HAL
open science

Toolbox d'étalonnage pour Kinect : Application à la fusion d'une Kinect et d'un télémètre laser

Jean-Clément Devaux, Hicham Hadj-Abdelkader, Etienne Colle

To cite this version:

Jean-Clément Devaux, Hicham Hadj-Abdelkader, Etienne Colle. Toolbox d'étalonnage pour Kinect : Application à la fusion d'une Kinect et d'un télémètre laser. Reconnaissance de Formes et Intelligence Artificielle (RFIA) 2014, Jun 2014, France. hal-00989007

HAL Id: hal-00989007

https://hal.science/hal-00989007

Submitted on 9 May 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Toolbox d'étalonnage pour Kinect : Application à la fusion d'une Kinect et d'un télémètre laser

Jean-Clément Devaux[1] Hicham Hadj-Abdelkader[1] Etienne Colle[1]

[1] Laboratoire IBISC, Université d'Evry-Val-d'Essonne
40 rue du Pelvoux, CE1455 Courcouronnes, 91020 Evry Cedex, FRANCE

prénom.nom arobase ibisc.fr

Résumé

Dans la plupart des applications de robotique mobile, l'évitement d'obstacles est une tâche cruciale et doit être fiable. Fusionner plusieurs sources de données permet de détecter les obstacles quelle que soit leur forme. On utilise généralement des télémètres laser 2D pour assumer cette tâche mais au contraire des caméras 3D comme la Kinect, ils ne permettent pas de détecter des obstacles de forme complexe. Ce papier propose une méthode d'étalonnage extrinsèque d'un télémètre laser avec une caméra 3D. La problématique réside dans le calcul de la transformation euclidienne entre chaque capteur. Dans ce papier nous montrons que la méthode d'étalonnage à base de damier (utilisée pour les caméras conventionnelles) peut être étendue aux caméras 3D. Dans une expérimentation réelle, nous montrons aussi le bénéfice d'une fusion entre le laser et la Kinect pour détecter de façon fiable les obstacles de forme complexe.

Mots Clef

Etalonnage extrinsèque, caméra 3D, caméra de profondeur, télémètre laser, Kinect.

Abstract

In most of indoor mobile robot navigation, obstacle avoidance is a crucial task and should be reliable. Fusion of sources of data can be used to detect any obstacle shapes. Laser range finder is usually used to deal with this task in presence of simple obstacles. wheread 3D sensors can be used for more complex obstacle detection. This paper proposes a calibration of a laser range finder coupled with a Kinect. The basic idea is to compute the Euclidian transformation between each sensor. In this paper, we show that chessboard methods used to calibrate color cameras can be extended to deal with 3D depth sensors. We show in a real experiment the benefit of the fusion based on the calibration in order to detect complex obstacles reliably.

Keywords

Extrinsic calibration, 3D cameras, depth cameras, laser range finder, Kinect.

1 Introduction

Une bonne connaissance de l'environnement est essentielle pour la plupart des applications de robotique mobile. L'évitement d'obstacles, notamment, se base très souvent sur des télémètres laser qui fournissent des informations fiables et précises. Mais l'efficacité de ce capteur dépend beaucoup du matériau, de la couleur ou de la forme de l'obstacle. Ainsi, les lasers 2D ne sont pas capables de détecter les tables et les chaises mais seulement leurs pieds. Quelques solutions sont proposées dans la littérature. Dans [1], les auteurs exploitent les données d'un télémètre 2D déployé sur une tourelle pan-tilt. D'autres utilisent plusieurs télémètres 2D [2, 3] ou même des télémètres 3D. Certains se sont intéressés à la fusion des télémètres 2D avec des systèmes de vision stéréoscopiques moins onéreux que les technologies laser 3D [4].

Le capteur Kinect de Microsoft (l'Asus XTion Pro ou la PrimeSense) est intéressant dans un contexte de robotique intérieure car il offre une description tridimensionnelle de l'environnement à un coût très faible. Mais si on le compare aux télémètres laser avec des champs de détection horizontaux supérieurs à 200°, la Kinect a un champ de vision très limité (environ 57°). De même, la portée d'une Kinect est limitée à moins de 500 cm alors que certains télémètres laser sont capables de fournir des données jusqu'à plusieurs dizaines de mètres. Dans la plupart des applications de robotique mobile en milieu intérieur, il sera donc intéressant de combiner les données issues de la Kinect avec celles issues d'un télémètre laser 2D.

Pour pouvoir réaliser cette fusion, la principale contrainte est de connaître la transformation euclidienne entre les repères des deux capteurs. Beaucoup de travaux se sont penchés sur l'étalonnage extrinsèque et intrinsèque des caméras couleurs. Z. Zhang [5] [6] a notamment proposé une méthode basée sur l'observation d'un damier implémentée notamment dans *Camera Calibration Toolbox for Matlab* par J.-Y. Bouguet [7].

Certains travaux se sont intéressés à l'étalonnage extrinsèque d'une caméra couleur et d'un télémètre laser lorsque le rayon était composé de lumière visible [8, 9]. S'agissant du cas où le rayon est invisible dans l'image, Q. Zhang et

R. Pless ont proposé dans [10] un étalonnage extrinsèque entre un télémètre laser 2D et une caméra couleur en utilisant un damier dessiné sur un plan. Notre papier étend cette méthode dans la section 2.1 et l'adapte au cas du capteur Kinect.

Ces travaux sont présentés comme suit : La section 2 présente le modèle des données et le processus d'étalonnage du système laser-Kinect. La section 3 présente et analyse les résultats avec des données réelles (non simulées) pour les différentes méthodes d'étalonnage laser-Kinect. Enfin, nous appliquons dans la section 4 les résultats d'étalonnage pour la fusion laser-Kinect pour un système de détection et d'évitement d'obstacle fiable à bas coût.

2 Etalonnage Laser-Kinect

2.1 Modélisation des capteurs

Comme le capteur de profondeur de la Kinect est construit autour d'une caméra IR, nous supposons qu'il peut être décrit par le modèle du sténopé. Soit la matrice de rotation $\mathbf{R}_K$ et le vecteur de translation $\mathbf{t}_K$ entre le répère de la Kinect $\mathcal{F}_K$ et le repère du monde $\mathcal{F}_W$. Une observation aura pour coordonnées dans le repère monde $\mathbf{p}_W = [x_W, y_W, z_W]^\top$, et peut être exprimée dans l'image de la caméra 3D comme un point de coordonnées $\mathbf{m}_K = [u_K, v_K]^\top$ en utilisant la projection du modèle du sténopé :

$$\mathbf{m}_K \sim \mathbf{K}_K(\mathbf{R}_K.\mathbf{p}_W + \mathbf{t}_K) \tag{1}$$

avec

$$\mathbf{K}_K = \begin{pmatrix} f_u & s_{uv} & c_u \\ 0 & f_v & c_v \\ 0 & 0 & 1 \end{pmatrix}$$

la matrice des paramètres intrinsèques de la caméra de profondeur où c_u et c_v sont les coordonnées du point principal et f_u et f_v sont les distance focales. Nous supposons s_{uv} négligeable puisque les pixels semblent carrés ou rectangulaires. Nous pouvons noter que ce modèle ne tient pas compte d'une quelconque distortion peu visible dans l'image de profondeur. Sans perte de généralité, nous supposerons que les repères de la Kinect et du monde sont confondus($\mathbf{R}_K = \mathbf{I_{3X3}}$ et $\mathbf{t}_K = [0, 0, 0]^\top$). L'équation (1) devient donc :

$$\mathbf{m}_K \sim \mathbf{K}_K.\mathbf{p}_K$$

où $\mathbf{p}_K = [x_K, y_K, z_K]^\top$ est le vecteur des coordonnées du point 3D observé. Puisque z_K est fourni par le capteur de profondeur ($z_K = \text{depth}(u_K, v_K)$), les composantes restantes peuvent être calculées comme $x_K = z_K \frac{u_K - c_u}{f_u}$ et $y_K = z_K \frac{v_K - c_v}{f_v}$.

Soit $\mathbf{R}$ et $\mathbf{t}$ respectivement une matrice de rotation et un vecteur de translation liant le repère Kinect $\mathcal{F}_K$ au repère laser $\mathcal{F}_L$. Le but de l'étalonnage extrinsèque entre la Kinect et le laser est d'estimer $\mathbf{R}$ et $\mathbf{t}$ qu'on appelle aussi *paramètres extrinsèques* dans le modèle d'étalonnage laser-Kinect.

La relation entre les vecteurs de coordonnées $\mathbf{p}_K$ et $\mathbf{p}_L$, expression d'un point 3D observé respectivement dans le repère Kinect et le repère laser est donnée par :

$$\mathbf{p}_K = \mathbf{R}.\mathbf{p}_L + \mathbf{t} \tag{2}$$

Soit (π) un plan défini par son vecteur normal unitaire $\mathbf{n}$ et sa distance d exprimés dans le repère Kinect $\mathcal{F}_K$. Lorsque les points 3D sont situés sur le plan (π), nous pouvons déduire la contrainte suivante en utilisant (2) :

$$\mathbf{n}^\top \mathbf{p}_K - d = \mathbf{n}^\top(\mathbf{R}\mathbf{p}_L + \mathbf{t}) - d = 0 \tag{3}$$

Supposons maintenant que le point 3D considéré appartient aussi au plan laser, il est alors situé sur le segment représentant l'intersection entre le plan (π) et le plan laser. Le vecteur de coordonnées $\mathbf{p}_L$ est issu des données du laser. Les paramètres du plan $\mathbf{n}$ et d peuvent être estimés à l'aide des données de la Kinect comme décrit dans la section suivante. $\mathbf{R}$ et $\mathbf{t}$ peuvent alors être estimés en utilisant la contrainte décrite par (3).

Comme le laser 2D effectue seulement des mesures sur un plan, nous pouvons fixer le repère laser $\mathcal{F}_L$ tel que le plan laser est défini par le plan contenant les axes x et y. Par conséquent, le vecteur $\mathbf{p_L}$ peut être réduit en un vecteur de coordonnées homogènes dans $\mathbb{R}^2$: $\tilde{\mathbf{p}}_L = [X_L, Y_L, 1]^\top$ et (3) devient :

$$\mathbf{n}^\top \mathbf{H}\tilde{\mathbf{p}}_L - d = 0 \tag{4}$$

où $\mathbf{H} = \mathbf{R}\begin{pmatrix} 1 & 0 & \\ 0 & 1 & \mathbf{t} \\ 0 & 0 & \end{pmatrix}$.

2.2 Modélisation des données

Pour résoudre le problème d'étalonnage, l'image de profondeur et les données du laser sont utilisées conjointement. Les paramètres du plan ($\mathbf{n}$ et d) peuvent être extraits de l'image de profondeur fournie par le capteur Kinect alors que le vecteur de coordonnées $\tilde{\mathbf{p}}_L$ peut être calculé en utilisant les données issues du télémètre laser. Nous supposons que les régions d'intérêt (le plan) sont sélectionnées manuellement par l'utilisateur. Généralement quelques points de la sélection sont en dehors de la zone d'intérêt. Ces points peuvent être rejetés par un algorithme de régression robuste basé sur RANSAC et nous pouvons ainsi obtenir une bonne estimation des paramètres du plan. La figure 1 montre une représentation d'un plan extrait à partir d'un nuage de points 3D fourni par la Kinect (échelle en cm). Les points bleus représentent la projection des points sélectionnés par l'utilisateur à l'aide du modèle obtenu après regression avec RANSAC. Sur la partie droite de la figure, le même nuage de points est représenté avec un autre point de vue pour mieux voir la précision de la régression malgré les nombreux points perturbateurs. La régression robuste permet à l'utilisateur de sélectionner rapidement le plan d'étalonnage dans les images de profondeur sans craindre une perturbation par l'inclusion de points non désirés. De plus, une régression basée sur RANSAC permet

de filtrer avec efficacité le bruit issu de la Kinect et améliore la résolution des mesures de profondeur. Ce phénomène est particulièrement visible sur la figure 2 qui montre la discrétisation de la profondeur ainsi que la régression par RANSAC.

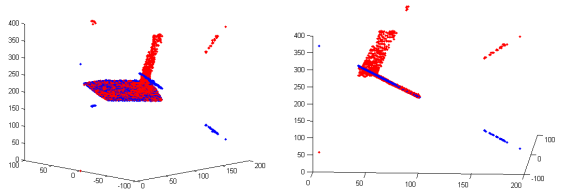


FIGURE 1 – Régression par RANSAC (en bleu) sur un nuage de points 3D issus de la Kinect (en rouge) avec de nombreux points perturbateurs

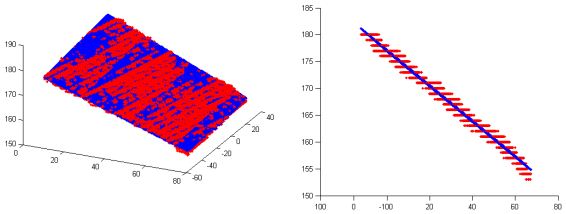


FIGURE 2 – Régression par RANSAC (en bleu) sur un nuage de points 3D issus de la Kinect (en rouge) montrant la résolution de la profondeur

Les données des télémètres laser dépendent fortement de la couleur de l'objet détecté. Un damier en noir et blanc présente ainsi des profondeurs variables. Nous appellerons ce phénomène *créneaux*. Il serait préférable d'utiliser un plan monochromatique mais cela ne permettrait pas d'inclure les caméras conventionnelles dans ce processus d'étalonnage. La figure 4 montre un exemple de différents damiers à différentes orientations et distances. Comme nous pouvons le voir sur la partie gauche de la figure, les créneaux sont plus marqués à mesure que l'angle et la distance augmentent. De plus, à partir d'un certain angle (e.g. 30°) nous faisons face à une perte de données le long du segment (figure en bas à gauche).
Puisque les alternances de noir et de blanc ont un effet néfaste sur les données du laser nous proposons un damier modifié (figure 3 à droite) pour améliorer les données du laser comme le montre la figure 4 à droite. Nous ajoutons une bande blanche à l'intersection entre la plan laser et le plan étalon. La hauteur et la taille de cette bande dépendent fortement de la position du laser sur le robot ainsi que de l'inclinaison maximum du plan lors de l'acquisition des données. Comme le montre la figure 3, nous ajoutons des marqueurs visuel pour conserver la possibilité d'étalonner une caméra conventionnelle et enfin, nous conservons le damier dans la partie basse où il n'y a jamais d'intersection avec le plan laser. Cela permet d'améliorer la précision d'un étalonnage avec une caméra conventionnelle puisque nous ajoutons plus de point sur l'axe vertical.

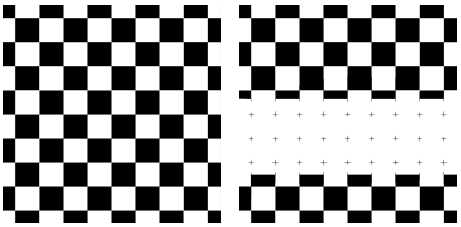


FIGURE 3 – Plan damier pour étalonner les caméras couleurs (gauche) et plan modifié pour une meilleure qualité des données laser (droite)

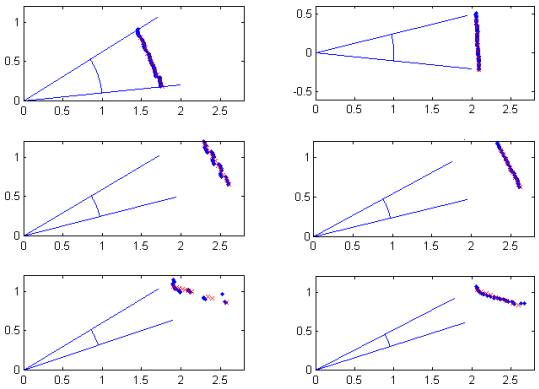


FIGURE 4 – Données laser brutes (bleu) avec régression (rouge) (plan cartésien en mètres), colonne de gauche avec le damier, colonne de droite avec le plan modifié

Puisque le damier est un plan, le laser doit le détecter comme un segment. Nous corrigeons donc les mesures du laser en utilisant une régression basée sur les moindres carrés pour obtenir n points $\tilde{\mathbf{p}}_L$ à introduire dans l'équation (4).

2.3 Etalonnage

Cette section décrit et compare différentes méthodes d'étalonnage (une résolution linéaire et plusieurs optimisations non-linéaires). La solution linéaire est utilisée pour initialiser les méthodes non linéaires utilisant l'algorithme de Levenberg-Macquardt.

Solution linéaire. Pour résoudre la contrainte (4), les données sont composées de n points pour chacun des m échantillons capturés à différentes positions et orientations du plan étalon. Nous devons estimer 9 paramètres inconnus et nous avons donc besoin de 9 équations pour résoudre le problème. Un plus grand nombre d'équations permettra de réduire l'influence du bruit des capteurs. Ce problème peut être décrit comme m equations de la forme :

$$(\mathbf{n}_j^\top \otimes \mathbf{P}_{L_{ij}}^\top)\mathbf{h} = \mathbf{d}_j \otimes \mathbb{1}_n \tag{5}$$

où $\otimes$ est le produit de Kronecker, $i = 1 \cdots n$ le nombre de points, $j = 1 \cdots m$ le nombre de plans et $\mathbb{1}_n$ un vecteur colonne de taille n dont tous les éléments vallent

1. $\mathbf{P}_{L_{ij}}$ est la matrice de n vecteurs de coordonnées $\tilde{\mathbf{p}}_{L_i}$ empilés sur n lignes. Le vecteur colonne solution $\mathbf{h} = [h_{11}, h_{12}, h_{13}, h_{21}, h_{22}, h_{23}, h_{31}, h_{32}, h_{33}]^\top$ contient les éléments d'une matrice estimée $\widehat{\mathbf{H}}$.
Par construction de $\mathbf{H}$, nous pouvons extraire la matrice $\widehat{\mathbf{R}}$ et le vecteur $\widehat{\mathbf{t}}$ à partir de $\widehat{\mathbf{h}}_\mathbf{k}$, $(k = 1 \cdots 3)$:

$$\widehat{\mathbf{R}} = [\widehat{\mathbf{h}}_\mathbf{1}, \widehat{\mathbf{h}}_\mathbf{2}, \widehat{\mathbf{h}}_\mathbf{1} \times \widehat{\mathbf{h}}_\mathbf{2}] \tag{6}$$

$$\widehat{\mathbf{t}} = \widehat{\mathbf{R}}\widehat{\mathbf{h}}_\mathbf{3} \tag{7}$$

Selon [11], la SVD peut être utilisée pour approximer la matrice de rotation $\mathbf{R}$ à partir de la matrice dégénérée $\widehat{\mathbf{R}}$ grâce à la matrice $\widetilde{\mathbf{R}}$:

$$\widehat{\mathbf{R}} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$$
$$\mathbf{R} \approx \widetilde{\mathbf{R}} = \mathbf{U}\mathbf{V}^\top$$
$$\mathbf{t} \approx \widetilde{\mathbf{R}}\mathbf{h}_\mathbf{3}$$

A cause de la construction de $\mathbf{H}$ nous avons 9 paramètres à estimer au lieu de 6 (3 rotations et 3 translations). Chaque segment laser fournit deux points linéairement indépendants. Il faut donc minimum 5 plans pour résoudre (5).

Optimisation non linéaire de $\mathbf{t}$. A l'observation de la reprojection des données Kinect dans le repère laser, nous pouvons dire que la matrice $\widetilde{\mathbf{R}}$ est plutôt bien estimée. Nous commencerons donc par l'optimisation seule de $\widetilde{\mathbf{t}}$.
Le but est de minimiser l'erreur de projection des points du laser dans le repère Kinect. Nous utilisons la distance orthogonale au plan comme métrique pour optimiser $\breve{\mathbf{t}}$:

$$min \sum_{j=1}^{m}\sum_{i=1}^{n} \left\| \mathbf{n}_j.(\widetilde{\mathbf{R}}\mathbf{p}_{L_{ij}} + \breve{\mathbf{t}}) - d_j \right\|^2 \tag{8}$$

Optimisation non linéaire de$\mathbf{R}$ et $\mathbf{t}$ basée sur $\mathbf{H}$. Pour confirmer si l'estimation linéaire de la rotation est correcte nous optimisons maintenant la rotation et la translation en même temps. En utilisant la même métrique que pour (8) et la même décomposition de $\mathbf{H}$ que dans (4), le problème devient un problème de minimisation sous contraintes :

$$\min_{\breve{\mathbf{H}}} \sum_{j=1}^{m}\sum_{i=1}^{n} \left\| \mathbf{n}_j.\breve{\mathbf{H}}\mathbf{p}_{Lij} - d_j \right\|^2$$
$$s.c. \left\| \breve{\mathbf{h}}_\mathbf{1} \right\| = \left\| \breve{\mathbf{h}}_\mathbf{2} \right\| = 1 \; et \; \breve{\mathbf{h}}_\mathbf{1}.\breve{\mathbf{h}}_\mathbf{2} = 0$$

Les contraintes permettent d'obtenir une matrice de rotation bien formée à partir de (6) puisqu'elles font de $\breve{\mathbf{h}}_\mathbf{k}$ des vecteurs orthonormés. $\breve{\mathbf{R}}$ et $\breve{\mathbf{t}}$ peuvent donc être extraits de $\breve{\mathbf{H}}$ de la même façon que (6) et (7).

Métrique du rayon, *"beam"*. Dupont et al. dans [12] ont introduit une métrique différente qui tient compte de la distance réelle parcourue par le rayon laser jusqu'au plan. Nous utiliserons cette métrique pour comparer avec la métrique orthogonale. Cette métrique d_b est donnée par :

$$d_b = \left\| \mathbf{p}_K - \mathbf{s} + \overrightarrow{\mathbf{r}} e \right\|$$

où $\mathbf{s}$ est l'origine de $\mathcal{F}_L$ exprimée dans $\mathcal{F}_K$), $\overrightarrow{\mathbf{r}}$ est le vecteur représentant le rayon et e est le temps de collision entre le plan et la source :

$$\mathbf{s} = \mathbf{R}\mathbf{t}$$
$$\overrightarrow{\mathbf{r}} = \mathbf{p_K} - \mathbf{s}$$
$$e = \frac{d - \mathbf{n}.\mathbf{s}}{\mathbf{n}.\overrightarrow{\mathbf{r}}}$$

Le problème de minimisation (8) devient donc :

$$\min_{\mathbf{R},\mathbf{t}} \sum_{j=1}^{m}\sum_{i=1}^{n} d_{b_{ij}}^2(\mathbf{R}, \mathbf{t}).$$

3 Résultats d'étalonnages

3.1 Configuration

Dans cette expérimentation, nous étalonnons un télémètre laser 2D à bas prix avec une Kinect et une caméra couleur montées sur notre robot mobile (voir figure 5). Nos jeux de données sont tous issus de cette configuration réelle d'utilisation.

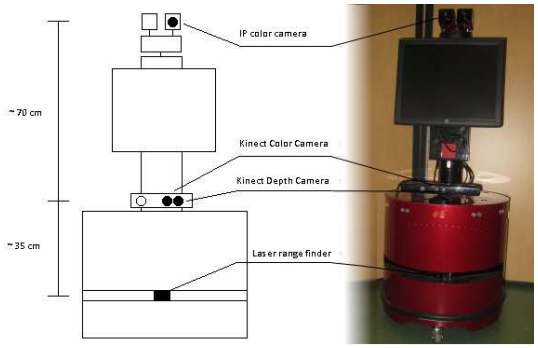


FIGURE 5 – Position des capteurs sur le robot Lina (Droïds Company)

Nous considérons deux jeux de données. Le premier est composé de 16 plans de 80×80 cm avec damier (cases de 89×89 mm) capturés à différentes distances et orientations. Le second est composé de la même façon mais en utilisant le motif d'étalonnage modifié.

3.2 Résultat d'étalonnage Laser-Kinect

Pour comparer les différentes méthodes, nous présentons les moyennes et écarts-types pour les deux métriques (orthogonale et rayon) dans la figure 6 et pour chaque jeu de données.
La figure 6 montre que les étalonnages utilisant l'une ou l'autre métrique produisent approximativement les mêmes résultats. Dans les deux cas, la méthode linéaire produit des résultats significativement moins bons que les optimisations non linéaires, ce qui était attendu.
Il est flagrant que l'utilisation du plan modifié améliore significativement les résultats d'étalonnage quelles que

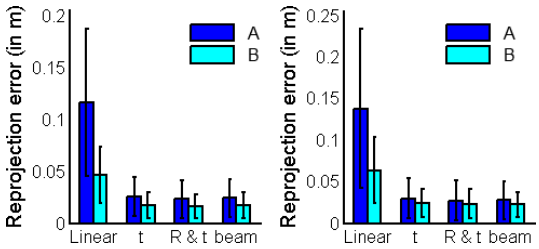


FIGURE 6 – Moyennes et écarts-types de l'erreur de reprojection utilisant les distances orthogonale (gauche) et du rayon (droite) pour le damier (A) et le plan modifié (B)

soient la métrique ou la méthode utilisées. Ceci est particulièrement vrai pour la solution linéaire ce qui permet d'obtenir une bonne initialisation des méthodes non linéaires et de réduire le risque de tomber dans un minimum local et améliore le temps d'optimisation.

Le tableau 1 (en cm) compile les moyennes, écarts-types, minima et maxima pour chaque métrique et chaque méthode. Le premier tableau présente les résultats obtenus avec le damier traditionnel alors que le tableau suivant montre les résultats obtenus avec l'étalon modifié. Comme nous pouvons l'observer sur le figure 6, la différence entre les méthodes non linéaires est inférieure au cm. Cela confirme la bonne estimation de R par la méthode linéaire. Les cases en vert représentent les meilleures valeurs pour la catégorie considérée.

TABLE 1 – Erreurs de reprojection en cm

Damier traditionnel

Stat	Reproj.	Linear	t opt.	R & t opt.	beam opt.
Mean	Orthog.	11.62	2.55	2.33	2.42
	Beam	13.80	3.00	2.71	2.79
Std	Orthog.	7.11	1.92	1.85	1.79
	Beam	9.54	2.43	2.45	2.29
Min	Orthog.	0.03	0.03	0.03	0.00
	Beam	0.03	0.03	0.03	0.00
Max	Orthog.	26.01	7.92	10.58	8.81
	Beam	41.01	12.49	15.64	13.23

Etalon modifié

Stat	Reproj.	Linear	t opt.	R & t opt.	beam opt.
Mean	Orthog.	4.66	1.75	1.64	1.71
	Beam	6.37	2.37	2.30	2.23
Std	Orthog.	2.74	1.27	1.16	1.27
	Beam	4.01	1.69	1.77	1.54
Min	Orthog.	0.51	0.02	0.04	0.02
	Beam	0.15	0.02	0.04	0.02
Max	Orthog.	12.35	6.60	4.89	6.66
	Beam	16.91	8.05	7.71	6.80

Nous présentons dans le tableau 2 les résultats de tests de Kruskal-Wallis afin de confirmer la significativité des résultats. Nous pouvons remarquer que la P-value est toujours inférieure à 10^{-6} entre la méthode linéaire et n'importe quelle autre méthode. L'optimisation de $\mathbf{t}$ seul ou de $\mathbf{R}$ et $\mathbf{t}$ quels que soient la métrique ou le plan étalon est toujours significativement plus précise que la solution linéaire. Le tableau 3 présente la différence entre l'utilisation du plan damier traditionnel et l'utilisation de l'étalon modifié. Le plan modifié est significativement plus performant avec la métrique orthogonale quelle que soit la méthode employée et significativement plus performant avec la métrique du rayon pour la méthode linéaire et la méthode d'optimisation de la distance du rayon.

TABLE 2 – Test de Kruskal-Wallis entre les méthodes

Traditional chessboard calibration plane

p > KW	Linear	t opt.	R & t opt.	beam opt.
Linear	X	$< 10^{-6}$	$< 10^{-6}$	$< 10^{-6}$
t opt.	X	X	0.3428	0.6856
R & t opt.	X	X	X	0.5203

Modified chessboard calibration plane

p > KW	Linear	t opt.	R & t opt.	beam opt.
Linear	X	$< 10^{-6}$	$< 10^{-6}$	$< 10^{-6}$
t opt.	X	X	0.6999	0.8438
R & t opt.	X	X	X	0.8514

TABLE 3 – Test de Kruskal-Wallis entre les motifs d'étalonnage

p > KW	Linear	t opt.	R & t opt.	beam opt.
Orthogonal	$< 10^{-6}$	$< 10^{-3}$	$< 10^{-2}$	$< 10^{-3}$
Beam	$< 10^{-6}$	0.7125	0.1242	$< 10^{-1}$

L'erreur induite par les méthodes non linéaires semble compatible avec le but final qui est de combiner les données du laser et de la Kinect. En particulier, la méthode non linéaire d'optimisation globale présente une erreur de 2.33 cm alors que la taille des créneaux est de 5.76 cm sur la figure 4. Cela signifie que l'étalonnage peut être plus précis que le capteur laser dans le cas précis d'une alternance de blanc et de noir. La Kinect n'étant pas sensible à cet effet, elle pourra donc venir améliorer les données du laser.

4 Evitement d'obstacles fiable

4.1 Modèle d'amélioration

Après avoir retiré le sol et le plafond des données de la Kinect, tous les objets détectés sont potentiellement des obstacles. Chaque point fourni par la Kinect est exprimé dans le repère du laser $(x_i \;\; y_i \;\; z_i)$ et projeté sur le plan laser. En coordonnées polaires, nous avons donc $(l_K, \; \theta_K)$:

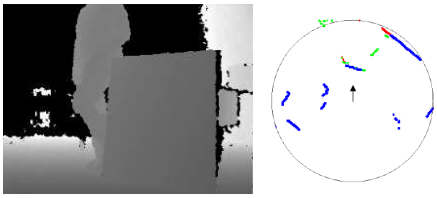
$$\theta_K = \tfrac{7\pi}{6} - \tan^{-1}(z_i/x_i) \quad and \quad l_K = \sqrt{x_i^2 + z_i^2}$$

$$\mathbf{d}(\theta) = \min\left(\mathbf{l}(\theta), l_K(\theta)\right)$$

Cette équation exprime un balayage laser avec $\mathbf{l}$ et $\mathbf{d}$ qui associent chacun une distance avec un angle respectivement pour le laser et pour la combinaison laser-Kinect.

4.2 Résultats expérimentaux

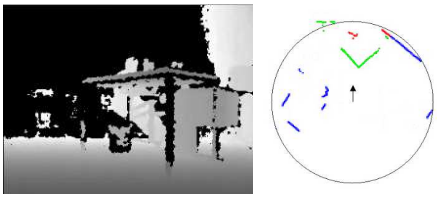
Dans la figure 7, nous montrons à gauche l'image de profondeur de la Kinect et à droite la carte contenant la combinaison laser-Kinect. Le cercle noir représente un rayon de 4m. Le bleu représente les obstacles détectés par le laser non modifié par l'adjonction de la Kinect. Le vert représente les obstacles détectés seulement par la Kinect alors que le rouge montre la position de l'obstacle correspondant détecté par le laser.



(a) Kinect depth image (b) Fusion

FIGURE 7 – Obstacle vertical. La flèche représente le centre et l'orientation du robot.

La figure 7 montre un cas où le laser et la Kinect détectent le même obstacle. La carte est peu améliorée par la Kinect. La figure8 montre le cas d'une table invisible par le laser. Comme le montre la carte, la Kinect détecte parfaitement le plan de la table et corrige les données du laser (en vert). Un évitement d'obstacle basé laser aurait mené à une collision directe avec la table. Cette situation est corrigée par l'ajout de la Kinect comme nous le montrons dans la vidéo attachée à ce papier.



(a) Kinect depth image (b) Fusion

FIGURE 8 – Scénario type : la table. La flèche représente le centre et l'orientation du robot.

5 Conclusion et perspectives

Nous avons présenté une méthode d'étalonnage basée sur un plan étalon entre un télémètre laser et une caméra 3D. Nous avons montré au travers d'expériences que cette méthode est suffisante pour améliorer les données du laser grâce à la Kinect en vue d'un évitement d'obstacles de forme complexe comme une table. Nous avons de plus proposé un motif étalon qui améliore significativement la précision d'étalonnage lorsqu'on utilise un télémètre laser à bas coût.

Notre extension de *Camera calibration toolbox for Matlab* [7] à l'étalonnage des caméras conventionnelles, des laser et des caméras 3D est disponible à `http://goo.gl/OlJmGX`.

Dans de futurs travaux, nous pourrons investiguer l'étalonnage de la Kinect en simplifiant le processus par une automatisation des sélections. Il serait aussi intéressant de se pencher sur la problématique d'étalonnage sur scène naturelle pour permettre un processus encore plus transparent pour l'utilisateur.

Références

[1] A. N. H. Surmann and J. Hertzberg, "An autonomous mobile robot with a 3d laser range finder for 3d exploration and digitalization of indoor environments," *Robotics and Autonomous Systems*, vol. 45, no. 3–4, pp. 181–198, December 2003.

[2] S. Thrun, W. Burgard, and D. Fox, "A real-time algorithm for mobile robot mapping with applications to multi-robot and 3d mapping," in *International Conference on Robotics and Automation*, ser. ICRA'00, 2000, pp. 321–328.

[3] C. Fruh and A. Zakhor, "3d model generation for cities using aerial photographs and ground level laser scans," in *Conference on Computer Vision and Pattern Recognition*, ser. CVPR'01. Kauai, HI, USA : IEEE Computer Society, 2001, pp. 31–38.

[4] S. Kumar, D. Gupta, and S. Yadav, "Sensor fusion of laser and stereo vision camera for depth estimation and obstacle avoidance," *International Journal of Computer Applications*, vol. 1, no. 25, pp. 20–25, February 2010.

[5] Z. Zhang, "Flexible camera calibration by viewing a plane from unknown orientations," in *International Conference on Computer Vision*, vol. 1, Corfu, Greece, September, 20-25 1999, p. 666.

[6] ——, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 1330–1334, 2000.

[7] J.-Y. Bouguet, "Camera calibration toolbox for matlab," 2003.

[8] O. Jokinen, "Self-calibration of a light striping system by matching multiple 3-d profile maps," in *International conference on 3-D digital imaging and modeling*, ser. 3DIM'99. Washington, DC, USA : IEEE Computer Society, 1999, pp. 180–190.

[9] A. M. McIvor, "Calibration of a laser stripe profiler," in *International conference on 3-D digital imaging and modeling*, ser. 3DIM'99. Washington, DC, USA : IEEE Computer Society, 1999, pp. 92–98.

[10] Q. Zhang and R. Pless, "Extrinsic calibration of a camera and laser range finder (improves camera calibration)," in *International Conference on Intelligent Robots and Systems*, ser. IROS '04, vol. 3, Oct. 2004, pp. 2301–2306.

[11] G. H. Golub and C. F. Van Loan, *Matrix computations (3rd ed.)*. Baltimore, MD, USA : Johns Hopkins University Press, 1996.

[12] R. Dupont, R. Keriven, and P. Fuchs, "An improved calibration technique for coupled single-row telemeter and ccd camera," in *International Conference on 3-D Digital Imaging and Modeling*, ser. 3DIM '05. Washington, DC, USA : IEEE Computer Society, 2005, pp. 89–94.