtenir le mouvement de cette caméra. Ces problèmes font tous intervenir trois groupes de paramètres.

- Les *paramètres intrinsèques* qui modélisent les caractéristiques internes de la caméra. Pour le modèle sténopé, le plus simple, ces caractéristiques sont au nombre de 4: la position du centre de l'image, et deux coefficients d'échantillonnage sur les deux axes de l'image.
- Les *paramètres extrinsèques*; c'est-à-dire la position et l'orientation de la caméra par rapport à un référentiel fixe. Suivant le problème ce référentiel peut être lié à un objet vu, ou à la caméra dans une autre position, ou enfin à une autre caméra.
- La *géométrie des objets* observés par la caméra.

Les paramètres extrinsèques sont à déterminer, dans les trois cas précisés, alors que les deux autres groupes de paramètres peuvent être supposés connus. Ainsi :

- Le problème de positionnement d'un objet par rapport à la caméra suppose que les paramètres intrinsèques soient connus, ainsi que tout ou une partie de la géométrie de l'environnement perçu, suivant la méthode de résolution adoptée. Les méthodes à base de points ou problème perspective à n points [Har80] [Hor89] [MM91] [TPP93] nécessitent au moins 3 points 3D connus. Des méthodes à base de lignes (au minimum 3 lignes) ont aussi été proposées [MDR88] [LH89] [CT90] ainsi que d'autres à base de primitives plus complexes comme

les angles [Kan87], les trièdres [CT86] ou les ellipses [CRF92].

- Les méthodes qui permettent de résoudre le problème du calcul du mouvement d'une caméra supposent que seuls les paramètres intrinsèques soient connus. Il existe des méthodes fondées sur les points, notamment la méthode des 8 points avec deux vues [LH81], ou basées sur les lignes [OF87] ce qui nécessite trois vues et au moins 6 lignes.
- Pour la calibration d'une caméra, c'est la géométrie de l'objet qui est supposée connue. Généralement pour que cette connaissance soit précise, on utilise une mire. De nombreuses méthodes à base de points [Tos87] (qui nécessite au moins six points) [Tsa86] [LT88] [Ma93] ou de lignes [Vai90] ont été proposées avec différents modèles de caméra, suivant le type de distorsions modélisées.
- Enfin, l'auto-calibration [MF92] ne nécessite théoriquement aucune connaissance a priori, si ce n'est une liste de points 2D mis en correspondance entre différentes vues.

On remarque que des solutions à cette catégorie de problèmes ont d'abord été proposées avec des points puis étendues à des types de primitives géométriques de plus en plus complexes.

Au cours du développement d'un système interactif de construction de modèles tridimensionnels, à partir de différentes vues d'une scène réelle, un problème pratique s'est posé à nous. Comment peut-on connaître les caractéristiques des différentes vues et obtenir la transformation qui fait passer du monde tridimensionnel à l'image? Nous avons donc besoin d'une méthode de calibration complète (paramètres intrinsèques et extrinsèques), avec des contraintes un peu particulières: la tâche du manipulateur doit être facilitée au maximum.

Le but de l'étude développée dans cet article n'est donc pas d'atteindre une grande précision dans la calibration. Cela demanderait d'avoir des informations redondantes, difficiles à obtenir sans avoir des mesures tridimensionnelles de la scène ou à partir d'images ne contenant pas de mire. Nous avons alors cherché à retrouver la transformation qui fait passer du monde tridimensionnel à l'image, avec un nombre rela-

tivement faible d'informations 3D mesurées de façon la plus simple possible.

Il nous a été en pratique difficile de disposer de nombreuses mesures de points 3D, notamment lorsque les points sont à plusieurs mètres, voire quelques dizaines de mètres les uns des autres, ou lorsque l'on ne dispose pas de mire. Nous avons donc été amenés à étudier une calibration nécessitant des données plus faciles à mesurer. Pour ce faire, nous avons développé une méthode à base de primitives géométriques plus complexes que des points, comme des segments et des rectangles. Cela évite d'avoir à connaître les positions, dans l'espace 3D, de ces primitives les unes par rapport aux autres.

Nous avons donc étudié une méthode de calibration à base de primitives qui ont un petit nombre de points (inférieur au nombre qui permettrait de faire directement une calibration à base de points, en considérant les sommets de cette primitive) comme les triangles, rectangles ou tétraèdres. Lorsque les *orientations* et les *tailles* des primitives de base sont connues dans le repère absolu, même si les *positions relatives* des primitives sont inconnues, il est possible d'effectuer la calibration complète des *paramètres intrinsèques et extrinsèques*. En choisissant des orientations particulières (des axes perpendiculaires), de simples mesures de longueurs suffisent.

Nous avons cherché à aboutir à des équations linéaires. Cette approche a été privilégiée, car ainsi nous avons pu réutiliser, dans un cadre plus général, l'algorithme proposé par Faugeras et Toscani [Tos87] qui présente comme avantage d'être analytique et permet la combinaison et la redondance des primitives. On peut alors réaliser de façon complète et simple la calibration pour chaque vue. Une fois la calibration effectuée, il est possible de calculer la position absolue de chaque primitive utilisée lors du processus de calibrage.

Après un rappel des équations de base liant un point de l'espace 3D à sa projection sur l'image (partie 2), nous décrivons dans la partie suivante une méthode de calibration à base de segments. A partir des équations obtenues, nous généralisons la méthode à des primitives plus complexes, constituées de segments (partie 4). Nous rappelons ensuite l'algorithme utilisé pour résoudre le système d'équations obtenu avec les segments comme avec des primitives plus complexes (partie 5), et discutons brièvement de

l'existence d'une solution (partie 6). Enfin, nous expliquons dans la partie 7 comment, une fois la calibration réalisée, nous construisons le modèle de la scène. Des résultats de calibration et de construction sur des scènes réelles sont présentés à titre d'illustration.

2 Rappel des équations de base

Il s'agit de calculer la transformation qui fait passer de l'espace tridimensionnel à l'image. La caméra est modélisée comme un trou d'épingle (modèle sténopé). La transformation est alors une projection perspective caractérisée par la

matrice 3×4 , $M = \begin{pmatrix} L_1 \\ L_2 \\ L_3 \end{pmatrix}$. L_1 , L_2 et L_3 sont des

vecteurs ligne de dimension quatre. On note l_i le vecteur constitué des trois premiers éléments de L_i . On pose ainsi $L_i = (l_i \quad l_{i,4})$.

Le point image (u, v) d'un point (x, y, z) est en coordonnées homogènes :

$$\begin{pmatrix} us \\ vs \\ s \end{pmatrix} = M \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = MX$$

On en déduit deux équations qui lient un point de l'espace 3D, X , à son image, (u, v) , linéaires en fonction des éléments de la matrice perspective M ¹.

$$\left. \begin{aligned} (uL_3 - L_1) \cdot X &= 0 \\ (vL_3 - L_2) \cdot X &= 0 \end{aligned} \right\} \quad (1)$$

En conséquence, si nous disposons de la connaissance d'un nombre suffisant de points 3D X_i et des images associées (u_i, v_i) , il est possible d'estimer M . Dans le modèle de caméra classique (avec une rétine rectangulaire) M ne dépend que de dix paramètres : 6 paramètres extrinsèques (position et orientation de la caméra), 4 intrinsèques (le centre de l'image (u_0, v_0) et deux coefficients d'échantillonnage a_u et a_v). La matrice perspective doit donc vérifier deux contraintes déduites de la modélisation de la caméra. Comme l'ont montré Faugeras et Toscani [Tos87], en ne tenant compte que d'une seule de ces contraintes ($\|l_3\| = 1$) le problème a une solution analytique. Chaque point fournissant

1. Un point "." représente le produit scalaire entre vecteurs.

deux équations (1), il faut donc au minimum six points, dont les coordonnées dans l'espace 3D sont définies dans le repère absolu de la scène. Un nombre supérieur de points rendra l'estimation moins sensible aux erreurs des données. Cela est en pratique difficile à obtenir lorsque l'on ne dispose pas de mire. Nous avons donc été amenés à étudier une calibration avec des primitives géométriques plus complexes que des points de manière à ne pas avoir à connaître la position dans l'espace 3D de ces primitives les une par rapport aux autres. Il n'est alors plus nécessaire de mesurer explicitement des coordonnées tridimensionnelles. Dans la partie suivante, la première primitive de base considérée sera le segment.

3 Segments

Un segment dans l'espace 3D est défini par ses deux extrémités X_0 et $X_0 + \Delta$, en coordonnées homogènes. Soit (u_1, v_1) et (u_2, v_2) , les images des deux sommets du segment. On a, d'après (1), les relations suivantes :

$$(u_1 L_3 - L_1) \cdot X_0 = 0 \quad (2)$$

$$(v_1 L_3 - L_2) \cdot X_0 = 0 \quad (3)$$

$$(u_2 L_3 - L_1) \cdot X_0 + (u_2 L_3 - L_1) \cdot \Delta = 0 \quad (4)$$

$$(v_2 L_3 - L_2) \cdot X_0 + (v_2 L_3 - L_2) \cdot \Delta = 0 \quad (5)$$

Combinons les équations de ce système sous une forme utile pour la suite :

$$(2)-(4) \quad (u_1 - u_2) L_3 \cdot X_0 = (u_2 L_3 - L_1) \cdot \Delta \quad (6)$$

$$(3)-(5) \quad (v_1 - v_2) L_3 \cdot X_0 = (v_2 L_3 - L_2) \cdot \Delta \quad (7)$$

$$u_2(2) - u_1(4) \quad (u_1 - u_2) L_1 \cdot X_0 = u_1(u_1 L_3 - L_1) \cdot \Delta \quad (8)$$

$$v_2(3) - v_1(5) \quad (v_1 - v_2) L_2 \cdot X_0 = v_1(v_2 L_3 - L_2) \cdot \Delta \quad (9)$$

On remarque alors que $(v_1 - v_2)(6) - (u_1 - u_2)(7)$ est une relation indépendante de la position 3D X_0 , du segment :

$$(v_1 - v_2) L_1 \cdot \Delta + (u_2 - u_1) L_2 \cdot \Delta + (u_1 v_2 - v_1 u_2) L_3 \cdot \Delta = 0$$

La dernière composante de Δ , en coordonnées homogènes, est nulle : $\Delta = \begin{pmatrix} \delta \\ 0 \end{pmatrix}$. δ caractérise le segment 3D en orientation et en longueur, indépendamment de sa position. En conséquence, avec les notations introduites $L_i = (l_i \quad l_{i,4})$, l'équation précédente s'écrit :

$$(v_1 - v_2) l_1 \cdot \delta + (u_2 - u_1) l_2 \cdot \delta + (u_1 v_2 - v_1 u_2) l_3 \cdot \delta = 0 \quad (10)$$

On retrouve l'équation de base de [MDR88]:

$$N \cdot M \Delta = N \cdot \begin{pmatrix} l_1 \\ l_2 \\ l_3 \end{pmatrix} \delta = 0$$

Où $N = \begin{pmatrix} u_1 \\ v_1 \\ 1 \end{pmatrix} \wedge \begin{pmatrix} u_2 \\ v_2 \\ 1 \end{pmatrix}$ est la normale au plan d'interprétation du segment 3D, défini comme le plan passant par le point focal et le segment image [LH89].

En conséquence, *l'observation d'au moins 9 segments de position inconnue dans l'espace tridimensionnel, mais de direction connue dans un repère absolu, permet de calibrer la caméra*. La solution est toujours analytique, puisque (10) est linéaire en fonction des éléments de M . Les éléments de la sous-matrice $\begin{pmatrix} l_1 \\ l_2 \\ l_3 \end{pmatrix}$ de M sont la

solution d'un système linéaire sous contrainte ($\|l_3\| = 1$). Une fois ces derniers obtenus, les équations (7) (8) et (9) associées au segment j dans l'espace 3D, choisi comme référence ($X_0 = 0$) fournissent un système qui permet le calcul des $l_{i,4}$:

$$\begin{cases} (u_1^j - u_2^j) l_{1,4} = u_1^j (u_2^j l_3 - l_1) \cdot \delta \\ (v_1^j - v_2^j) l_{2,4} = v_1^j (v_2^j l_3 - l_2) \cdot \delta \\ (v_1^j - v_2^j) l_{3,4} = (v_2^j l_3 - l_2) \cdot \delta \end{cases} \quad (11)$$

La matrice M étant estimée, on peut alors extraire les paramètres de la calibration [Tos87]. De plus, on peut calculer la position de chaque segment 3D, dans le repère absolu, grâce aux équations (7) (8) et (9), par exemple.

Il est remarquable que dans l'équation (10) la longueur du segment 3D n'ait pas besoin d'être connue, seule la direction compte. On n'a donc pas besoin des longueurs pour obtenir la partie (l_1, l_2, l_3) de la matrice perspective M . On établit ainsi le lien avec certaines approches à base de lignes, notamment le problème perspective à n lignes [LH89] et la calibration à base de lignes [Vai90]. Néanmoins, pour entièrement déterminer M , il est nécessaire de connaître la longueur des segments 3D (11).

Afin d'estimer les paramètres intrinsèques de la caméra, ainsi que la position et l'orientation

de la caméra, il est nécessaire, du fait de l'algorithme utilisé, de connaître l'orientation de 9 segments dans l'espace 3D, ainsi que leur longueur. La caméra est alors localisée par rapport au référentiel absolu utilisé pour repérer les orientations des différents segments. Son origine est le segment j choisi comme référence. Le choix de primitives plus complexes, comme des assemblages de segments, permet de diminuer le nombre de primitives à utiliser au moment de la phase de calibration. Cela est abordé dans la partie suivante.

4 Segments jointifs

Lorsque deux segments dans l'espace ont un sommet en commun, on peut en déduire une relation supplémentaire. Supposons que chaque segment 3D i ($1 \leq i \leq N$) soit décrit par le bipoint $(X_0, X_0 + \Delta_i)$ (tous ont en commun le premier sommet X_0). Pour le premier segment 3D ($i = 1$), on obtient les relations (6) (7) (8) et (9) où Δ est remplacé par Δ_1 . Pour un segment 3D différent, on a, en substituant (1) avec les équations précédentes :

$$\begin{aligned} (u_i - u_1) (u_2 l_3 - l_1) \cdot \delta_1 + \\ (u_1 - u_2) (u_i l_3 - l_1) \cdot \delta_i = 0 \end{aligned} \quad (12)$$

$$\begin{aligned} (v_i - v_1) (v_2 l_3 - l_2) \cdot \delta_1 + \\ (v_1 - v_2) (v_i l_3 - l_2) \cdot \delta_i = 0 \end{aligned} \quad (13)$$

Un ensemble de $(N + 1)$ points dans l'espace, décrit comme un ensemble de N segments 3D présentant un sommet en commun, conduit à $(2N - 1)$ équations. Les équations pour le cas du rectangle seront détaillées dans la partie 7.2. En conséquence, *avec des primitives tridimensionnelles à $N = 2$ (triangle), 3 (rectangle, tétraèdre), 4 et 5 segments, il faudra utiliser respectivement 3, 2, 2 et 1 primitives au minimum, afin d'obtenir un système d'équations cohérent.* Dans le dernier cas et pour des ensembles de segments plus importants, on retrouve la situation classique d'un ensemble de points 3D (au moins 6) connus dans un même repère. Pour chaque cas, nous avons obtenu des équations linéaires en fonction des éléments de la matrice perspective M (10) (12) (13). On peut donc combiner les diverses primitives tridimensionnelles et utiliser dans une même phase de calibration des segments, des triangles, des rectangles et des tétraèdres, entre autres. Néanmoins, le choix des

positions relatives des primitives tridimensionnelles doit être mené avec précaution, car certaines configurations spatiales conduisent à un système d'équations dégénéré. L'algorithme utilisé n'est alors plus applicable.

5 Algorithme de résolution

Nous rappelons dans cette partie, l'algorithme de résolution utilisé. Pour plus de détails se reporter à [Tos87], où l'algorithme est employé pour réaliser la calibration avec des points. Elle permet de résoudre un système d'équations linéaires au sens des moindres carrés, sous une contrainte de norme.

De par notre approche, le problème s'explique comme un système linéaire en $\begin{pmatrix} l_1 \\ l_2 \\ l_3 \end{pmatrix}$ à résoudre au sens des moindres carrés, sous contrainte :

$$\min_L \|AL\|$$

avec

$$L = \begin{pmatrix} l_1 \\ l_2 \\ l_3 \end{pmatrix}$$

et la contrainte

$$\|l_3\| = 1$$

où A est constituée de lignes formées à partir des équations linéaires du type de (10), (12) ou (13).

En utilisant les multiplicateurs de Lagrange, on déduit la solution analytique suivante :

$$l_{min} = \begin{pmatrix} -B_{6 \times 6}^{-1} C_{6 \times 3} l_{3min} \\ l_{3min} \end{pmatrix} \quad (14)$$

où l_{3min} est le vecteur propre associé à la valeur propre minimum de la matrice $D_{3 \times 3} - C_{3 \times 6}^t B_{6 \times 6}^{-1} C_{3 \times 6}$, avec

$$A^t A = \begin{pmatrix} B_{6 \times 6} & C_{3 \times 6} \\ C_{3 \times 6}^t & D_{3 \times 3} \end{pmatrix}$$

Cette méthode de résolution présente deux avantages :

- la résolution est analytique, à partir d'équations linéaires,
- elle peut traiter les cas où l'information est redondante, ce qui permet d'améliorer la précision du résultat en utilisant plus de primitives que le nombre minimum nécessaire à l'algorithme.

En revanche, comme pour des raisons de simplicité on ne manipule que des équations linéaires, le nombre minimal d'informations pour l'algorithme est un peu supérieur au nombre minimal théorique. Par exemple, utilisé avec des segments, cet algorithme requiert 9 segments, alors que théoriquement, 7 ou 8 segments peuvent suffire.

6 Existence d'une solution

Il existe des configurations spatiales qui rendent la résolution impossible par la méthode présentée. Il serait donc intéressant de les caractériser afin de les éviter. Ainsi il est bien connu que la calibration avec des points nécessite une distribution de points 3D non coplanaires. De même, pour la calibration à base de primitives tridimensionnelles, ces dernières, ramenées à une même origine ne doivent pas être coplanaires. Plus généralement, toute configuration qui fait de B une matrice non inversible provoque la dégénérescence du système. Malheureusement, ces configurations sont difficiles à caractériser ou à énumérer de façon exhaustive, et cela reste en soi un sujet à étudier.

Nous avons toutefois observé un autre type de configuration qui produit la dégénérescence du système, ici liée à la méthode de résolution. En particulier, cette situation apparaît lorsque les primitives à base de segments 3D, ramenés à une même origine, sont incluses dans un trièdre orthogonal. De façon analogue, la méthode fondée sur les points avec la contrainte $\|l_{3min}\| = 1$ dégénère avec l'algorithme de résolution de la partie précédente, si les points sont positionnés sur un trièdre. Ce type de dégénérescence est la conséquence du fait qu'une des contraintes sur la transformation perspective n'est pas utilisée : les éléments de la matrice perspective M ne sont plus contraints. Pour pallier cela, on peut employer une contrainte, posée a priori, qui agit sur l'ensemble de la matrice, comme par exemple $\|M\| = 1$.

Une fois la calibration réalisée sur plusieurs vues, à partir de quelques objets de la scène, il est possible de construire interactivement le modèle géométrique de l'ensemble de la scène par superposition sur chaque image.

7 Application à la construction interactive d'une scène

7.1 Construction interactive

Il s'agit de construire de manière interactive un objet d'une scène à partir de différentes images de cette scène [JP92]. Après la détermination de la matrice de transformation perspective de la caméra par calibration, l'utilisation d'un modeleur comme outil de modélisation géométrique, à travers une vue identique à celle de la caméra, nous permet de superposer la projection de l'objet tridimensionnel à construire sur son image. Le modeleur est ainsi utilisé comme un calque sur l'image qui est placée dans une vue dite caméra.

Nous disposons, comme capteur d'images, d'une caméra DTC couleurs Sony. Nous utilisons comme modeleur géométrique, Action 3D, créé en collaboration avec la société Sogitec. Pour le moment, nous nous sommes intéressés à une scène de bureau, qui est un univers polyédrique. Ce modeleur permet, pour notre application, de créer des modèles en fil de fer avec des primitives à déformer (cube, sphère, cônes par exemple). On peut faire subir trois types de transformations à une primitive : translation, rotation et affinité, sur les trois axes : x , y , z . L'utilisateur doit donc gérer interactivement 9 degrés de liberté. Pour éviter des erreurs de construction due à la profondeur (Cf. figure 12), chaque nouvelle primitive insérée dans la scène possède au moins un point de contact avec un objet correctement positionné. En pratique, les objets ont au moins une face commune (une fonction a été implémentée afin de satisfaire cette contrainte), ce qui réduit le nombre de degrés de liberté (6 : 2 pour la translation, 1 pour la rotation et 3 pour l'affinité) et permet la construction incrémentale de ce qui est perçu par le capteur.

La représentation des modèles géométriques dans le modeleur est une représentation par bords. La structure est hiérarchique : les volumes sont définis comme des polyèdres, définis eux-mêmes par un ensemble de polygones orientés fermant un volume. Il est à noter que le modeleur peut créer des primitives plus complexes, ce qui nous permettrait de construire une scène plus complexe qu'une scène de bureau.

La construction étant interactive, l'opérateur

peut très facilement visualiser et donc vérifier que les primitives qu'il construit sont définies avec cohérence (pas d'interpénétration ou pas d'objet flottant par exemple). De plus, grâce aux fonctionnalités du modèleur tels que le zoom ou la multiplicité des vues, la construction interactive est rendue plus facile.

La précision de la modélisation est également laissée au choix de l'utilisateur, en fonction de l'application qu'il veut en faire. En effet, dans le cadre du déplacement d'un robot, où il doit éviter les obstacles, la connaissance du volume englobant et de la position de l'objet est suffisante. Il n'est pas nécessaire d'avoir plus de précision. Par contre, pour d'autres applications, comme l'architecture, il serait nécessaire, d'avoir une description plus détaillée.

7.2 Résultats de calibration

Nous avons testé la calibration avec les primitives suivantes : des points, des segments, des rectangles et une combinaison de segments et de rectangles. Dans un premier temps, ces tests ont été réalisés sur le minimum possible de primitives : 6 pour les points, 9 pour les segments, 2 pour les rectangles et pour le cas mixte: 1 rectangle et 4 segments. Nous avons pris soin d'éviter les configurations qui posent problème : les points tridimensionnels ne doivent pas être coplanaires et ne pas reposer sur les axes d'un trièdre orthogonal.

fichier 1	fichier 2
(582,274)	(583,273)
(513,301)	(512,301)
(183,384)	(182,385)
(184,268)	(181,267)
(502,451)	(501,450)
(509,306)	(508,306)

TAB. 1 - Points 2D exprimés en pixels pour la calibration avec 2 rectangles : sur chaque point une différence de quelques pixels Cf. Figures 6 et 7 et tableaux 2 et 3

Les coordonnées dans le repère de l'image sont obtenues en cliquant les points à l'écran tandis qu'il faut mesurer leur correspondant 3D dans le repère de la scène. La précision des mesures 2D est donc au pixel près, ce qui est assez faible et fait varier énormément les valeurs des para-

mètres intrinsèques obtenues (Cf. tableaux 1, 2 et 3 et figures 6 et 7).

Pour les rectangles, de la même manière que nous obtenons l'équation (10) pour un segment et à partir des équations (12) et (13), nous avons le système d'équations suivant :

$$\left\{ \begin{array}{l} l_1.\delta_1(v_1 - v_2) + l_2.\delta_1(u_2 - u_1) + \\ \quad l_3.\delta_1(u_1v_2 - v_1u_2) = 0 \quad (10) \\ l_1.[\delta_1(u_1 - u_3) + \delta_2(u_2 - u_1)] + \\ l_3.[\delta_1(u_3 - u_1)u_2 + \delta_2(u_1 - u_2)u_3] = 0 \quad (12, i=3) \\ l_1.\delta_1(v_1 - v_3) + l_2.\delta_2(u_2 - u_1) + \\ l_3.[\delta_1(v_3 - v_1)u_2 + \delta_2(u_1 - u_2)v_3] = 0 \quad (13, i=3) \\ l_1.[\delta_1(u_2 - u_4) + \delta_2(u_2 - u_1)] + \\ l_3.[\delta_1(u_4 - u_2)u_1 + \delta_2(u_1 - u_2)u_4] = 0 \quad (12, i=4) \\ l_1.\delta_1(v_1 - v_4) + l_2.(\delta_1 + \delta_2)(u_2 - u_1) + \\ l_3.[\delta_1(u_1v_4 - v_1u_2) + \delta_2(u_1 - u_2)v_4] = 0 \quad (13, i=4) \end{array} \right. \quad (15)$$

où δ_1 et δ_2 représentent les vecteurs largeur et longueur du rectangle considéré.

La calibration avec pour base le rectangle nécessite au moins 2 primitives non coplanaires, et dans nos tests, pour des raisons de simplification de mesures, les deux rectangles considérés forment un trièdre orthogonal. Le système d'équations de la calibration avec pour primitives le segment et le rectangle est composé de l'équation (10) (segment) et du système d'équations (15) (rectangle).

Pour pouvoir comparer les résultats de calibration des différentes primitives nous avons essayé de prendre autant que possible, les mêmes points sur une image. Nous avons ainsi choisi les parties visibles du bureau comme primitives (coins, arêtes, plateau etc. Cf. figure 2).

Les résultats obtenus avec différentes primitives sont relativement similaires (Cf. tableaux 2 et 3). Le choix d'une primitive à la place d'une autre dépend uniquement des informations que l'on peut avoir de la primitive : ses points-image et ses mesures tridimensionnelles. En effet, il est plus simple de mesurer la longueur d'un segment de direction particulière que de mesurer des points situés sur des objets dont les repères locaux sont différents. Cependant, il faut aussi s'assurer de l'existence de leur correspondant sur l'image. De plus, nous avons observé le fait suivant : si les primitives choisies ne sont pas bien réparties sur l'image, la méthode ne donne pas toujours de résultats. L'estimation est meilleure avec une bonne répartition sur l'image, des primitives (Cf. Figures 8 et 9). Par ailleurs, si nous augmentons leur nombre, les résultats s'améliorent (Cf. Figures 4 et 10). En repérant

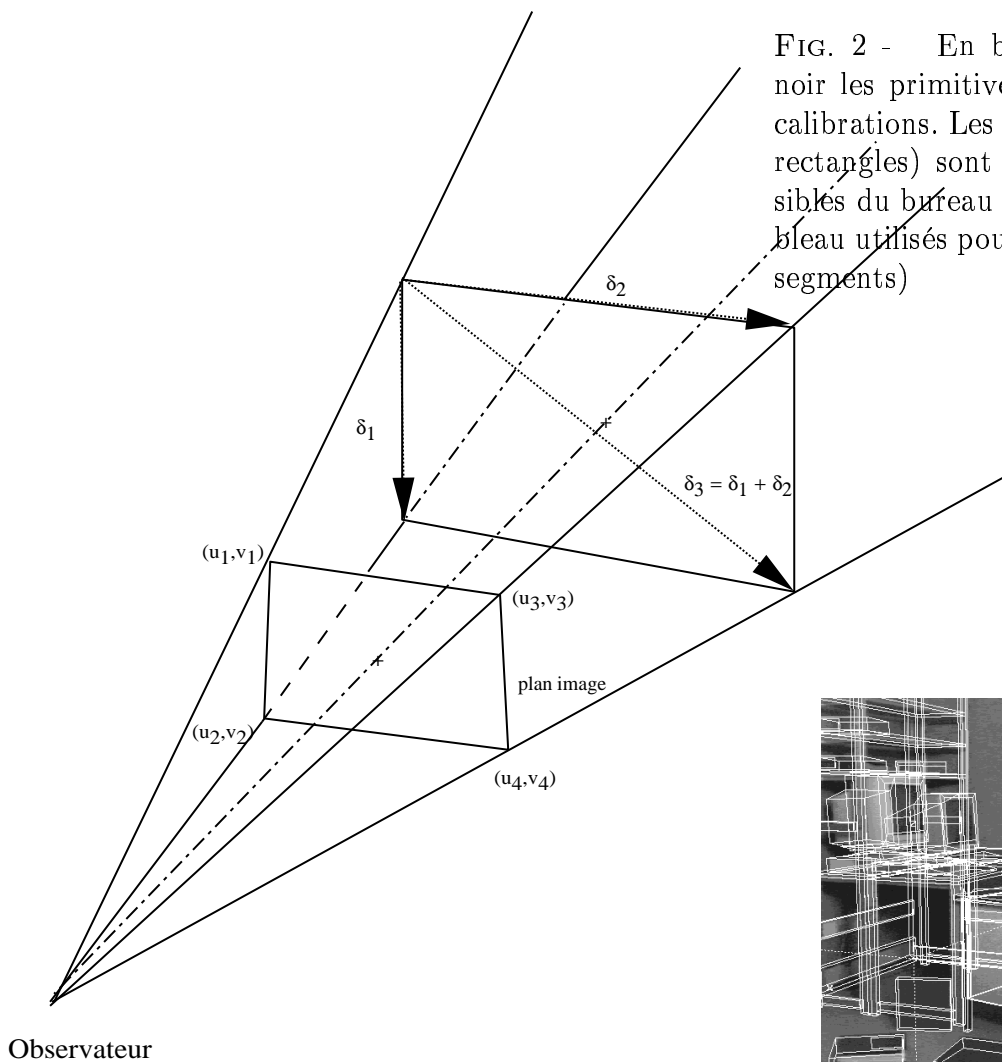


FIG. 1 - Les données d'un rectangle nécessaires à la calibration

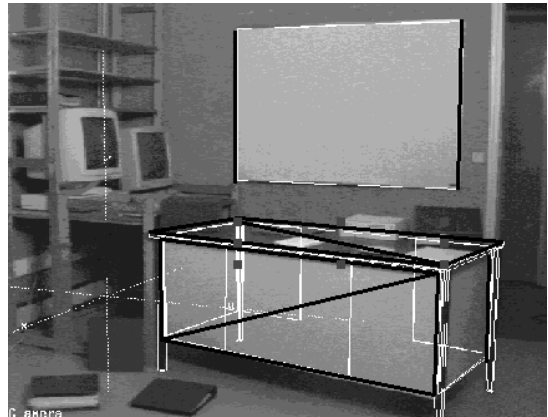


FIG. 2 - En blanc le modèle reconstruit, en noir les primitives utilisées lors des différentes calibrations. Les primitives (points, segments et rectangles) sont essentiellement les parties visibles du bureau et les 2 arêtes verticales du tableau utilisés pour les calibrations mixte et avec segments)

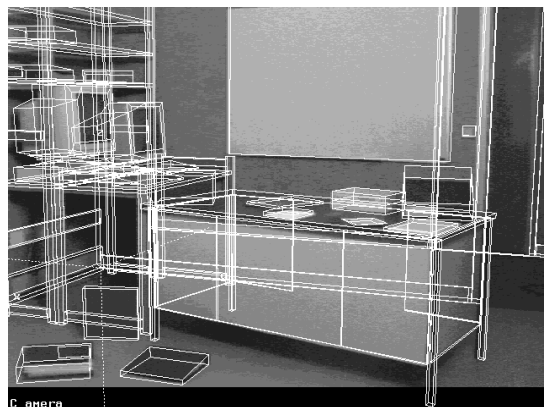


FIG. 3 - Caméra calibrée avec 6 points. Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.

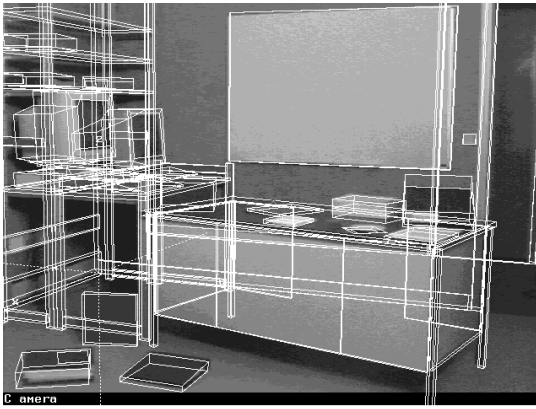


FIG. 4 - Caméra calibrée avec 8 points. Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.

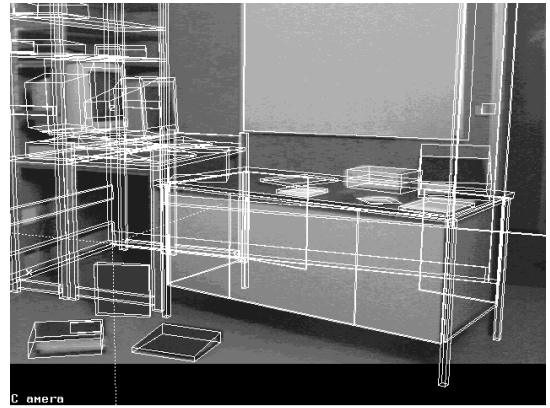


FIG. 7 - Caméra calibrée avec 2 rectangles avec un deuxième fichier 2D. Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.

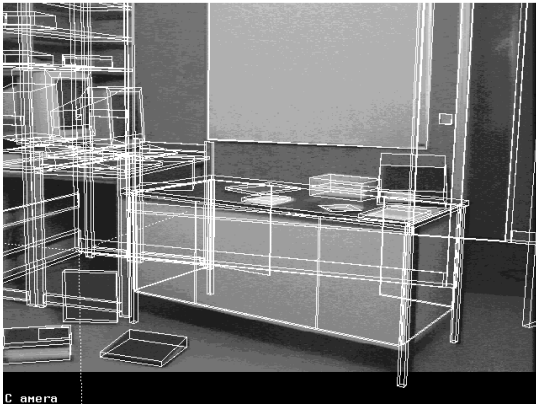


FIG. 5 - Caméra calibrée avec 9 segments. Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.

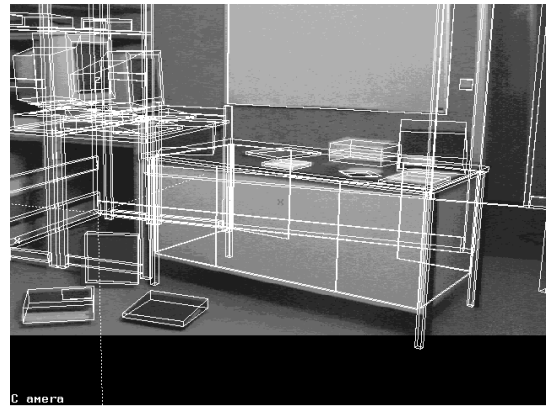


FIG. 8 - Caméra calibrée avec 1 rectangle (plateau) et 4 segments (appartenant à la face avant du bureau). Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.

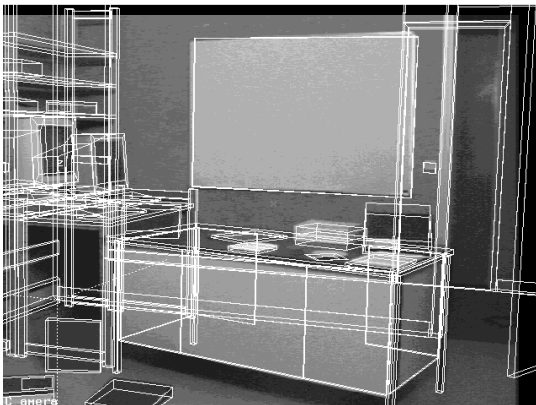


FIG. 6 - Caméra calibrée avec 2 rectangles : le plateau et la face avant du bureau (Premier fichier 2D). Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.

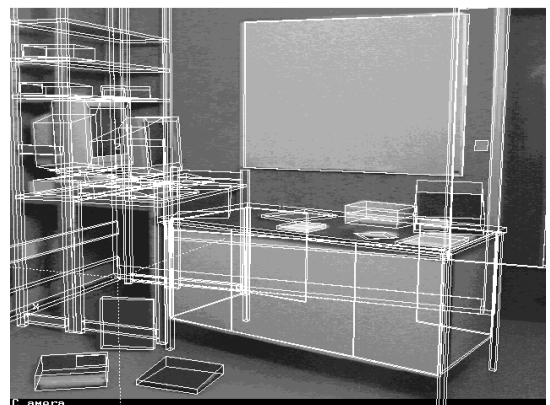


FIG. 9 - Caméra calibrée avec 1 rectangle (plateau) et 4 segments (dont 2 sont les arêtes verticales du tableau). Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra .

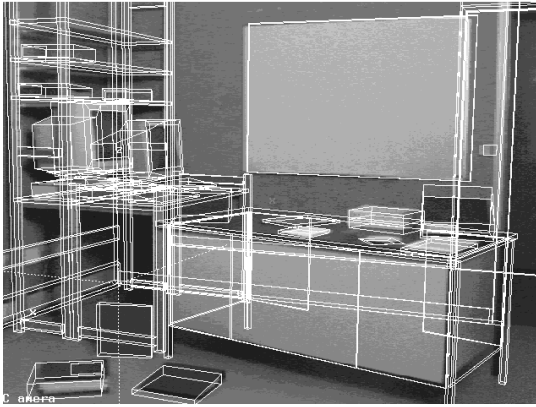


FIG. 10 - Caméra calibrée avec 1 rectangle et 5 segments dont 4 sont verticaux. Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.

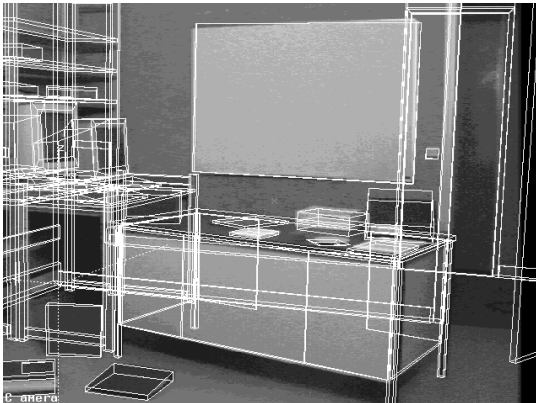


FIG. 11 - Caméra calibrée avec 2 rectangles et 1 segment vertical. Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.

dans l'image deux rectangles formant un trièdre orthogonal dans l'espace 3D, nous choisirons de préférence la calibration mixte car elle offre la possibilité de rajouter des segments de même direction, par exemple verticaux, améliorant ainsi l'estimation des paramètres de la caméra.

paramètres	référence	6 points (Figure 3)	9 segments (Figure 5)	2 rectangles (Fig.6, Tab.1 Fich.1)	1 rectangle 4 segments (Fig.9)
(u_0, v_0) (pixels)	(318,238)	(325,266)	(344,278)	(363,225)	(312,256)
(a_u, a_v)	(871,874)	(872,883)	(900,903)	(950,939)	(886, 877)
tx (dm)	49.34	48.35	49.68	51.57	48.83
ty (dm)	31.25	331.08	31.58	31.72	31.37
tz (dm)	12.8	12.56	12.92	13.27	12.86
rx (degrés)	-0.64	-0.56	-1.15	-1.57	-0.59
ry (degrés)	6.31	7.99	8.93	5.7	7.36
rz (degrés)	-160.55	-160.71	-161.94	-163.50	-156.71

TAB. 2 - Paramètres de la caméra obtenus par calibration avec le nombre minimum de primitives: centre de l'image (u_0, v_0) , facteurs d'échelles (a_u, a_v) , translations (tx, ty, tz), rotation (rx, ry, rz)

paramètres	référence	8 points (Figure 3)	2 rectangles (Figure 7) (Tab.1 Fich.2)	1 rectangle 4 segments (Figure 8)	1 rectangle 5 segments (Figure 10)	2 rectangles 1 segment (Figure 9)
(u_0, v_0) (pixels)	(318,238)	(320,254)	(311,288)	(331,322)	(297,240)	(362,239)
(a_u, a_v)	(871,874)	(877,889)	(881,898)	(924, 909)	(850,856)	(949,940)
tx (dm)	49.34	48.74	48.68	50.40	47.48	51.49
ty (dm)	31.25	31.25	31.27	31.54	31.06	31.67
tz (dm)	12.8	12.8	12.69	13.14	12.5	13.27
rx (degrés)	-0.64	-0.71	-0.92	-1.47	-0.82	-1.72
ry (degrés)	6.31	7.36	9.48	11.80	6.32	6.55
rz (degrés)	-160.55	-160.55	-159.72	-161.42	-158.53	-163.49

TAB. 3 - Paramètres de la caméra obtenus par calibration sur d'autres données

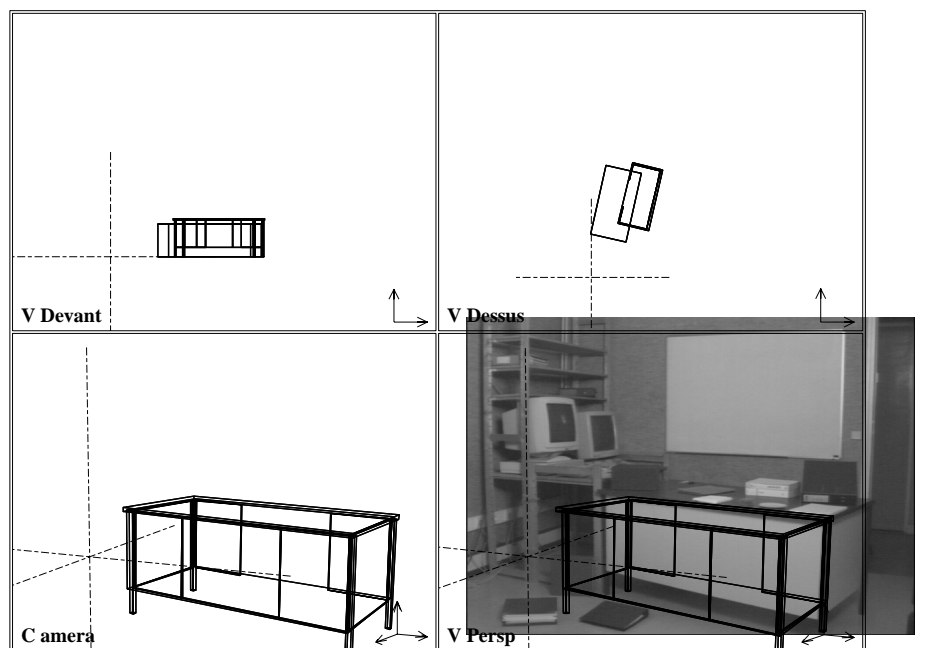


FIG. 12 - Phase de construction: erreur de construction en profondeur sur le corps du bureau : sa position et ses dimensions paraissent correctes sur les vue caméra et perspective alors qu'elles sont fausses

A partir d'une image de scène réelle, les résultats de la calibration sont essentiellement évalués par la qualité de la superposition des primitives utilisées sur l'image ainsi que les coordonnées du centre de l'image. Cette évaluation manque de précision. Les tests des calibrations ont pu se faire sans problème sur des images de synthèse et pour la scène présentée, nous avons au départ, des informations précises sur les paramètres de la caméra.

8 Conclusion

Nous avons proposé une méthode de calibration adaptée au besoins d'un système interactif de construction de modèles tridimensionnels, à partir de différentes vues d'une scène réelle. Cette méthode ne manipule que des équations linéaires. L'algorithme utilisé permet d'obtenir la solution d'un système linéaire, au sens des moindres carrés, sous une contrainte de norme. Cette approche a l'avantage de permettre l'utilisation combinée de différentes primitives polyédriques. Pour l'ensemble des primitives, il faut connaître la taille et l'orientation dans le même repère, des segments dont elles sont constituées. En revanche, il n'est pas nécessaire d'avoir les positions relatives de ces primitives. Dans la pratique, les orientations des segments sont choisies perpendiculaires les unes par rapport aux autres, ainsi seule leur longueur est à mesurer.

Nous avons observé que certaines configurations en orientation des primitives tridimensionnelles, conduisent à un système d'équations dégénéré. Ces configurations sont difficiles à énumérer de façon exhaustive, cela reste donc un sujet d'étude ouvert.

Présentant de nombreux avantages, la construction interactive à l'aide du modéleur ne suffit pas. Pour avoir une modélisation correcte des objets, un minimum de connaissance concernant les dimensions des objets sont nécessaires pour corriger les incertitudes des mesures 2D de la calibration ainsi que de la construction. En particulier, il faudrait un système de contrôle des résultats de la calibration. En effet, les résultats de la calibration sont essentiellement évalués par la qualité visuelle de la superposition des primitives utilisées sur l'image ainsi que les coordonnées du centre de l'image. Cette évaluation reste grossière et très subjective. La modélisation ne peut être fiable, si la calibration n'est

pas correcte.

Dans notre étude, nous nous sommes surtout intéressés à la modélisation géométrique, mais il est également tout à fait possible, de définir des informations autres propres à l'objet, telles sa couleur ou sa texture, etc.

9 Remerciements

Nous voudrions remercier chaleureusement Véronique Serfaty, qui a construit le modèle tridimensionnel de la scène qui illustre cet article, nous permettant ainsi d'effectuer nos tests de calibration plus facilement.

Références

- [CRF92] C.I. Marinou, C.A. Rothwell, A. Zisserman and D.A. Forsyth. Relative motion and pose from arbitrary plane curves. *Image and Vision Computing*, 10(4):250–262, 1992.
- [CT86] H.L. Chou and W.H. Tsai. A new approach to robot location by house corners. *Pattern Recognition*, 19(6):439–451, 1986.
- [CT90] B. Caprile and V. Torre. Using vanishing points for camera calibration. *International Journal of Computer Vision*, 4:127–140, 1990.
- [Har80] R.M. Haralick. Using perspective transformations in scenes analysis. *Computer Graphics and Image Processing*, 13:191–221, 1980.
- [Hor89] R. Horaud. An analytic solution for the p-4p problem. *Computer Vision, Graphics and Image Processing, Academic Press*, 1989.
- [JP92] P. Jaillon and B. Peroche. Aide à la saisie d'objets en synthèse d'images. Technical report, Ecole Nationale Supérieure des Mines de Saint-Etienne, 1992.
- [Kan87] K.I. Kanatani. 3d recovery from angle clues. In *Image Analysis*, pages 167–179, Stockholm, June 1987.

- [LH81] H.C. Longuet-Higgins. An computer algorithm for reconstructing a scene from two projections. *Nature*, september:133–138, 1981.
- [LH89] E. Lutton and H.Maitre. Etude des symétries du problème de perspective à trois lignes. In *7ème congrès AFCET, Reconnaissance des Formes et Intelligence Artificielle*, pages 537–546, Paris, November 1989.
- [LT88] R.K. Lenz and R.Y. Tsai. Techniques for calibration of scale factor and image center for high accuracy 3-d machine vision metrology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(5):713–720, 1988.
- [Ma93] S. De Ma. A complete two-plane camera calibration method and experimental comparisons. In *IEEE International Conference on Computer Vision*, pages 439–446, Berlin, March 1993.
- [MDR88] J.T. Lapreste M. Dhome, M. Richetin and G. Rives. The inverse perspective problem from a single view for polyhedra location. In *IEEE Computer Vision and Pattern Recognition*, pages 61–66, Ann Arbour, June 1988.
- [MF92] S.J. Maybank and O.D. Faugeras. A theory of self-calibration of a moving camera. *International Journal of computer Vision*, 8(2):123–151, 1992.
- [MM91] R. Mohr and L. Morin. Relative positioning from geometric invariants. In *IEEE Computer Vision and Pattern Recognition*, pages 139–144, Hawaii, June 1991.
- [OF87] G. Toscani O. Faugeras, F. Lustman. Calcul du mouvement et de la structure à partir de points et de droites. In *6ème congrès AFCET, Reconnaissance des Formes et Intelligence Artificielle*, pages 75–89, Antibes, November 1987.
- [Tos87] G. Toscani. *Systèmes de calibration et perception du mouvement en Vision Artificielle*. PhD thesis, Université Paris-Sud, 1987.
- [TPP93] A. Yassine T.Q. Phong, R. Horaud and D.T. Pham. Optimal estimation of object pose from a single perspective view. In *IEEE International Conference on Computer Vision*, pages 439–446, Berlin, March 1993.
- [Tsa86] R.Y. Tsai. : An efficient and accurate camera calibration technique for 3d machine vision. In *IEEE Computer Vision and Pattern Recognition*, pages 364–374, Miami beach, Florida, June 1986.
- [Vai90] R. Vaillant. *Géométrie différentielle et vision par ordinateur: détection et reconstruction des contours d'occlusion de la surface d'un objet non-polyédrique*. PhD thesis, Université Paris-Sud, 1990.