

Reconstruction métrique minimale à partir de trois caméras affines

Minimal Metric Structure and Motion from Three Affine Images

Marc-André Ameller

Adrien Bartoli

Long Quan

INRIA Rhône-Alpes, 655, av. de l'Europe
38334 St. Ismier cedex, France.
prénom.nom@inria.fr

Résumé

Le calcul du mouvement des caméras et de la structure 3D à partir de données minimales est essentiel pour la mise en œuvre de méthodes robustes telles RANSAC (RANDOM SAmple Consensus) ou LMS (moindre carrés médians). Pour les images issues de caméras affines, 4 points et 3 caméras sont nécessaires lorsque le rapport d'aspect et la distortion entre les axes sont supposées connues. Nous proposons ici une nouvelle paramétrisation basée sur la structure métrique du problème par opposition à l'approche classique impliquant les paramètres du mouvement des caméras. Les coordonnées des 4 points sont obtenues par la résolution d'une équation quadratique. Cette modélisation permet aussi l'étude des configurations instables. Les résultats expérimentaux, sur des données simulées ainsi que sur des images réelles ont montré la stabilité du nouvel algorithme. Les solutions calculées à partir de données minimales sont aussi comparées aux solutions optimales obtenues à partir de données redondantes.

Mots Clef

Reconstruction métrique, caméra affine.

Abstract

Structure and motion from minimal data is essential to bootstrap robust methods based on random sampling such as RANSAC (RANDOM SAmple Consensus) or LMS (Least Median of Squares). Let us consider the affine camera model and make the hypotheses of zero skew and unity aspect ratio. In this case, at least 4 points in 3 images are necessary to recover structure and motion. We propose a parametrization based on metric structure rather than camera motion parameters which have been previously used. The structure of 4 points is computed in closed-form by solving a quadratic equation. Unstable configurations are also investigated. Experimental results on simulated data and real images demonstrate the stability of our algorithm. Computed solutions with minimal data are compared with those obtained after optimization on redundant data.

Keywords

Metric Structure and Motion, Affine Camera.

1 Introduction

Le calcul d'un modèle 3D à partir de correspondances de primitives géométriques entre des images est un sujet fondamental en vision par ordinateur. En particulier, reconstruire des points, et estimer le mouvement relatif des caméras sur des données minimales est d'une importance cruciale pour de nombreuses procédures d'estimation. Ceci est valable pour les algorithmes robustes, dont le but est l'élimination des données aberrantes, procédant par tirages aléatoires, tels RANSAC ou les moindres carrés médians. Dans cet article, nous proposons un algorithme pour obtenir la reconstruction métrique à partir de 3 images affines avec un minimum de 4 correspondances de points. La reconstruction à partir de caméras affines a été étudiée pour les points [8, 7] et les droites [1]. Elle est définie seulement à un paramètre près, pour la reconstruction à partir de deux caméras, et ceci indépendamment du nombre de correspondances données, mais devient possible, sans ambiguïté lorsqu'au moins 3 caméras sont considérées. Les caméras affines peuvent être aussi bien des projections orthogonales, perspectives faible, que paraperspectives [4]. Une reconstruction métrique signifie que les points sont définis à une similitude directe près de l'espace.

Nous proposons une nouvelle formulation du problème de reconstruction. Nous montrons comment la structure métrique des 4 points dans l'espace peut être contrainte par un système de polynômes. À l'aide d'outils issus de la géométrie algébrique, une solution exacte est proposée, ainsi qu'une étude des cas instables. Le principal avantage de la méthode est qu'elle donne directement la structure 3D, sans passer par les paramètres du mouvement, en ne nécessitant que la résolution d'une équation quadratique.

Cet article est organisé de la manière suivante. La section 2 fournit un bref rappel du modèle de caméra affine. La section 3 donne la formulation du problème en terme d'un système de polynômes. La section 4 donne les résultats des expériences sur des données synthétiques, et la section 5

présente les résultats comparatifs entre la méthode présentée dans cet article et la solution obtenue après ajustement de faisceaux, ainsi que des expériences sur des données réelles.

2 Le modèle de caméra affine

Le modèle de caméra perspective est donné par une matrice à 3 lignes et 4 colonnes. Ainsi, pour un point $M = (X, Y, Z, T)^T$ de l'espace projectif, exprimé en coordonnées homogènes, sa projection $m = (x, y, z)^T$ sur l'image (considérée comme un plan projectif) est donnée par la formule :

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} \sim \begin{pmatrix} p_{1,1} & p_{1,2} & p_{1,3} & p_{1,4} \\ p_{2,1} & p_{2,2} & p_{2,3} & p_{2,4} \\ p_{3,1} & p_{3,2} & p_{3,3} & p_{3,4} \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ T \end{pmatrix},$$

où \sim signifie "égal à un facteur non-nul près". Dans cet article, on utilise un modèle de caméra affine, ce qui signifie que les points à l'infini de l'espace 3D sont projetés à l'infini dans l'image. Ceci implique que :

$$p_{3,1} = p_{3,2} = p_{3,3} = 0.$$

L'équation de projection peut ainsi être réduite à :

$$\frac{1}{z} \begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{T} \begin{pmatrix} p'_{1,1} & p'_{1,2} & p'_{1,3} \\ p'_{2,1} & p'_{2,2} & p'_{2,3} \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + \begin{pmatrix} t_1 \\ t_2 \end{pmatrix}.$$

Notez qu'il s'agit là d'une égalité stricte. En utilisant une décomposition QR [5], on obtient :

$$\begin{pmatrix} u \\ v \end{pmatrix} = \lambda \begin{pmatrix} 1 & \epsilon \\ 0 & \xi \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \mathbf{R} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + \begin{pmatrix} t_1 \\ t_2 \end{pmatrix}, \quad (1)$$

où \mathbf{R} est une matrice de rotation 3×3 , λ est le facteur d'échelle, ξ est le rapport d'aspect et ϵ est la distortion entre les axes. La calibration de la caméra est équivalente à la connaissance des paramètres λ , ϵ et ξ . Si ϵ et ξ sont connus, on peut effectuer un changement de repère [6] de telle sorte que la projection s'écrive :

$$\begin{pmatrix} u \\ v \end{pmatrix} = \lambda \begin{pmatrix} \mathbf{R}_1^T \\ \mathbf{R}_2^T \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}, \quad (2)$$

où \mathbf{R}_1^T et \mathbf{R}_2^T sont les deux premières lignes d'une matrice de rotation 3×3 . En pratique, on peut supposer $\epsilon = 0$ et $\xi = 1$, ce qui est valable pour la plupart des caméras actuelles.

3 La reconstruction métrique

3.1 Formulation du problème

Considérons 3 caméras affines, chacune d'elles étant représentée par sa matrice de projection :

$$\mathbf{P}_i = \lambda_i \begin{pmatrix} \mathbf{R}_1^i \\ \mathbf{R}_2^i \end{pmatrix},$$

où $i \in \{1, 2, 3\}$. Pour chaque caméra, on peut définir un plan \mathcal{P}_i arbitraire: le plan focal de cette caméra (c'est-à-dire le plan où les points 3D se projettent orthogonalement). Ces plans sont définis à une translation près. De plus, en exprimant tous les repères relativement à celui de la première caméra, on obtient $\mathbf{P}_1 = (\mathbf{I}|0)$.

Minimalité de la configuration. Commençons par compter le nombre de degrés de liberté du problème. La première caméra peut être fixée arbitrairement, et pour chaque caméra supplémentaire, on ajoute 6 degrés de liberté au problème. Ceci correspond aux 3 degrés de liberté de la rotation, aux 2 degrés de liberté de la translation (la translation n'a de signification que lorsque sa direction est dans le plan de projection lui-même), ainsi qu'à un degré de liberté correspondant au facteur d'échelle λ , voir équation (1), les paramètres internes ϵ et ξ étant supposés connus. Ensuite, nous allons calculer le nombre de contraintes imposés par les points de l'espace 3D: chaque point donne 6 contraintes et 3 inconnues. Le problème avec 3 caméras et n points possède donc une solution si et seulement si

$$3n + 6 \times (3 - 1) \leq 6 \times n.$$

Ceci implique que l'on a besoin d'au moins 4 points pour résoudre le problème avec 3 caméras. Cette configuration est minimale.

3.2 Résolution du problème

Pour chaque point M de l'espace, on note $\delta_i(M)$, la distance entre M et \mathcal{P}_i (Plan défini au paragraphe 3.1). Ainsi, pour chaque couple (M_p, M_q) dans l'espace, et pour la $i^{\text{ème}}$ caméra, on peut utiliser le théorème de Pythagore. Ceci donne :

$$|\delta_i(M_p) - \delta_i(M_q)|^2 + \lambda_i^2 \|m_p^i - m_q^i\|^2 = \|M_p M_q\|^2, \quad (3)$$

où $m_p^i = \mathbf{P}_i M_p$, et $m_q^i = \mathbf{P}_i M_q$ (voir figure 1). Cette équation possède des propriétés intéressantes.

Pour notre problème de reconstruction, les paramètres ϵ et ξ de la caméra sont connue de telle sorte que \mathbf{P}_i peut être mis sous la forme donnée dans l'équation 2. Les coordonnées des correspondances dans les images sont connues. Ceci signifie que seules les distances $\|m_p^i - m_q^i\|$ sont connues.

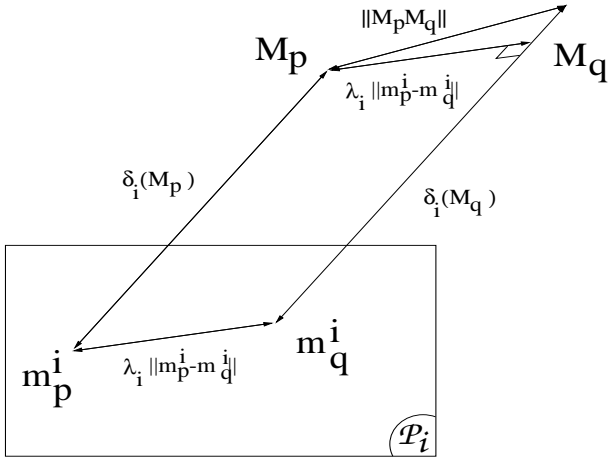


FIG. 1 – Théorème de Pythagore appliqué à la projection affine de deux points.

Elimination des facteurs d'échelle λ_i . Il est clair que le problème est défini à un facteur d'échelle global près. C'est-à-dire que si l'on multiplie toutes les distances par un scalaire non nul, cela ne change pas la structure 3D. Ainsi, on peut poser $\lambda_1 = 1$. Il est bien connu qu'on peut calculer la matrice fondamentale affine à partir de 4 correspondances de points [6]. On la calcule entre la caméra 1 et la caméra i , pour $i \neq 1$. Le facteur d'échelle peut être calculé à partir de la matrice fondamentale selon la formule donnée dans [6]:

$$\lambda_i = \sqrt{\frac{c^2 + d^2}{a^2 + b^2}},$$

si la matrice fondamentale affine entre la première image et la seconde image est donnée par:

$$\begin{pmatrix} 0 & 0 & a \\ 0 & 0 & b \\ c & d & e \end{pmatrix}.$$

Réduction de l'équation (3). Pour simplifier, on introduit les notations suivantes:

$$\begin{aligned} \alpha_{i,p,q} &= \lambda_i^2 \|m_p^i - m_q^i\|^2 \\ X_p &= \delta_1(M_p) \\ Y_p &= \delta_2(M_p) \\ Z_p &= \delta_3(M_p). \end{aligned}$$

L'équation 3 devient ainsi, pour chacune des 3 paires d'images possibles:

$$\begin{aligned} (X_p - X_q)^2 + \alpha_{1,p,q} &= \|M_p M_q\|^2 \\ (Y_p - Y_q)^2 + \alpha_{2,p,q} &= \|M_p M_q\|^2 \\ (Z_p - Z_q)^2 + \alpha_{3,p,q} &= \|M_p M_q\|^2. \end{aligned}$$

En soustrayant deux à deux les équations ci-dessus, on obtient:

$$\begin{aligned} (X_p - X_q)^2 + \alpha_{1,p,q} &= (Y_p - Y_q)^2 + \alpha_{2,p,q} \\ (X_p - X_q)^2 + \alpha_{1,p,q} &= (Z_p - Z_q)^2 + \alpha_{3,p,q}. \end{aligned}$$

On rappelle que X_p, Y_p, Z_p sont définis à la définition près des plans P_i , eux-mêmes définis à une translation près. On peut ainsi poser $X_0 = Y_0 = Z_0 = 0$. Le système ainsi obtenu pour 4 correspondances de points dans 3 images est un système polynomial $\{P_{i,j}, Q_{i,j} \mid 1 < i < j < 4\}$ avec:

$$\begin{aligned} P_{1,2} &= X_1^2 - Y_1^2 + k_{1,2} \\ P_{1,3} &= X_2^2 - Y_2^2 + k_{1,3} \\ P_{1,4} &= X_3^2 - Y_3^2 + k_{1,4} \\ P_{2,3} &= (X_1 - X_2)^2 - (Y_1 - Y_2)^2 + k_{2,3} \\ P_{2,4} &= (X_1 - X_3)^2 - (Y_1 - Y_3)^2 + k_{2,4} \\ P_{3,4} &= (X_2 - X_3)^2 - (Y_2 - Y_3)^2 + k_{3,4} \\ Q_{1,2} &= X_1^2 - Z_1^2 + l_{1,2} \\ Q_{1,3} &= X_2^2 - Z_2^2 + l_{1,3} \\ Q_{1,4} &= X_3^2 - Z_3^2 + l_{1,4} \\ Q_{2,3} &= (X_1 - X_2)^2 - (Z_1 - Z_2)^2 + l_{2,3} \\ Q_{2,4} &= (X_1 - X_3)^2 - (Z_1 - Z_3)^2 + l_{2,4} \\ Q_{3,4} &= (X_2 - X_3)^2 - (Z_2 - Z_3)^2 + l_{3,4}, \end{aligned} \quad (4)$$

où

$$\begin{aligned} k_{i,j} &= \alpha_{1,i-1,j-1} - \alpha_{2,i-1,j-1}, \\ l_{i,j} &= \alpha_{1,i-1,j-1} - \alpha_{3,i-1,j-1}. \end{aligned} \quad (5)$$

Le but est de calculer les X_i, Y_j et Z_k à partir des $k_{i,j}$ et $l_{i,j}$. On va d'abord s'intéresser aux cas instables pour lesquels une petite erreur sur les données donne lieu à des erreurs importantes sur le résultat. Enfin, on va donner un algorithme complet pour le calcul de la structure et du mouvement.

3.3 Les conditions d'instabilité

Pour étudier les cas instables, remarquons d'abord que le calcul d'une seule des inconnues ne nécessite la considération que de seulement 6 équations. On restreint donc le système aux équations:

$$\begin{aligned} P_{1,2} &= X_1^2 - Y_1^2 + k_{1,2} \\ P_{1,3} &= X_2^2 - Y_2^2 + k_{1,3} \\ P_{2,3} &= (X_1 - X_2)^2 - (Y_1 - Y_2)^2 + k_{2,3} \\ Q_{1,2} &= X_1^2 - Z_1^2 + l_{1,2} \\ Q_{1,3} &= X_2^2 - Z_2^2 + l_{1,3} \\ Q_{2,3} &= (X_1 - X_2)^2 - (Z_1 - Z_2)^2 + l_{2,3}. \end{aligned}$$

On introduit un nouveau système de polynômes obtenu avec les coefficients mesurés:

$$\begin{aligned} k_{i,j}^{mes} &= k_{i,j} + \delta k_{i,j} \\ l_{i,j}^{mes} &= l_{i,j} + \delta l_{i,j}. \end{aligned}$$

On note les nouvelles solutions issues du système "mesuré":

$$\begin{aligned} X_i^{mes} &= X_i + \delta X_i \\ Y_i^{mes} &= Y_i + \delta Y_i \\ Z_i^{mes} &= Z_i + \delta Z_i. \end{aligned}$$

Supposons que X_i, Y_i, Z_i soient les solutions du système original (exact). Si l'on soustrait les équations du système "mesuré" à celle du système exact, après quelques manipulations algébriques, le système s'écrit sous la forme:

$$\delta \mathbf{K} = \mathbf{M} \delta \mathbf{X} + \varepsilon$$

où

$$\delta \mathbf{K} = \begin{pmatrix} \delta k_{1,2} \\ \delta k_{1,3} \\ \delta k_{2,3} \\ \delta l_{1,2} \\ \delta l_{1,3} \\ \delta l_{2,3} \end{pmatrix} \quad \text{et} \quad \delta \mathbf{X} = \begin{pmatrix} \delta X_1 \\ \delta X_2 \\ \delta Y_1 \\ \delta Y_2 \\ \delta Z_1 \\ \delta Z_2 \end{pmatrix}.$$

La matrice M a des coefficients linéaires en les X_i, Y_j, Z_k , et ε est un vecteur dont les composantes s'expriment en fonction des $\delta X_i^2, \delta Y_i^2, \delta Z_i^2$. Les cas instables apparaissent quand la matrice M est singulière, c'est-à-dire quand son déterminant s'annule. Dans ce cas, de petites variations sur $\delta \mathbf{K}$ induisent de grosses erreurs sur $\delta \mathbf{X}$. Après développement et factorisation, on obtient :

$$\det M = 64(Z_2 Y_1 - Y_2 Z_1)(Z_2 X_1 - X_2 Z_1)(Y_2 X_1 - Y_1 X_2).$$

On doit maintenant interpréter les configurations instables correspondant à l'annulation de ce déterminant de manière géométrique. Il existe deux cas:

- les trois plans des caméras sont linéairement dépendants (i.e. les trois plans de projections ne se coupent pas en un unique point);
- les images des points sont alignées dans une des images.

Ceci peut être démontré de la manière suivante. Posons \mathbf{i}_k un vecteur unité orthogonal au plan \mathcal{P}_k .

Regardons maintenant le cas où les trois vecteurs \mathbf{i}_k sont linéairement dépendants. Ce cas est équivalent au cas de deux images (aucune reconstruction ne peut en être déduite). En effet, ce cas signifie que l'on peut calculer la troisième image par combinaison linéaire des deux précédentes.

Le second cas est celui où les trois vecteurs \mathbf{i}_k sont linéairement indépendants. Ainsi, les vecteurs $\{\mathbf{i}_1, \mathbf{i}_2, \mathbf{i}_3\}$ définissent une base. Il est clair que les points M_0, M_1, M_2 ont pour coordonnées dans cette base respectivement $(0,0,0), (X_1, Y_1, Z_1), (X_2, Y_2, Z_2)$. La condition d'instabilité $\det M = 0$ est équivalente au fait qu'il existe $k \in \{1, \dots, 3\}$ tel que les projections orthogonales de $M_0 M_1$ et de $M_0 M_2$ sur \mathcal{P}_k soient collinéaires, autrement dit, les images des points M_0, M_1 , et M_2 sont alignés dans une des images.

3.4 Description de l'algorithme

Nous sommes maintenant en mesure de donner l'algorithme complet de reconstruction. L'idée principale est d'utiliser les résultants [2] pour éliminer successivement les variables, de manière analogue au pivot de Gauss pour les systèmes linéaires. Le calcul est effectué de manière à obtenir un

polynôme en une seule variable. Voici le détail de l'algorithme (les expressions manipulées ici ne sont pas explicitées, car trop complexes, mais sont facile à obtenir en utilisant un logiciel de calcul formel) :

1. calculer les facteurs d'échelle λ_i à l'aide des matrices fondamentales affines [6];
2. en déduire les coefficients $k_{i,j}$ et $l_{i,j}$, en utilisant l'équation (5);
3. calculer symboliquement le résultant R_1 de $P_{1,2}$ et $P_{2,3}$ en la variable Y_1 ;
4. calculer symboliquement le résultant R_2 de R_1 et $P_{1,3}$ en la variable Y_2 ;
5. en factorisant R_2 , on constate que c'est un carré. On pose donc $R_{1,2} = \sqrt{R_2}$;
6. calculer formellement le résultant S_1 de $Q_{1,2}$ et $Q_{2,3}$ en la variable Z_1 ;
7. calculer formellement le résultant S_2 de S_1 et $Q_{1,3}$ en la variable Z_2 ;
8. poser $S_{1,2} = \sqrt{S_2}$ (de même que R_2, S_2 est un carré);
9. calculer formellement le résultant T de $R_{1,2}$ et $S_{1,2}$ en la variable X_2 .

On obtient un polynôme T qui est un polynôme de degré 2 en X_2^2 . L'équation $T = 0$ peut être résolue de manière exacte. On résoud ainsi l'équation pour chacune des variables. En remplaçant les solutions obtenues dans le système de polynôme initial, on peut sélectionner la bonne solution. La sélection des solutions peut aussi être faite à l'aide du théorème d'extension (cf [2], p162). Il s'ensuit que l'algorithme donne les coordonnées 3D des points dans l'espace de manière exacte. De ceci, on peut évidemment déduire les matrices de projection.

4 Les Résultats expérimentaux

4.1 Estimation au maximum de vraisemblance

La méthode décrite dans ce papier est minimale, en d'autres termes, il n'existe aucune fonction de coût et les résultats obtenus sont exacts, ne dépendant que des données entrées. Le but de l'utilisation du maximum de vraisemblance est de prouver la stabilité du résultat obtenu. Cela consiste en une minimization aux moindres carrés de l'erreur de reprojection en utilisant une paramétrisation adéquate des matrices de caméras et des points de la scène. Notons Θ l'ensemble des paramètres. Cet ensemble contient $12 + 3n$ paramètres de mouvement qui correspondent à deux rotations, deux translations, deux facteurs d'échelle et $3n$ paramètres de structure, où n est le nombre de points considérés. Les matrices des caméras vérifient l'équation (1). L'estimateur au maximum de vraisemblance est donné par $\operatorname{argmin}_{\Theta} \mathcal{C}(\Theta)$ où la fonction de coût est donnée par :

$$\mathcal{C}(\Theta) = \sum_{i=1}^{i=3} \sum_{j=1}^{j=n} d^2(m_j^i, \hat{m}_j^i),$$

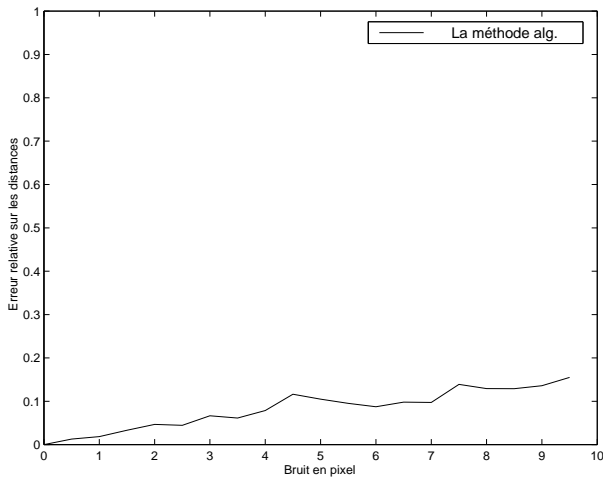


FIG. 2 – L’erreur médiane avec les différents niveaux de bruit sur les données simulées.

où $d^2(.,.)$ désigne la distance Euclidienne dans l’image, m_j^i désigne les points mesurés dans les images et \hat{m}_j^i désigne les points reprojétés. Pour l’optimisation, on utilise l’algorithme de Levenberg-Marquardt [3].

Remarquons que théoriquement, les deux résultats devraient être les mêmes, puisqu’on compare un algorithme minimal et une solution optimisée à partir de son résultat. La comparaison des deux solutions a pour but d’évaluer la stabilité effective de l’algorithme. Le maximum de vraisemblance est aussi appelé dans ce cas l’ajustement de faisceaux.

4.2 Expériences sur des données simulées

Expériences sur des données minimales. Le protocole est fixé de la manière suivante. On choisit au hasard 3 caméras affines, et 4 points 3D. On calcule les projections des points sur chacune des caméras, et on ajoute un bruit uniforme à ces projections.

Avec ces reprojctions, on calcule la structure 3D à l’aide de l’algorithme qui est décrit dans cet article. Ensuite, on compare la structure 3D obtenue par le calcul et celle définie par les points 3D que l’on a tirés au hasard. Ceci permet une évaluation de l’erreur de reconstruction. On répète cette expérience 100 fois pour chaque niveau de bruit, en faisant varier le bruit de 0 à 10 pixels. Le résultat est représenté sur la figure 2. Ceci montre que l’algorithme est stable.

Expériences avec l’utilisation de RANSAC. On prend maintenant 157 points 3D issus d’une mire de calibration (Figure 4). On prend 3 images de cette mire. On effectue une reconstruction à l’aide de l’algorithme décrit dans l’article et de ces 3 images. On projette maintenant les points 3D (dont on connaît les coordonnées dans l’espace) avec les projections calculées par l’algorithme, et on ajoute un bruit uniforme sur les points images obtenus. On calcule ensuite la reconstruction à l’aide de RANSAC. Dans un premier temps, on compare la reconstruction obtenue avec celle issue de la solution optimisée par ajustement de faisceaux sur

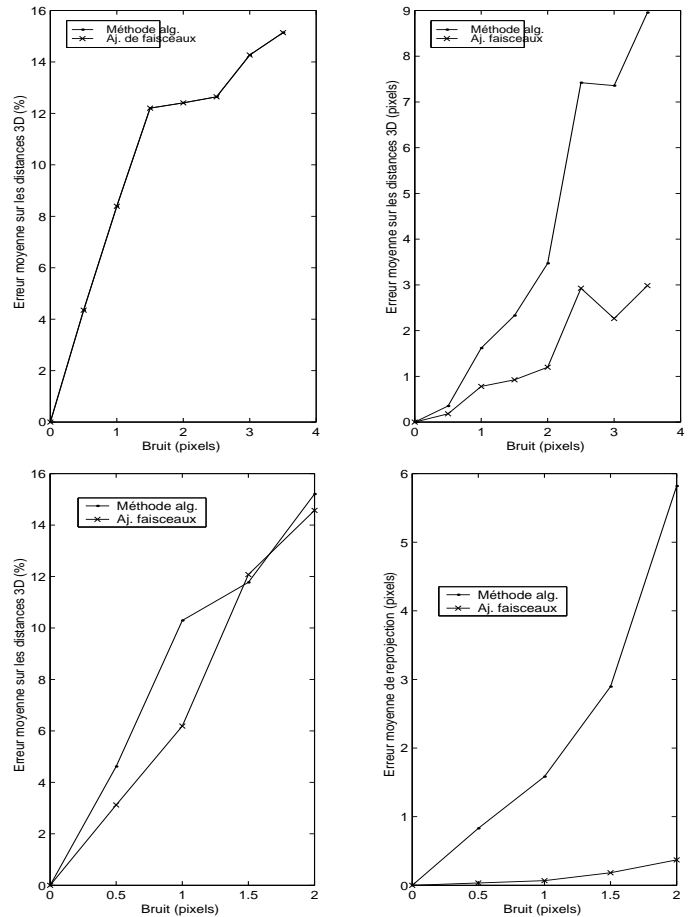


FIG. 3 – Les deux premiers graphes présentent une comparaison entre la méthode présentée et la méthode présentée suivie d’un ajustement de faisceaux sur les mouvements des caméras. Les deux graphes suivants présentent une comparaison entre la méthode présentée et la méthode présentée suivie d’un ajustement de faisceaux sur les mouvements des caméras et la structure. Note: dans le premier graphique, les deux courbes sont superposées.

le mouvement des caméras. L’ajustement de faisceaux est fait sur les données issues de l’algorithme présenté dans l’article. Ensuite, on fait la même chose sauf qu’au lieu d’effectuer l’ajustement de faisceaux uniquement sur le mouvement des caméras, on le fait en plus sur la structure. Les résultats sont présentés dans la figure 3. On voit que les solutions données par l’algorithme sont presque aussi bonnes que celles optimisées, produites à la suite d’un ajustement de faisceaux.

4.3 Expériences sur des images réelles

Expériences sur la mire. Nous avons expérimenté la technique de reconstruction sur des images d’une mire. Avec un logiciel approprié on a mis les points en correspondances (Il n’y a pas d’outliers dans ces images). La structure de la mire nous permet d’en déduire les paramètres ϵ et ξ de la caméra affine (On a remarqué qu’entre les trois

images, ces paramètres ne fluctuait pas de plus de 0.5%, et étaient proches de 0 et 1 respectivement). La mire est un objet composé de trois plans, formant entre eux deux angles droits (figure 4). Dans un premier temps, on a remarqué qu'après reconstruction et reprojection des points, on a une erreur de reprojection de moins de 0.5 pixel.

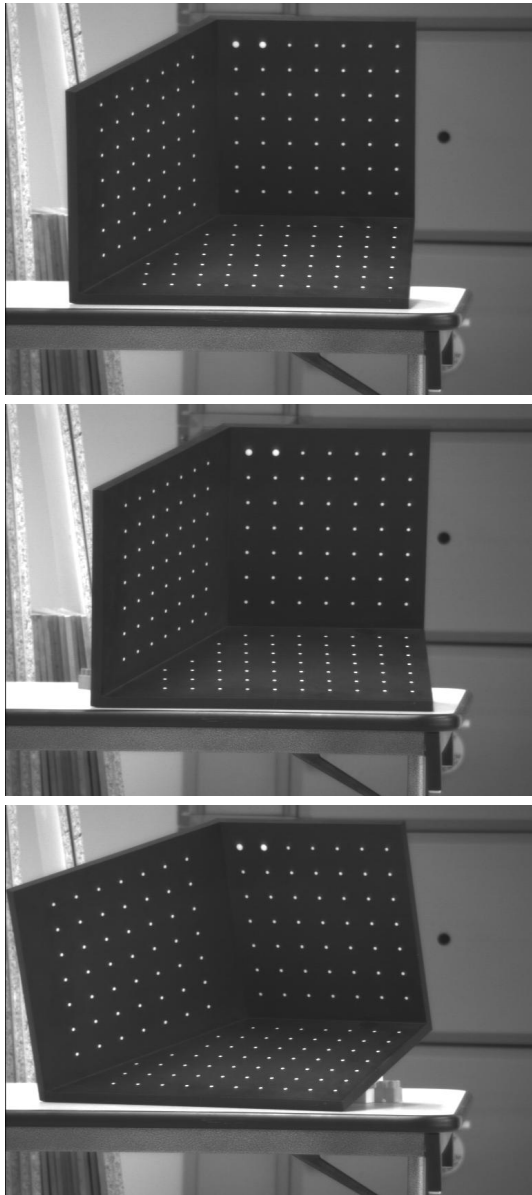


FIG. 4 – Les trois images de la mire.

En fait, comme la reconstruction est métrique, on peut espérer obtenir deux angles droits dans le modèle de la reconstruction. La figure 6 montre que ces deux angles apparaissent effectivement sur le modèle.

Expériences sur les images de la maison. On a aussi essayé l'algorithme sur des images d'une maison. On a pris trois images d'une maison, on a détecté des points d'intérêt à l'aide du détecteur de Harris, on les a mis en correspondances par corrélation croisée entre les images les plus



FIG. 5 – Les trois images de la maison avec les points mis en correspondances.

proches, et enfin on a utilisé ces résultats pour constituer des correspondances entre triplet de points, voir figure 5.

On peut voir qu'il y a de nombreuses mises en correspondances erronées, et que les images contiennent une certaine perspective, ce qui va fausser la reconstruction. On détecte dans un premier temps les outliers avec l'algorithme RAN-SAC, ceci afin de les éliminer. Ensuite, on calcule un modèle qui minimise l'erreur de reprojection sur 100 essais. Pour le modèle obtenu, voir la figure 7.

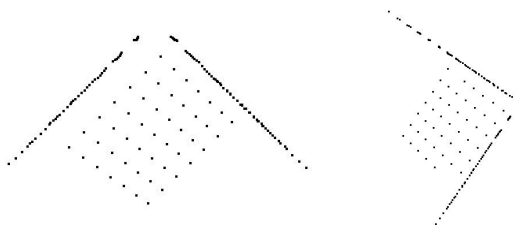


FIG. 6 – Les deux angles droit du modèle reconstruit.

5 Conclusion

Nous avons présenté une nouvelle méthode pour la reconstruction métrique à partir de caméras affines. La méthode fonctionne avec des données minimales, c'est-à-dire 4 points dans 3 images. Les coordonnées des 4 points 3D sont calculées de manière exacte et ceci directement, sans passer au préalable par les paramètres de mouvement des caméras comme on le fait couramment. On a aussi étudié les cas instables dans le cas d'une reconstruction à partir de données minimales. Les résultats expérimentaux prouvent que la méthode est stable. Un autre point important est que la méthode ne nécessite que le calcul des zéros d'un polynôme de degré 2, ce qui est très rapide. D'autre part, l'ajustement de faisceaux allonge considérablement le temps de calcul.

Références

- [1] K. Astrom, A. Heyden, F. Kahl, and M. Oskarsson. Structure and motion from lines under affine projections. In *Proceedings of the 7th International Conference on Computer Vision, Kerkyra, Greece*, pages 285–292, September 1999.
- [2] D. Cox, J. Little, and D. O'Shea. *Ideals, Varieties, and Algorithms*. Springer, 1996.
- [3] P.E. Gill, W. Murray, and M.H. Wright. *Practical Optimization*. Academic Press, 1981.
- [4] C.J. Poelman and T. Kanade. A paraperspective factorization method for shape and motion recovery. In J.O. Eklundh, editor, *Proceedings of the 3rd European Conference on Computer Vision, Stockholm, Sweden*, pages 97–108. Springer-Verlag, May 1994.
- [5] L. Quan. Self-calibration of an affine camera from multiple views. *International Journal of Computer Vision*, 19(1):93–105, May 1996.
- [6] L.S. Shapiro, A. Zisserman, and M. Brady. 3D motion recovery via affine epipolar geometry. *International Journal of Computer Vision*, 16(2):147–182, 1995.
- [7] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization method. *International Journal of Computer Vision*, 9(2):137–154, November 1992.
- [8] S. Ullman. *The Interpretation of Visual Motion*. The MIT Press, Cambridge, MA, USA, 1979.

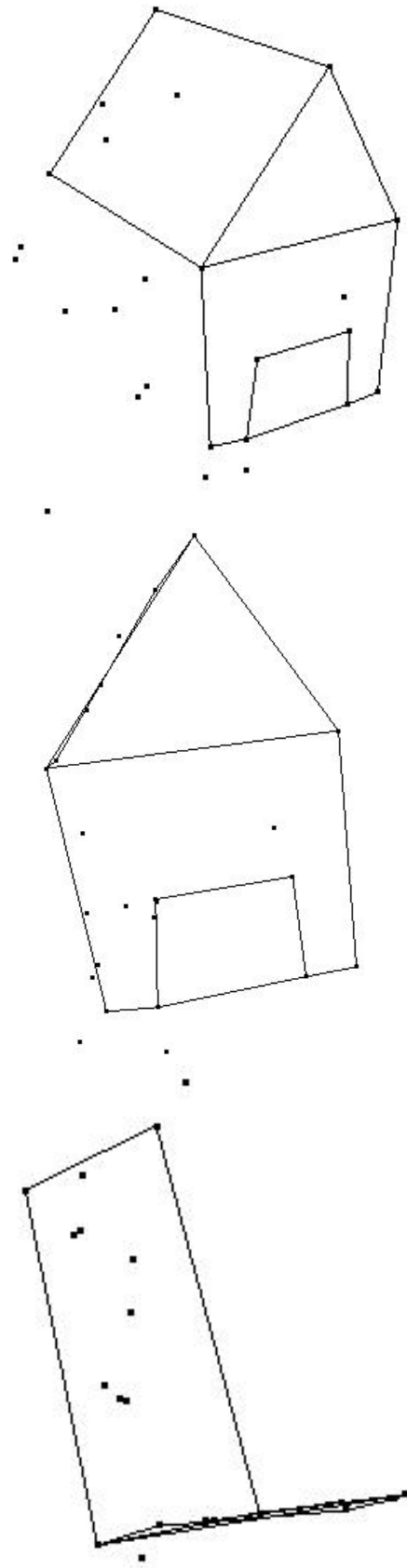


FIG. 7 – Le modèle reconstruit de la maison.