

Une solution garantie pour l'autocalibrage plan

A guaranteed solution to plane-based self-calibration

Benoît Bocquillon

Pierre Gurdjos

Alain Crouzil

IRIT, Toulouse, France

bocquillon@irit.fr

gurdjos@irit.fr

crouzil@irit.fr

Résumé

Nous nous intéressons à l'autocalibrage d'une caméra, à partir de plusieurs images d'une scène plane. Par autocalibrage, nous faisons référence au problème de l'estimation simultanée des paramètres intrinsèques de la caméra et de la structure euclidienne d'un plan 3D, tel que Triggs l'a formulé en 1998 [17]. Une solution est généralement obtenue en résolvant un système d'équations non linéaires grâce à des méthodes d'optimisation locale. L'initialisation des paramètres à estimer et, en particulier, de la distance focale, a un impact important sur la solution. En rappelant le fait que ces cinq paramètres sont interdépendants, nous proposons une formulation différente, avec seulement trois degrés de liberté, correspondant à trois inconnues. Vu le nombre restreint de paramètres à estimer, nous nous intéressons à l'optimisation globale afin d'obtenir une solution garantie, avec un temps de calcul raisonnable. Le calcul par intervalles fournit un cadre numérique efficace, qui se révèle être très performant, en termes de précision de l'estimation et de temps de calcul.

Mots Clef

Autocalibrage plan, rectification euclidienne, homographie, conique absolue, points cycliques.

Abstract

We investigate the problem of self-calibrating a camera, from multiple views of a planar scene. By self-calibrating, we refer to the problem of simultaneously estimate the camera intrinsic parameters and the Euclidean structure of one 3D plane, as stated in [17]. A solution is usually obtained by solving a non-linear system via local optimization, with the critical issue of parameter initialization, especially the focal length. Arguing that these five parameters are inter-dependent, we propose an alternate problem formulation, with only three d.o.f., corresponding to three parameters to estimate. In the light of this, we are concerned with global optimization in order to get a guaranteed solution, with the shortest response time. Interval analysis provides an efficient numerical framework, that reveals to

be highly performant, with regard to both estimation accuracy and time-consuming.

Keywords

Plane-based self-calibration, metric rectification, homography, absolute conic, circular points.

1 Introduction

L'autocalibrage consiste à déterminer, partiellement ou complètement, les propriétés euclidiennes d'une caméra et/ou de la scène, à partir d'une séquence d'images acquises par une caméra non calibrée. Le principe de l'autocalibrage est d'utiliser des «entités absolues» comme cibles, contraintes géométriquement par des informations *a priori* sur les paramètres internes ou externes de la caméra. Ces cibles sont des entités abstraites, localisées à l'infini, qui décrivent la structure euclidienne de l'espace à d dimensions considéré. Elles ont la propriété caractéristique d'être invariantes par les similitudes¹ de cet espace [5, 14]. En 3D, la cible est la *conique absolue* (CA), qui est un cercle de rayon imaginaire sur le plan à l'infini π_∞ . La CA a la propriété bien connue que son image (ICA) ne dépend pas du mouvement de la caméra, mais seulement des paramètres internes de la caméra. C'est le point de départ de nombreuses méthodes d'autocalibrage 3D (voir [5, chapitre 19] pour plus de détails). À partir d'une reconstruction projective de la scène, l'autocalibrage 3D détermine la structure euclidienne de l'espace 3D en spécifiant la CA et le plan à l'infini. Ceci peut être fait séparément ou simultanément. Dans ce dernier cas, la CA est représentée par une quadrique dégénérée définie tangentiellement, constituée de ses plans tangents, et appelée quadrique absolue duale dans la littérature. En supposant que la distance focale est la seule inconnue, des solutions directes et linéaires peuvent être obtenues, comme par exemple dans [1].

Le problème étudié dans ce travail est l'autocalibrage 2D d'une caméra, c.-à-d. en observant les points d'un plan 3D π , relativement à un mouvement général de caméra. Dans

¹C.-à-d. des transformations préservant les angles et changeant les distances dans le même rapport.

l'espace 2D, les cibles d'autocalibrage sont les *points cycliques* (PC) qui sont deux points complexes conjugués de π sur la droite à l'infini, communs à tous les cercles de π et à la CA de π_∞ . Depuis les travaux de Triggs [17], nous savons que l'autocalibrage 2D est possible, en utilisant la contrainte imposant que les images des PC (IPC) appartiennent à l'ICA. Cette contrainte ne nécessite que la connaissance des homographies inter-vues induites par π . Puisqu'aucune autre invariance (générale) des IPC ne peut être exprimée, peu de méthodes d'autocalibrage 2D ont vu le jour [10, 15, 17], excepté pour des mouvements particuliers de caméra [6, 9]. De plus, contrairement à l'autocalibrage 3D, même avec un modèle simplifié de la caméra, il n'existe pas de solution directe ou linéaire. Un tel problème, consistant à déterminer simultanément les PC et la CA, est non linéaire par essence. Comme précisé dans [17], le paramétrage du problème nécessite la prise en compte de quatre degrés de liberté pour les IPC plus cinq degrés de liberté pour la CA. Une solution peut être obtenue par optimisation locale, à partir d'au moins cinq vues. L'initialisation des paramètres et en particulier de la distance focale, a un impact important sur la solution.

Notre idée initiale est de réduire le nombre de paramètres à estimer en utilisant le fait que, puisque les PC appartiennent à la CA, il y a une redondance dans le paramétrage du problème. L'interdépendance des paramètres dans la formulation de Triggs est une contrainte théorique du modèle qui n'a pas de raison de ne pas être exactement satisfaite. En fait, Triggs a traité cette contrainte comme une équation, ce qui ne se justifie pas vraiment, comme nous le verrons plus tard. Ceci étant dit, notre contribution est de proposer un nouveau paramétrage minimal du problème de l'autocalibrage 2D, en introduisant comme cible la «conique duale aux points cycliques», c.-à-d., l'ensemble des droites isotropes² de π , dont l'enveloppe coïncide avec les PC. Grâce à nos propositions 1 et 2, nous proposons une décomposition de cette matrice, fondée sur le fait que la droite à l'infini intersecte la CA en les PC. Nous montrons donc que les seules inconnues du problème sont les paramètres de la structure affine du plan et les paramètres internes de la caméra. Ceci conduit à une formulation avec sept inconnues/degrés de liberté au lieu des neuf introduits par Triggs dans [17]. En supposant que la distance focale est la seule inconnue, trois paramètres seulement doivent être estimés. Ce faible nombre d'inconnues étant bien adapté au calcul par intervalles, ceci nous permet d'envisager l'obtention d'une solution garantie [13]. Le calcul par intervalles a été largement utilisé dans les problèmes d'optimisation globale [4] et offre la garantie que le minimum global a été trouvé. Le calcul par intervalles a été appliqué avec succès à l'autocalibrage 3D [2]. Il fournit un cadre numérique efficace, qui se révèle très performant, en termes de précision d'estimation et de temps de calcul.

Cet article est organisé de la manière suivante. Tout

²On appelle droite isotrope toute droite ayant pour point à l'infini l'un des PC.

d'abord, en partant des équations de base de l'autocalibrage 2D de [17], nous expliquons comment obtenir un paramétrage minimal du problème. Nous calculons une fonction de coût à partir de ce paramétrage. Ensuite, nous rappelons les règles principales du calcul par intervalles et nous exposons l'algorithme d'optimisation globale que nous avons utilisé. Enfin, nous donnons les résultats obtenus avec des images simulées et des images réelles puis nous terminons par la conclusion.

2 Paramétrage minimal de l'autocalibrage 2D

Avant-propos. Le problème que nous étudions consiste à retrouver la structure euclidienne d'un plan 3D π , appelé plan de la scène, observé dans plusieurs vues, relativement à une caméra *non calibrée*. Nous supposons que les seules données du problème sont les homographies inter-vues induites par π .

Sans autre information, ce problème ne peut pas être séparé de celui consistant à calibrer la caméra, c.-à-d. à retrouver ses paramètres intrinsèques. Ces deux problèmes sont regroupés sous le nom d'autocalibrage plan [17]. Dans [10], Malis décrit une alternative à [17] ; nous montrerons en §2.1 le lien entre ces deux contraintes.

Notations. Nous utilisons quelques notations issues de MATLAB. La notation $1 : n$ décrit l'intervalle $1, \dots, n$. $M_{(1:r, 1:c)}$ décrit la sous-matrice (r, c) de M sélectionnée par l'intervalle des lignes $1 : r$ et l'intervalle des colonnes $1 : c$. La notation $M_{(:, 1:c)}$, resp. $M_{(1:r, :)}$, sélectionne les c , resp. r , premières colonnes, resp. lignes, de M . Nous définissons aussi les vecteurs canoniques :

$$\mathbf{e}_1 \equiv (1, 0, 0)^\top, \quad \mathbf{e}_2 \equiv (0, 1, 0)^\top, \quad \mathbf{e}_3 \equiv (0, 0, 1)^\top. \quad (1)$$

La matrice $[\mathbf{x}]_\wedge$ fait référence à la matrice antisymétrique, d'ordre 3, telle que $[\mathbf{x}]_\wedge \mathbf{y} = \mathbf{x} \wedge \mathbf{y}$, $\mathbf{y} \in \mathbb{R}^3$. Dans cet article, nous utiliserons souvent l'égalité

$$[\mathbf{T}\mathbf{x}]_\wedge = \det(\mathbf{T})\mathbf{T}^{-\top} [\mathbf{x}]_\wedge \mathbf{T}^{-1}. \quad (2)$$

La notation i fait *toujours* référence au nombre imaginaire $\sqrt{-1}$.

Dans la suite, nous supposons que certains résultats de base en géométrie projective sont connus. Ceux-ci peuvent être trouvés dans des livres traitant ce sujet comme [5, 14]. Nous rappelons néanmoins quelques notions essentielles et établissons de nouvelles propriétés importantes pour notre travail.

La matrice représentant *l'image de la conique absolue* (ICA) est

$$\boldsymbol{\omega} = \mathbf{K}^{-\top} \mathbf{K}^{-1}, \quad (3)$$

où \mathbf{K} est la matrice de calibrage [5, §5.1] qui contient les paramètres internes de la caméra et qui s'écrit dans sa forme générale :

$$\mathbf{K} \equiv \begin{pmatrix} \alpha_u & \gamma & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{pmatrix}, \quad (4)$$

où α_u, α_v représentent la distance focale en termes de dimension des pixels dans les directions u, v respectivement, (u_0, v_0) sont les coordonnées en pixels du point principal et γ est le «facteur d'obliquité».

Les principales entités géométriques de cet article sont rassemblées dans le tableau 1.

SYMBOLES	ENTITÉS GÉOMÉTRIQUES
π, π_∞	Plan de la scène et plan à l'infini.
\mathbf{n}	Normale au plan π
l_∞, \mathbf{v}	Droite à l'infini, intersection de π et π_∞ et son image, la ligne de fuite.
\mathbf{K}	Matrice de calibrage.
Ω, ω	Conique absolue (CA) de π_∞ et son image (ICA).
$\mathbf{I}_\pm, \mathbf{x}_\pm, \mathbf{x}_1, \mathbf{x}_2$	Points cycliques (PC), intersections de Ω et l_∞ , leurs images (IPC) et les parties réelles et imaginaires des IPC.
\mathbf{C}, \mathbf{C}^*	Conique duale aux PC (CDPC), enveloppe constituée de \mathbf{I}_+ et \mathbf{I}_- et son image (ICDPC), enveloppe constituée de \mathbf{x}_+ et \mathbf{x}_- .
H_j	Homographie de la vue clef vers la vue j .

TAB. 1 – Principales entités géométriques.

2.1 Les équations de l'autocalibrage plan

Soit P l'homographie entre le plan (euclidien) de la scène et le plan de l'image, supposée *inconnue*, que nous appellerons *homographie scène-vers-image*. Soit H_j l'homographie inter-vues, supposée *connue*, induite par π , entre la vue courante et la vue j .

Les équations (ponctuelles) de l'autocalibrage plan.

Rigoureusement, on appelle structure euclidienne de π la donnée des *images de ses points cycliques* (IPC), soient $P(\mathbf{I}_\pm)$. Rappelons que les points cycliques (PC) \mathbf{I}_\pm sont, par définition [5, pages 52-53], des points complexes conjugués sur π , communs à tous ses cercles. Dans toute représentation euclidienne, les PC ont des coordonnées homogènes canoniques $\mathbf{e}_\pm \equiv \mathbf{e}_1 \pm i\mathbf{e}_2 = (1, \pm i, 0)^\top$, qui sont invariantes par toute similitude 2D S de π , c.-à-d. $\mathbf{e}_\pm \sim S\mathbf{e}_\pm$, où $S \in \mathbb{R}^{3 \times 3}$ est la matrice de S . Dans le plan de l'image, les coordonnées des IPC $P(\mathbf{I}_\pm)$, notées $\mathbf{x}_\pm \equiv \mathbf{x}_1 \pm i\mathbf{x}_2$, satisfont $\mathbf{x}_\pm \sim P\mathbf{e}_\pm$, où $P \in \mathbb{R}^{3 \times 3}$ est la matrice de P . Notons que \mathbf{x}_\pm a seulement quatre degrés de liberté, à savoir les huit degrés de liberté de P moins les quatre degrés de liberté de S .

Par invariance projective, les IPC sont sur la ligne de fuite, commune à toutes les images des cercles, y compris l'image ω de la conique absolue [5, pages 81-83] du plan à l'infini. L'ICA ω est le lieu de tous les IPC (c.-à-d. de tous les plans 3D) ce qui implique l'égalité $\mathbf{x}_\pm^\top \omega \mathbf{x}_\pm = 0$, ou de manière équivalente (voir [5, page 211]) :

$$\mathbf{x}_1^\top \omega \mathbf{x}_2 = 0 \quad \text{et} \quad \mathbf{x}_1^\top \omega \mathbf{x}_1 - \mathbf{x}_2^\top \omega \mathbf{x}_2 = 0. \quad (5)$$

Dans la vue numéro j , la contrainte est décrite par $\mathbf{x}_\pm^\top H_j^\top \omega_j H_j \mathbf{x}_\pm = 0$, ce qui est équivalent à :

$$\begin{aligned} \mathbf{x}_1^\top H_j^\top \omega_j H_j \mathbf{x}_2 &= 0 \quad \text{et} \\ \mathbf{x}_1^\top H_j^\top \omega_j H_j \mathbf{x}_1 - \mathbf{x}_2^\top H_j^\top \omega_j H_j \mathbf{x}_2 &= 0, \end{aligned} \quad (6)$$

où ω_j est la matrice de l'ICA dans la vue numéro j et H_j est la matrice de H_j .

Les équations duales de l'autocalibrage plan. Une description (peut-être) plus intuitive de la structure euclidienne peut aussi être donnée en termes d'une homographie scène-vers-image $P \circ S$, où S représente une similitude 2D arbitraire. En effet, en appliquant $(P \circ S)^{-1}$ au plan de l'image, nous obtenons une reconstruction euclidienne de π . Une telle homographie $P \circ S$ sera appelée *homographie de rectification*.

Nous allons considérer les IPC comme l'enveloppe d'une conique dégénérée définie tangentiellement. Une telle conique duale est constituée de deux faisceaux de droites imaginaires (qui sont les images des droites isotropes, cf. note de bas de page numéro 2) et de la ligne de fuite. Cette conique, appelée *image de la conique duale aux points cycliques* (ICDPC) dans [5, p.52], a une matrice de la forme :

$$\begin{aligned} \mathbf{C}^* &\sim \mathbf{x}_- \mathbf{x}_+^\top + \mathbf{x}_+ \mathbf{x}_-^\top \\ &\sim P(\mathbf{e}_- \mathbf{e}_+^\top + \mathbf{e}_+ \mathbf{e}_-^\top) P^\top \\ &\sim PS(\mathbf{e}_- \mathbf{e}_+^\top + \mathbf{e}_+ \mathbf{e}_-^\top) S^\top P^\top, \end{aligned} \quad (7)$$

où $S \in \mathbb{R}^{3 \times 3}$ est la matrice de S . Comme $\mathbf{e}_- \mathbf{e}_+^\top + \mathbf{e}_+ \mathbf{e}_-^\top \sim \text{diag}(1, 1, 0)$, une homographie de rectification peut être obtenue à partir de la factorisation adéquate [5, pages 55-56] de \mathbf{C}^* , c.-à-d. fondée sur la décomposition en valeurs singulières (SVD), avec les valeurs singulières $\sigma_1 \geq \sigma_2 > 0$ et $\sigma_3 = 0$:

$$\begin{aligned} \pm \mathbf{C}^* &= \mathbf{U} \Sigma \mathbf{U}^\top \equiv \mathbf{X} \text{diag}(1, 1, 0) \mathbf{X}^\top \\ &= \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 \end{bmatrix}^\top. \end{aligned} \quad (8)$$

C'est pourquoi les IPC peuvent être écrits sous la forme $\mathbf{x}_\pm = \mathbf{U}_{(:,1:2)} \sqrt{\Sigma}_{(1:2,1:2)} \sim \mathbf{X} \mathbf{S} \mathbf{e}_\pm$. Par conséquent, les contraintes (6) peuvent être mises sous la forme matricielle :

$$\begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 \end{bmatrix}^\top H_j^\top \omega_j H_j \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 \end{bmatrix} \sim \mathbf{I}_{2 \times 2}. \quad (9)$$

Nous mettons maintenant en avant une décomposition intéressante de la matrice de l'ICDPC \mathbf{C}^* . Fondamentalement, notre but est de mettre en équation le fait que l'enveloppe de la conique dégénérée \mathbf{C}^* coïncide avec les deux points correspondant à l'intersection de la ligne de fuite \mathbf{v} et de l'ICA ω . Puisque la CA est un cercle sur le plan à l'infini, ces deux points sont les IPC.

Proposition 1 La matrice de l'ICDPC satisfait la décomposition suivante :

$$\mathbf{C}^* \sim [\mathbf{v}]_\wedge \omega [\mathbf{v}]_\wedge, \quad (10)$$

où ω est la matrice de l'ICA et \mathbf{v} est le vecteur représentant la ligne de fuite.

Preuve. Définissons $\Delta \equiv [\mathbf{v}]_{\wedge} \omega [\mathbf{v}]_{\wedge}$. Il est clair que Δ est de rang 2, puisque $[\mathbf{v}]_{\wedge}$ est de rang 2. Δ est donc la matrice d'une conique duale dont l'enveloppe est constituée de deux points distincts \mathbf{p} et \mathbf{q} , c.-à-d. $\Delta \sim \mathbf{p}\mathbf{q}^{\top} + \mathbf{q}\mathbf{p}^{\top}$. Montrons que ces points sont les IPC. D'une part, nous voyons que $\Delta\mathbf{v} = \mathbf{0}$, ce qui implique que \mathbf{p} et \mathbf{q} sont tous les deux sur la ligne de fuite \mathbf{v} . D'autre part, toute droite $\mathbf{w} \neq \mathbf{v}$, vérifiant $\mathbf{w}^{\top}\Delta\mathbf{w} = 0$, passe soit par \mathbf{p} soit par \mathbf{q} . Supposons que \mathbf{w} contienne \mathbf{p} : ceci entraîne que $\mathbf{v} \wedge \mathbf{w} \sim \mathbf{p}$ et donc $\mathbf{p}^{\top}\omega\mathbf{p} = 0$. Au final, puisque ω est le lieu de tous les IPC, \mathbf{p} est un PC de π et \mathbf{q} est son conjugué. ■

Paramétrage minimal. Comme expliqué ci-dessus, les IPC peuvent être décrites à partir de \mathbf{C}^* sous la forme $\mathbf{x}_1 \pm \mathbf{x}_2$, avec $\begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 \end{bmatrix} \equiv \mathbf{U}_{(:,1:2)} \sqrt{\Sigma}_{(1:2,1:2)}$ obtenu à partir de (8).

Dans ce travail, nous avons besoin d'une expression formelle de \mathbf{x}_1 et \mathbf{x}_2 .

Proposition 2 *Les vecteurs $\mathbf{x}_1, \mathbf{x}_2$ satisfaisant (8), et donc (9), sont de la forme suivante : pour $k \in \{1, 2, 3\}$,*

$$\begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 \end{bmatrix} \sim \begin{bmatrix} [\mathbf{v}]_{\wedge} \mathbf{e}_k & \mu [\mathbf{v}]_{\wedge} \omega [\mathbf{v}]_{\wedge} \mathbf{e}_k \end{bmatrix}, \quad (11)$$

où $\mu \equiv \alpha_u \alpha_v / \|\mathbf{K}^{\top} \mathbf{v}\|$ et \mathbf{e}_k est un vecteur canonique, tel que défini dans (1).

Avant de démontrer cette proposition, rappelons que la ligne de fuite \mathbf{a} pour vecteur de paramètres $\mathbf{v} = \mathbf{K}^{-\top} \mathbf{n}$, où \mathbf{n} est la normale unitaire à π dans le repère caméra. De plus, définissons l'ICDPC «calibrée» $\bar{\mathbf{C}}^* \equiv \xi \mathbf{K}^{-1} \mathbf{C}^* \mathbf{K}^{-\top}$, où ξ est un scalaire quelconque tel que $\bar{\mathbf{C}}^* = [\mathbf{n}]_{\wedge}^{\top} [\mathbf{n}]_{\wedge}$.

Preuve. Les valeurs singulières de $\bar{\mathbf{C}}^*$ sont $\{1, 1, 0\}$ donc $\text{Im}(\bar{\mathbf{C}}^*) = \text{Im}([\mathbf{n}]_{\wedge})$ et $\text{Ker}(\bar{\mathbf{C}}^*) = \text{Ker}([\mathbf{n}]_{\wedge})$. Grâce au théorème de la SVD [3], nous savons que la matrice $\mathbf{W} \in \mathbb{R}^{3 \times 3}$, $\mathbf{W}\mathbf{W}^{\top} \sim \mathbf{I}_3$, telle que $\bar{\mathbf{C}}^* \sim \mathbf{W} \text{diag}(1, 1, 0) \mathbf{W}^{\top}$, a les propriétés $\text{Im}(\bar{\mathbf{C}}^*) = \text{Vect}\{\mathbf{w}_1, \mathbf{w}_2\}$ et $\text{Ker}(\bar{\mathbf{C}}^*) = \text{Vect}\{\mathbf{w}_3\}$. Au final, nous pouvons en déduire :

$$\mathbf{w}_1 = [\mathbf{n}]_{\wedge} \mathbf{e}_k, \quad \mathbf{w}_2 = [\mathbf{n}]_{\wedge}^2 \mathbf{e}_k, \quad \mathbf{w}_3 = \mathbf{n}, \quad (12)$$

où $\mathbf{w}_2 = \mathbf{w}_3 \wedge \mathbf{w}_1 = [\mathbf{w}_3]_{\wedge} \mathbf{w}_1$. En substituant $\mathbf{K}^{\top} \mathbf{v}$ à \mathbf{n} dans (12) et après quelques normalisations, nous obtenons (11). ■

L'expression proposée (11) offre l'avantage d'un paramétrage minimal du problème de l'autocalibrage plan. En remplaçant (11) dans (6), il y a maintenant sept degrés de liberté au lieu des neuf dans [17].

Lien avec la contrainte de Malis [10]. Si nous définissons $\bar{\mathbf{H}}_j \equiv \mathbf{K}_j^{-1} \mathbf{H}_j \mathbf{K}_j$, l'ICDPC «calibrée», dans la vue numéro j , a pour matrice

$$\bar{\mathbf{C}}_j^* \equiv [\mathbf{n}_j]_{\wedge}^{\top} [\mathbf{n}_j]_{\wedge} \sim \bar{\mathbf{H}}_j [\mathbf{n}]_{\wedge}^{\top} [\mathbf{n}]_{\wedge} \bar{\mathbf{H}}_j^{\top}, \quad (13)$$

où \mathbf{n}_j est la normale unitaire à π dans le repère caméra numéro j . Ceci est intéressant, car les valeurs singulières de $\bar{\mathbf{C}}_j^*$ sont $\{1, 1, 0\}$ et celles de $\bar{\mathbf{H}}_j [\mathbf{n}]_{\wedge}^{\top}$ sont également

$\{1, 1, 0\}$, à un facteur d'échelle près. Cette dernière propriété est le fondement théorique des contraintes d'autocalibrage de [10].

2.2 Formulation du problème

Supposons que l'ICA est constante dans toutes les vues, c.-à-d. $\omega \sim \omega_j$. Étant données N vues, c.-à-d. $(N - 1)$ homographies inter-vues H_j , $2 \leq j \leq N$, le problème d'autocalibrer une caméra est celui de résoudre le système composé des deux équations (5) et $2(N - 1)$ équations (6) pour les p degrés de liberté dans la matrice de l'ICA plus q dans les vecteurs des IPC. C'est un problème non linéaire, qui peut être contraint et qui a, jusqu'à aujourd'hui, été résolu en utilisant des méthodes itératives. Il nécessite des valeurs initiales qui sont critiques pour la suite, comme cela est mentionné dans [17].

Du fait de l'expression proposée (11) des IPC, comparée à [17], notre modélisation du problème fait intervenir seulement sept inconnues au lieu de neuf. Cependant, il n'y a pas de magie : avec la formulation proposée, l'équation (5), relative à la vue-clef, est implicitement satisfaite, tandis que dans [17], elle est considérée comme une équation à satisfaire. Nous posons la question : devons-nous considérer (5) comme une contrainte ou une équation ? Puisqu'aucune donnée d'entrée, c.-à-d. aucune homographie estimée, n'intervient dans (5), il n'y a pas de raison logique pour ne pas satisfaire exactement cette équation. En fait, les neuf paramètres de [17] ne sont pas indépendants et doivent satisfaire la contrainte additionnelle (5). Plus généralement, si on considère l'estimation d'une homographie, de la vue-clef à une vue numéro j , en utilisant des correspondances de points d'intérêt, l'introduction d'une erreur sur les positions (arbitraires) des points d'intérêt de la vue-clef ne se justifie pas. Comme on pouvait s'y attendre, il n'y a que deux contraintes dans le problème de l'autocalibrage plan, mais plusieurs façons de les exprimer.

Modèle de caméra simplifié. Nous utilisons maintenant le paramétrage minimal avec l'hypothèse d'un modèle de caméra simplifié. Soit la matrice de calibrage $\mathbf{K} = \text{diag}(\alpha, \alpha, 1)$, où α représente la distance focale en pixels. Soit $\mathbf{v} \equiv (\cos \phi, \sin \phi, -\rho)^{\top}$, où ρ est la distance orthogonale du point principal à la ligne de fuite en pixels et ϕ est l'angle entre la ligne de fuite et l'axe u du repère image. Les vecteurs de (11) peuvent alors s'écrire sous la forme :

$$\begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} -\sqrt{\alpha^2 + \rho^2} \sin \phi & \rho \cos \phi \\ \sqrt{\alpha^2 + \rho^2} \cos \phi & \rho \sin \phi \\ 0 & 1 \end{bmatrix}. \quad (14)$$

3 Optimisation globale par calcul d'intervalles

3.1 Calcul par intervalles

Le calcul par intervalles est né il y a environ quarante ans [11]. Plusieurs bonnes introductions au calcul par intervalles sont disponibles dans [4, 12]. Nous résumons ici les règles de base du calcul par intervalles.

Un intervalle est noté $\mathbf{x} = [\underline{x}, \bar{x}]$, où \underline{x} et \bar{x} sont les bornes inférieure et supérieure de \mathbf{x} respectivement. Les vecteurs d'intervalles sont appelés des pavés. Si \mathbf{x} et \mathbf{y} sont deux intervalles, alors les quatre opérations de base sont définies par

$$\mathbf{x} \text{ op } \mathbf{y} = \{x \text{ op } y \mid x \in \mathbf{x} \text{ et } y \in \mathbf{y}\},$$

$$\text{pour op} \in \{+, -, \times, \div\}.$$

Si $\mathbf{x} = [\underline{x}, \bar{x}]$ et $\mathbf{y} = [\underline{y}, \bar{y}]$, alors

$$\mathbf{x} + \mathbf{y} = [\underline{x} + \underline{y}, \bar{x} + \bar{y}],$$

$$\mathbf{x} - \mathbf{y} = [\underline{x} - \bar{y}, \bar{x} - \underline{y}],$$

$$\mathbf{x} \times \mathbf{y} = [\min\{\underline{x}\underline{y}, \underline{x}\bar{y}, \bar{x}\underline{y}, \bar{x}\bar{y}\}, \max\{\underline{x}\underline{y}, \underline{x}\bar{y}, \bar{x}\underline{y}, \bar{x}\bar{y}\}],$$

$$1/\mathbf{x} = [1/\bar{x}, 1/\underline{x}] \text{ si } \underline{x} > 0 \text{ ou } \bar{x} < 0,$$

$$\mathbf{x} \div \mathbf{y} = \mathbf{x} \times 1/\mathbf{y}.$$

En composant ces opérations, nous pouvons calculer une extension de l'image d'une fonction sur un intervalle donné. Par exemple, si $f(x) = x(x - 1)$, alors une extension de f sur $[-1, 1]$ est

$$f([-1, 1]) = [-1, 1]([-1, 1] - 1) = [-1, 1][-2, 0] = [-2, 2],$$

qui contient nécessairement l'image exacte $[-1/4, 2]$ de f . Maintenant, si nous écrivons $f(x) = x(x - 1) = x^2 - x$ et si nous utilisons le fait que $\mathbf{x}^2 = \{x^2 \mid x \in \mathbf{x}\}$, alors nous pouvons calculer une autre extension

$$f([-1, 1]) = [-1, 1]^2 - [-1, 1] = [0, 1] - [-1, 1] = [-1, 2],$$

qui est plus proche de l'image exacte que $[-2, 2]$ ne l'est. Nous soulignons ici le fait que l'expression symbolique de la fonction f a des conséquences critiques sur la qualité de l'extension de f .

3.2 Optimisation globale fondée sur le calcul par intervalles

L'idée d'utiliser le calcul par intervalles pour l'optimisation globale a suscité l'intérêt de nombreux auteurs [4, 8], pour n'en citer que quelques uns. Ces dernières années, l'optimisation globale fondée sur le calcul par intervalles a été envisagée dans des domaines variés. En vision par ordinateur, elle a été appliquée de façon convaincante dans le cadre de l'autocalibrage 3D [2].

Le problème de l'optimisation globale par calcul d'intervalles est le suivant : trouver le minimum global f^* d'une fonction continue f , $f^* = \min\{f(x) \mid x \in \mathbf{D}\}$, ainsi que l'ensemble des points pour lesquels ce minimum global est obtenu, soit $\mathbf{X}^* = \{x \in \mathbf{D} \mid f(x) = f^*\}$, où \mathbf{D} est un pavé. L'optimisation globale par calcul par intervalles utilise généralement le calcul par intervalles avec un algorithme de type *branch and bound*. Soit \mathbf{X} le pavé représentant l'espace de recherche et \mathcal{L} une liste de pavés à traiter. Le processus de base de la méthode peut être décrit comme suit :

1. Initialiser \mathcal{L} en y plaçant l'espace de recherche initial \mathbf{X}_0 .
2. Tant que $\mathcal{L} \neq \emptyset$ faire :

- a Enlever un pavé \mathbf{X} de \mathcal{L} .
- b Traiter \mathbf{X} (rejeter, réduire, existence de points critiques, ...).
- c Subdiviser \mathbf{X} et insérer les pavés issus de \mathbf{X} dans \mathcal{L} .

Plusieurs détails, tels que les critères d'arrêt, ont été omis ici. Nous renvoyons le lecteur à [4] pour une description complète de la méthode. Diverses techniques sont utilisées à l'étape 2.b de l'algorithme, par exemple :

- Test d'inclusion : si $f(\mathbf{X}) > \bar{f}$, avec \bar{f} la meilleure borne supérieure courante, alors éliminer \mathbf{X} ,
- Test de monotonie : si $0 \notin \nabla f(\mathbf{X})$ alors éliminer \mathbf{X} ,
- Test de concavité : si la matrice hessienne $\nabla^2 f$ n'est pas définie positive partout dans \mathbf{X} alors éliminer \mathbf{X} ,
- Méthode de Newton par intervalles : utilisé pour réduire \mathbf{X} , éliminer \mathbf{X} ou vérifier qu'un point critique unique existe dans \mathbf{X} .

3.3 Implémentation

Nous donnons ici quelques détails pratiques concernant notre implémentation. Nous utilisons le modèle de caméra simplifié décrit en §2.2. La fonction de coût que nous minimisons est la somme des carrés des résidus des équations (6) dans lesquelles nous utilisons les expressions simplifiées des IPC données en (14) :

$$f(\alpha, \rho, \phi) = \sum_{j=2}^N (\mathbf{x}_1^T H_j^T \omega H_j \mathbf{x}_1 - \mathbf{x}_2^T H_j^T \omega H_j \mathbf{x}_2)^2 + (\mathbf{x}_1^T H_j^T \omega H_j \mathbf{x}_2)^2.$$

Nous calculons préalablement l'expression symbolique des résidus. À chaque évaluation de la fonction, nous calculons numériquement une extension des résidus développés. Ce choix (la limite entre le calcul formel et le calcul des extensions) semble être un bon compromis entre le temps d'évaluation et la qualité de l'extension de la fonction. Le critère d'arrêt de l'algorithme d'optimisation que nous avons choisi est le suivant : nous stoppons les itérations lorsque la distance focale α et la distance orthogonale ρ sont obtenues à moins d'un pixel près et lorsque l'angle ϕ est obtenu à moins de 0.2° près. Notons que nous devons exclure le cas où le plan de la scène est parallèle au plan image dans la vue-clef (noté le cas fronto-parallèle) car notre paramétrage ne prend pas en compte une ligne de fuite à l'infini (dans ce cas ρ est infini). C'est pourquoi nous imposons à l'angle entre le plan de la scène et le plan image d'être supérieur à 10° . Ce n'est pas une réelle limitation en pratique puisque des solutions directes et linéaires existent dans le cas fronto-parallèle [15] et il est alors mieux d'utiliser ces méthodes dans ce cas précis.

Nous avons écrit un code C++ à partir de la bibliothèque logicielle PROFIL/BIAS³. Afin de simplifier l'implémentation, nous n'avons utilisé que deux tests à l'étape 2.b de l'algorithme d'optimisation : le test d'inclusion et le test de monotonie.

³<http://www.ti3.tu-harburg.de/Software/>

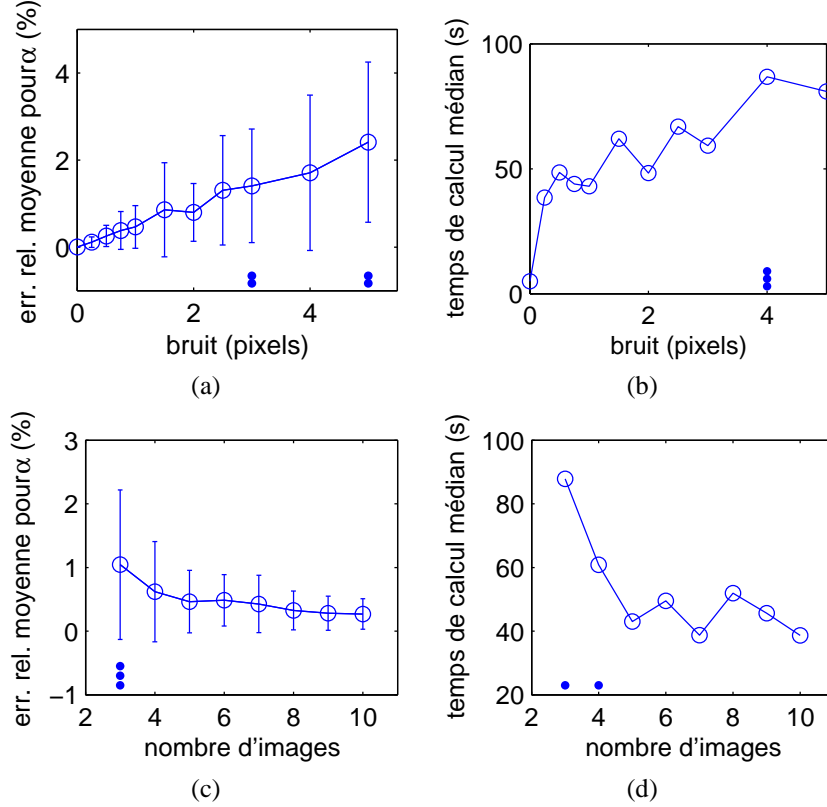


FIG. 1 – Erreur de l’estimation (moyenne et écart type) et temps de calcul pour différents niveaux de bruit et un nombre d’images variable. Chaque point noir compte pour un test exclu (dans (a) et (c)) ou un échec (dans (b) et (d)).

Expérience	(u_0, v_0) fixés, τ fixé	(u_0, v_0) variables, τ fixé	(u_0, v_0) fixés, τ variable
erreur relative moyenne de α (%)	0.5 ± 0.5	2.1 ± 1.8	2.7 ± 2.3
temps de calcul médian (s)	43	92	83
nombre d’échecs	0	2	34

TAB. 2 – Erreur de l’estimation et temps de calcul pour des variations du point principal et du facteur d’échelle. Dans chaque cas, 100 tests ont été réalisés.

4 Résultats expérimentaux

4.1 Simulations

Le protocole expérimental est le suivant : le plan de la scène est une grille composée de 100 points, projetés sur des images 720×576 . Un bruit gaussien avec un écart type égal à σ pixels est ajouté sur les coordonnées de ces points. Dans notre modèle de caméra simplifié, le point principal est fixé au centre de l’image, le facteur d’échelle est égal à 1 et la distorsion des pixels est nulle. La distance focale α est fixée à 1024 pixels. La caméra fixe le centre du plan de la scène. Elle est éloignée de celui-ci d’une distance variable de 1460 ± 570 pixels et son orientation est aléatoirement déterminée : le repère caméra fait un angle compris dans l’intervalle $[10^\circ, 70^\circ]$ autour de l’axe X du plan de la scène, un angle compris dans $[-30^\circ, 30^\circ]$ autour de l’axe Y et un angle compris dans $[-90^\circ, 90^\circ]$ autour de l’axe

Z . Les homographies inter-vues sont estimées en utilisant la méthode NDLT [5, chapitre 4]. Les homographies sont alors transformées de sorte que le point principal coïncide avec l’origine du repère image et telles que $\alpha \rightarrow \alpha/360$ et $\rho \rightarrow \rho/360$.

Dans le but de montrer l’intérêt d’utiliser une méthode d’optimisation globale, nous avons minimisé la fonction de coût en utilisant une méthode itérative : nous avons réalisé des tests avec 5 images et $\sigma = 1$ pixel. Pour chaque test, les inconnues ont été initialisées aléatoirement telles que $\alpha = \alpha^* \pm 30\%$, $\rho = \rho^* \pm 30\%$ et $\phi = \phi^* \pm 30\%$, où $(\alpha^*, \rho^*, \phi^*)$ était le minimum global. La méthode a convergé vers le minimum global (avec une tolérance de 20%) dans 38% des cas. La méthode d’optimisation globale que nous avons utilisée a trouvé la solution dans 100% des cas.

Dans nos expériences, nous avons pris un intervalle ini-

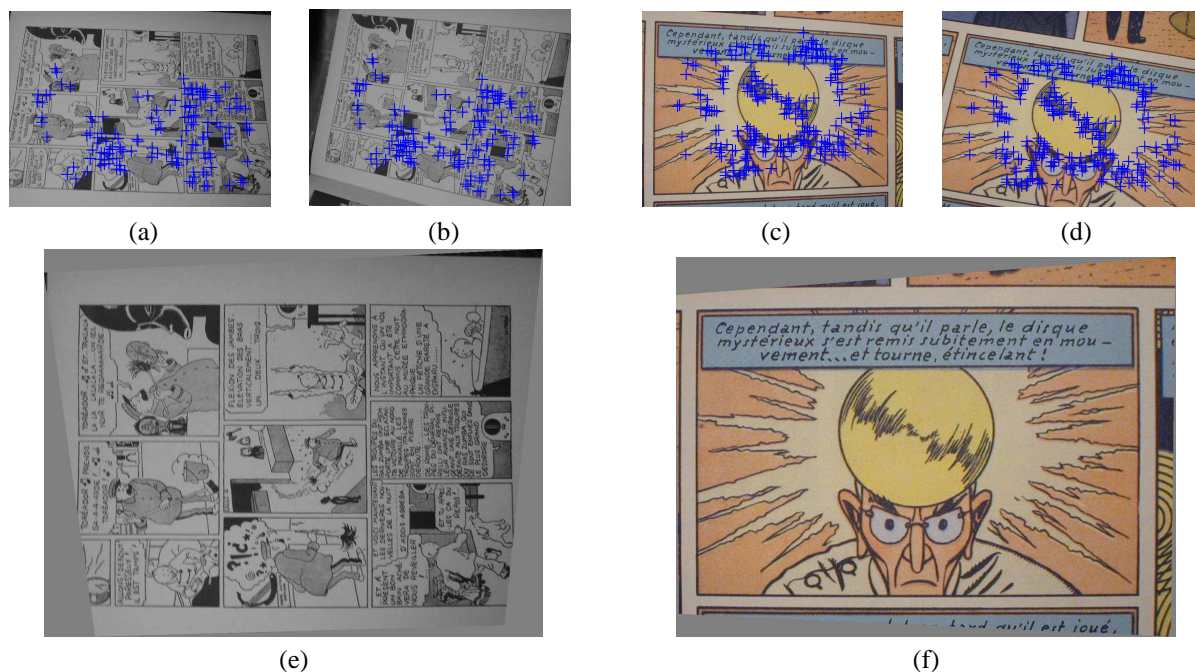


FIG. 2 – Résultats pour des images réelles : (a) and (b) vue-clef et une autre vue de la séquence *tintin*, ainsi que les points appariés utilisés pour l’estimation des homographies inter-vues ; (e) la vue-clef rectifiée ; (c), (d) et (f) mêmes résultats pour la séquence *septimus*.

tial correspondant à $[300, 3000] \times [100, 12000] \times [0, 360]$ pour les inconnues α , ρ et ϕ . Notons que l’intervalle initial n’a pas d’effet sur la précision du minimum calculé ; il influe seulement sur le temps de calcul. Lorsqu’un test ne s’est pas terminé dans un temps raisonnable, nous le déclarons comme un *échec*. Un petit nombre de tests demandant de longs calculs augmenteraient de manière critique la moyenne du temps de calcul donc nous donnerons la médiane du temps de calcul à la place. Quelques fois, le mouvement induit par la séquence d’images est proche d’une configuration dégénérée. Dans ces cas, le minimum global est un peu éloigné du minimum global théorique. C’est pourquoi nous excluons les tests tels que l’erreur relative commise est supérieure à 10% dans le calcul des moyennes et des écarts types. Nous donnerons le nombre de ces tests exclus. Les temps de calcul sont donnés pour un Pentium M 2GHz. La figure 1.a donne quelques résultats concernant l’estimation de α en fonction du niveau de bruit. Les résultats pour ρ et ϕ sont très similaires et ne sont pas présentés ici. La précision reste très bonne lorsque le niveau de bruit augmente, même pour $\sigma = 5$ pixels. Il n’y a que quelques tests exclus (seulement 4 tests ont été exclus sur un total de 11000 tests). L’estimation de α en fonction du nombre d’images est montrée sur la figure 1.c. L’erreur de l’estimation diminue lorsqu’on ajoute plus d’images mais le bénéfice s’avère moins important après 5-6 images. Les figures 1.b et 1.d montrent le temps de calcul en fonction du niveau de bruit et du nombre d’images. Dans le premier cas, la présence de bruit a un impact critique sur le temps de calcul tandis que le niveau de bruit est lui moins impor-

tant. En effet, le temps de calcul n’augmente pas de manière critique avec le niveau de bruit (la croissance semble linéaire). Dans le second cas, la figure montre qu’avec plus de 5 images le temps de calcul reste approximativement le même : il y a plus de termes à évaluer dans la fonction de coût tandis que le fait d’ajouter plus d’images contraint mieux les inconnues et le minimum global est plus facile à encadrer.

Puisque nous avons fait des hypothèses sur τ , u_0 et v_0 , nous avons testé la tolérance de notre méthode à une variation de ces paramètres internes. Nous avons d’abord fait varier la position du point principal dans un carré de 50×50 pixels. Ensuite, nous avons fait varier τ dans l’intervalle $[0.95, 1.05]$ (c.-à-d. une variation de 5%). Nous avons utilisé 5 images et le niveau de bruit a été fixé à $\sigma = 1$ pixel. Les résultats sont montrés dans le tableau 2. Les variations de (u_0, v_0) et de τ n’ont pas de conséquences critiques sur l’estimation des paramètres. Cependant, nous pouvons voir qu’une variation de τ influe sur le nombre d’échecs. Ce n’est pas une limitation réelle de la méthode car en pratique τ est généralement très proche de l’unité.

4.2 Images réelles

Nous avons testé la méthode sur deux séquences, *tintin* et *septimus* (cette dernière a été précédemment utilisée dans [15]), composées de 7 et 4 images 640×480 pixels respectivement. Les homographies ont été estimées en utilisant la bibliothèque⁴ implémentant la méthode optimale de Kanatani [7], fournie par l’auteur. Les points d’intérêt ont été ap-

⁴<http://www.ail.cs.gunma-u.ac.jp/Labo/programs-e.html>

pariés automatiquement. Ensuite, nous avons appliqué une rectification euclidienne de la vue-clef. Dans l'image rectifiée, le plan de la scène doit être parallèle au plan image et le parallélisme ainsi que les angles doivent être retrouvés. La figure 2 montre deux images de chaque séquence, ainsi que les points appariés et la vue-clef rectifiée. Nous pouvons voir que les rectifications sont plutôt bonnes. Le calcul a demandé 4 minutes 34 secondes dans le cas de *tintin* et 8 minutes 14 secondes pour *septimus*.

5 Conclusion

Nous avons proposé un paramétrage minimal du problème de l'autocalibrage 2D. En supposant que la distance focale est la seule inconnue, il y a trois paramètres qui peuvent être estimés avec une méthode d'optimisation globale, nous garantissant ainsi l'obtention de la solution. Une solution garantie a récemment été proposée pour l'autocalibrage 3D [2], problème pour lequel il existe des solutions directes [1], contrairement à l'autocalibrage 2D. Notre modèle de caméra simplifié ne montre pas de réelle limitation car nous avons vu que la méthode est tolérante à des variations conséquentes de la position du point principal et puisque l'hypothèse d'un facteur d'échelle égal à l'unité est très réaliste vis-à-vis des caractéristiques techniques des appareils photo numériques récents. La méthode proposée semble bien fonctionner tant que les cas dégénérés sont évités. L'étude de ces cas dégénérés sont en dehors du cadre de ce travail, mais elle se révèle être essentielle si nous voulons utiliser un nombre minimal d'images [16]. Nous avons aussi pour but d'améliorer les performances de l'algorithme que nous avons utilisé.

Remerciements

Les auteurs souhaitent remercier Frédéric Messine pour des discussions enrichissantes.

Références

- [1] S. BOUGNOUX. « From Projective to Euclidean Space under any Practical Situation, a Criticism of Self-Calibration ». Dans *International Conference on Computer Vision*, volume 1, pages 790–796, Bombay, Inde, janvier 1998.
- [2] A. FUSIELLO, A. BENEDETTI, M. FARENZENA et A. BUSTI. « Globally Convergent Autocalibration Using Interval Analysis ». *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(12) :1633–1638, décembre 2004.
- [3] G. GOLUB et C. Van LOAN. « *Matrix computations*. ». Troisième édition. John Hopkins University Press, 1996.
- [4] E. R. HANSEN et G. W. WALSTER. « *Global Optimization Using Interval Analysis*. ». Seconde édition. Marcel Dekker, 2003.
- [5] R. HARTLEY et A. ZISSERMAN. « *Multiple View Geometry*. ». Seconde édition. Cambridge University Press, 2003.
- [6] G. JIANG, H. T. TSUI et L. QUAN. « Circular Motion Geometry Using Minimal Data ». *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6) :721–731, juin 2004.
- [7] K. KANATANI et N. OHTA. « Accuracy Bounds and Optimal Computation of Homography for Image Mosaicing Applications ». Dans *International Conference on Computer Vision*, volume 1, pages 73–78, Corfu, Grèce, septembre 1999.
- [8] R. B. KEARFOTT. « *Rigorous Global Search : Continuous Problems*. ». Kluwer Academic Publishers, 1996.
- [9] J. KNIGHT, A. ZISSERMAN et I. REID. « Linear Auto-Calibration for Ground Plane Motion ». Dans *International Conference on Computer Vision and Pattern Recognition*, pages 503–510, Madison, Wisconsin, États-Unis, juin 2003.
- [10] E. MALIS et R. CIPOLLA. « Multi-view Constraints between Collineations : Application to Self-Calibration from Unknown Planar Structures ». Dans *European Conference on Computer Vision*, volume 2, pages 610–624, Dublin, Irlande, juin 2000.
- [11] R. E. MOORE. « *Interval Analysis*. ». Prentice-Hall, 1966.
- [12] A. NEUMAIER. « *Interval Methods for Systems of Equations*. ». Cambridge University Press, 1990.
- [13] A. NEUMAIER. « *Introduction to Numerical Analysis*. ». Cambridge University Press, 2001.
- [14] J. SEMPLE et G. KNEEBONE. « *Algebraic Projective Geometry*. ». Oxford Classic Series, Clarendon Press, 1952, réédité, 1998.
- [15] P. STURM et P. GURDJOS. « Méthodes et fondements géométriques pour l'autocalibrage plan ». Dans *Reconnaissance des Formes et Intelligence Artificielle*, volume 1, pages 351–360, Toulouse, France, janvier 2004.
- [16] P. STURM et S. J. MAYBANK. « On plane-based camera calibration : A general algorithm, singularities, applications ». Dans *International Conference on Computer Vision and Pattern Recognition*, volume 1, pages 432–437, Fort Collins, Colorado, États-Unis, juin 1999.
- [17] B. TRIGGS. « Autocalibration from Planar Scenes ». Dans *European Conference on Computer Vision*, volume 2, pages 89–105, Friburg, Allemagne, juin 1998.