

Estimation de mouvement par appariement de droites 2D dans des images de distance

Motion Estimation by 2D Straight Lines Matching in Range Images

G. A. Borges*

M-J. Aldon

O. Strauss

Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier - LIRMM
161, rue ADA. 34392 - Montpellier - Cedex 5 - France
{borges,aldon,strauss}@lirmm.fr

Résumé

Cet article présente une méthode de mise en correspondance de droites dans des images de distance fournies par un télémètre laser à balayage azimutal. L'objectif est d'estimer le mouvement relatif 2D du capteur (ou du véhicule qui le transporte), c'est à dire le changement de position et de cap entre deux images successives. L'algorithme réalise simultanément et de manière itérative l'appariement d'un nombre minimum de droites significatives extraites de ces images, et l'estimation du mouvement. Il est basé sur la minimisation, par la méthode des multiplicateurs de Lagrange, d'une fonction de distance soumise à une contrainte d'égalité. Sa robustesse est due à la mise en œuvre d'un mécanisme de réjection des appariements aberrants. Une simulation, puis des expérimentations réalisées dans des environnements réels illustrent le comportement et les bonnes performances de la méthode.

Mots clef

Mise en correspondance de droites, analyse du mouvement, traitement de séquences d'images de distance.

Abstract

This paper presents a method for matching straight lines in range images provided by a 2D laser radar. It aims to estimate the 2D relative motion of the sensor, i.e. the change of position and heading between two consecutive images. The algorithm simultaneously and iteratively matches a minimum number of significant lines and estimates the motion. Lagrange multipliers are used to achieve the constrained minimization of a distance function. False matchings are rejected in order to increase the robustness of the method. Results

from simulation and real experiments are provided to illustrate the performance of this algorithm.

Keywords

Matching of straight lines, motion analysis, processing of range images.

1 Introduction

Le développement de systèmes de perception extéroceptifs pour les véhicules autonomes a conduit à la mise au point de méthodes de navigation dites *références terrain* qui utilisent des images vidéo ou des images de distance pour résoudre les problèmes de localisation. Ces solutions ne nécessitent pas d'équipement particulier de l'environnement. C'est le cas des méthodes qui exploitent des cartes 2D délivrées par des télémètres laser pour localiser des véhicules d'intérieur navigant sur sol plan. Elles sont basées sur l'appariement des mesures courantes soit avec un modèle de référence connu *a priori* (localisation absolue) soit avec un modèle obtenu lors d'une mesure précédente (localisation relative).

Les méthodes de localisation absolue développées jusqu'à présent utilisent généralement comme modèle de référence une représentation polygonale de l'environnement. Elles sont basées sur la mise en correspondance avec ce modèle de données locales acquises en ligne qui sont : soit les points de mesure bruts fournis par un télémètre, soit des primitives géométriques issues de la segmentation de ces mesures [1]. Ainsi, le système de localisation du robot Blanche [2] ne fait appel à aucun prétraitement. Il utilise une mise en correspondance de chaque point de mesure avec un segment-cible de la carte de référence. L'appariement optimal est celui qui minimise la somme des carrés des écarts des points aux segments-cibles associés. Les méthodes

*Boursier de la CAPES - Brasília/Brésil

basées sur l'appariement de primitives géométriques différent suivant le type de primitives choisies. On peut mettre en correspondance: (i) des points caractéristiques du modèle, comme par exemple les sommets de l'approximation polygonale [3]; (ii) des segments de droite qui correspondent aux limites de l'espace libre [4]; (iii) des amers constitués de segments connexes qui décrivent des éléments particuliers de l'environnement (coins, portes, ...) [5]. Par ailleurs, dans la plupart des algorithmes proposés, la procédure d'appariement est initialisée grâce aux mesures odométriques disponibles sur le véhicule. Ces mesures étant imprécises, on met à jour la position du robot grâce à un filtrage numérique (filtre de Kalman) en utilisant uniquement les appariements les plus certains [6]. Un test sur la distance de Mahalanobis permet ensuite de valider les appariements restants.

Lorsque les appariements sont effectués entre deux images laser acquises en ligne, il s'agit de localisation relative, ou encore d'estimation du mouvement entre ces deux images. Par exemple, Gonzalez *et al.* [7] apparie directement les points de mesure par une technique de type *flot optique* appliquée à des images de distance. Lu *et al.* [8] ont développé un algorithme d'optimisation pour estimer le mouvement relatif entre deux images de distance à partir de l'appariement itératif des points de mesure.

Notre objectif est de mettre au point une méthode de localisation relative, autrement dit d'estimation du mouvement entre deux images de distance successives, qui possède les propriétés suivantes :

- Elle s'applique à tout type de mouvement plan. La seule contrainte à respecter est l'existence d'un espace d'observation commun aux deux images;
- Elle est exploitable dans des environnements structurés, partiellement évolutifs (présence d'objets en mouvement dans les images), et non modélisés a priori.

La solution que nous avons choisie est basée sur la mise en correspondance d'indices de type *droite*. La notion de segment n'est pas utilisée ici, car nous considérons que les extrémités des droites observées dans l'image ne représentent pas obligatoirement des points caractéristiques de l'environnement. Elles correspondent souvent aux limites des zones occultées ou parfois aux points extrêmes du secteur balayé par le capteur. Notre méthode suppose une segmentation préalable de l'image que nous ne présenterons pas dans le cadre de cette publication. Nous nous attacherons ici à décrire l'algorithme de mise en correspondance. Nous montrerons comment il permet, en sélectionnant un nombre minimum d'appariements fiables, d'obtenir de manière

robuste la rotation et la translation entre deux images consécutives.

2 Définition du problème

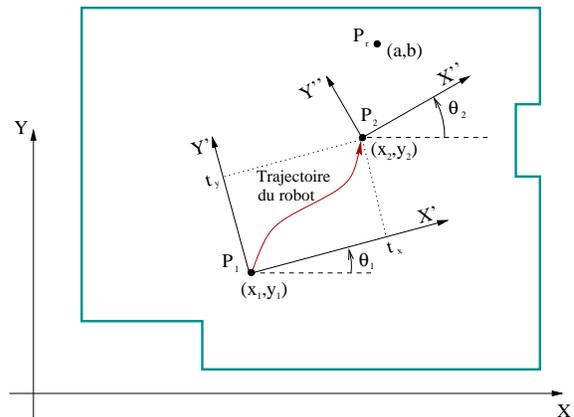


FIG. 1: Références de positionnement d'un robot mobile.

Dans ce travail, nous considérons un robot mobile équipé d'un télémètre laser 2D. Ce robot évolue dans un environnement intérieur comportant des surfaces planes qui se traduisent par la présence de segments de droites dans l'image télémétrique. La localisation absolue de ce robot par rapport à l'environnement est complètement décrite par $P = (x, y, \theta)$, où (x, y) représentent les coordonnées d'un point de référence sur le robot et θ l'angle d'orientation du robot. La Figure 1 montre le mouvement réalisé par le robot entre deux positions absolues $P_1 = (x_1, y_1, \theta_1)$ et $P_2 = (x_2, y_2, \theta_2)$. Les images lidar correspondantes I_1 et I_2 ont comme repères de référence respectifs (X', Y') et (X'', Y'') , centrés en P_1 et P_2 . Pour définir le mouvement relatif, nous considérons un point de référence $P_r = (a, b)$ fixe dans le repère absolu (X, Y) . Soient (a', b') et (a'', b'') les coordonnées de P_r dans les repères (X', Y') et (X'', Y'') . Le mouvement relatif entre P_1 et P_2 est décrit par un vecteur de translation $\mathbf{t} = \{t_x, t_y\}^T$ et un angle de rotation ω qui satisfont :

$$\begin{pmatrix} a'' \\ b'' \end{pmatrix} = R(\omega) \left\{ \begin{pmatrix} a' \\ b' \end{pmatrix} - \mathbf{t} \right\}, \quad (1)$$

$$\theta_2 = \theta_1 + \omega, \quad (2)$$

avec

$$R(\omega) = \begin{pmatrix} \cos(\omega) & \sin(\omega) \\ -\sin(\omega) & \cos(\omega) \end{pmatrix}. \quad (3)$$

Nous cherchons à estimer les paramètres de mouvement \mathbf{t} et ω du robot entre P_1 et P_2 à partir de mesures que sont les images de distance I_1 et I_2 . Notre approche est basée sur l'appariement de droites extraites

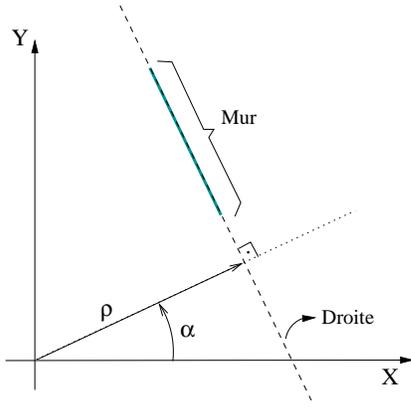


FIG. 2: Paramètres polaires d'une droite.

par segmentation des images I_1 et I_2 . Deux droites sont correspondantes si elles ont été générées par un même objet de l'environnement. Nous considérons seulement le cas où l'objet est fixe dans le repère absolu. Si un objet a réalisé un mouvement non négligeable, l'estimateur de mouvement ne doit pas prendre en compte ses images dans la procédure d'appariement. Nous considérons par exemple un mur comme un obstacle statique. Il est vu comme une droite dans les positions P_1 et P_2 du robot. Dans les repères (X', Y') et (X'', Y'') ces droites sont représentées par les paramètres polaires $\mathbf{d}' = (\rho', \alpha')^T$ et $\mathbf{d}'' = (\rho'', \alpha'')^T$. La Figure 2 définit cette représentation pour une droite quelconque $\mathbf{d} = (\rho, \alpha)^T$. Si \mathbf{d}' et \mathbf{d}'' sont correspondantes, nous pouvons montrer à partir des équations (1) et (2) que les paramètres de ces droites sont liés par la relation :

$$\begin{pmatrix} \rho'' \\ \alpha'' \end{pmatrix} = \begin{pmatrix} \rho' - t_x \cos(\alpha') - t_y \sin(\alpha') \\ \alpha' - \omega \end{pmatrix}. \quad (4)$$

L'équation (4) est vérifiée pour tous les droites \mathbf{d}' extraites de l'image I_1 qui sont en correspondance avec les droites \mathbf{d}'' de l'image I_2 . Elle montre que pour vérifier la correspondance de deux droites, il faut connaître non seulement les paramètres polaires des droites, mais aussi les paramètres du mouvement \mathbf{t} et ω . Néanmoins, comme la phase d'estimation de \mathbf{t} et ω est réalisée après la phase de mise en correspondance, l'appariement initial des droites doit utiliser des estimations $\hat{\mathbf{t}}$ et $\hat{\omega}$ qui sont données par un autre capteur ou par une autre méthode. En général, sur un robot à roues, $\hat{\mathbf{t}}$ et $\hat{\omega}$ sont obtenus par odométrie. Ces mesures peuvent être dégradées par des erreurs non-systématiques [9]. Dans ce cas, il peut y avoir de mauvaises mises en correspondance et l'estimation de mouvement est faussée.

Nous proposons un algorithme itératif qui réalise la

mise en correspondance et l'estimation de mouvement de façon alternée, et utilise seulement une estimation initiale de ω . Cette initialisation de ω peut être fournie par un capteur plus fiable comme par exemple un gyromètre laser.

3 Présentation de l'algorithme

Nous considérons qu'à partir des images télémétriques I_1 et I_2 , un algorithme de segmentation a généré deux listes de droites $L_1 = \{\mathbf{d}_1, \dots, \mathbf{d}_i, \dots, \mathbf{d}_{N_1}\}$ et $L_2 = \{\mathbf{d}_1, \dots, \mathbf{d}_j, \dots, \mathbf{d}_{N_2}\}$. Ces listes de droites sont ordonnées par ordre décroissant du nombre du points que les constituent. Néanmoins, il n'y a pas encore d'appariement entre les droites \mathbf{d}_i de l'image I_1 et \mathbf{d}_j de l'image I_2 . La méthode présentée ici est inspirée de l'algorithme *fuzzy c-means* [10], et propose la minimisation itérative de la fonction de coût suivante :

$$V(\hat{\mathbf{t}}, \hat{\omega}) = \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \mu_{ij}^m D(\mathbf{d}_i, \mathbf{d}_j, \hat{\mathbf{t}}, \hat{\omega}), \quad (5)$$

où $\hat{\mathbf{t}} = (\hat{t}_x, \hat{t}_y)^T$ est l'estimation du vecteur de translation et $\hat{\omega}$ est l'estimation de l'angle de rotation. Comme dans l'algorithme *fuzzy c-means*, $\mu_{ij} \in [0, 1]$ est une mesure de la correspondance entre les droites \mathbf{d}_i et \mathbf{d}_j , et m est une constante. Nous avons choisi d'utiliser $m = 1,5$. D est une fonction qui mesure la distance ou l'écart entre la droite \mathbf{d}_j et la droite \mathbf{d}_i en fonction de $\hat{\mathbf{t}}$ et $\hat{\omega}$. Dans l'équation (5), le terme μ_{ij}^m permet de sélectionner uniquement les droites qui sont en correspondance.

Pour minimiser $V(\hat{\mathbf{t}}, \hat{\omega})$ nous employons la technique d'optimisation en alternance, qui comporte deux phases: l'estimation de μ_{ij} qui minimise $V(\hat{\mathbf{t}}, \hat{\omega})$ en considérant $\hat{\mathbf{t}}$ et $\hat{\omega}$ comme constants, et le calcul de $\hat{\mathbf{t}}$ et $\hat{\omega}$ qui minimisent $V(\hat{\mathbf{t}}, \hat{\omega})$ en considérant μ_{ij} constant. Pour le calcul de $\hat{\mathbf{t}}$ et $\hat{\omega}$, nous utilisons la fonction de distance suivante :

$$D(\mathbf{d}_i, \mathbf{d}_j, \hat{\mathbf{t}}, \hat{\omega}) = (\rho_j - \hat{\rho}_j)^2 + (\cos(\alpha_j) - \cos(\hat{\alpha}_j))^2 + (\sin(\alpha_j) - \sin(\hat{\alpha}_j))^2, \quad (6)$$

qui est basée sur l'équation (4), avec :

$$\hat{\rho}_j = \rho_i - \hat{t}_x \cos(\alpha_i) - \hat{t}_y \sin(\alpha_i), \quad (7)$$

$$\hat{\alpha}_j = \alpha_i - \hat{\omega}, \quad (8)$$

Le calcul de $\hat{\mathbf{t}}$ et $\hat{\omega}$ est obtenu par la résolution du système d'équations suivant :

$$\frac{\partial V(\hat{\mathbf{t}}, \hat{\omega})}{\partial \hat{\mathbf{t}}} = \mathbf{0}, \quad \frac{\partial V(\hat{\mathbf{t}}, \hat{\omega})}{\partial \hat{\omega}} = 0.$$

Ainsi, nous avons :

$$\hat{\mathbf{t}} = \mathbf{A}^{-1}\mathbf{b}, \quad (9)$$

$$\hat{\omega} = \tan^{-1} \left(\frac{-\sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \mu_{ij}^m \sin(\alpha_j - \alpha_i)}{\sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \mu_{ij}^m \cos(\alpha_j - \alpha_i)} \right) \quad (10)$$

dans lesquels:

$$\mathbf{A} = \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \mu_{ij}^m \begin{pmatrix} \cos^2(\alpha_j - \alpha_i) & \frac{1}{2} \sin(2(\alpha_j - \alpha_i)) \\ \frac{1}{2} \sin(2(\alpha_j - \alpha_i)) & \sin^2(\alpha_j - \alpha_i) \end{pmatrix},$$

$$\mathbf{b} = \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \mu_{ij}^m \begin{pmatrix} (\rho_i - \rho_j) \cos(\alpha_i) \\ (\rho_i - \rho_j) \sin(\alpha_i) \end{pmatrix}.$$

En considérant $\hat{\mathbf{t}}$ et $\hat{\omega}$ comme constants, nous allons déterminer les coefficients μ_{ij} qui qualifient la correspondance des droites \mathbf{d}_i et \mathbf{d}_j , tels que:

$$\sum_{i=1}^{N_1} \mu_{ij} = 1, \forall j = 1, \dots, N_2. \quad (11)$$

On retrouve cette contrainte dans l'algorithme *fuzzy c-means*. Ainsi, le paramètre μ_{ij} qui minimise $V(\hat{\mathbf{t}}, \hat{\omega})$ est également celui qui minimise la fonction de coût :

$$L(\hat{\mathbf{t}}, \hat{\omega}) = V(\hat{\mathbf{t}}, \hat{\omega}) + \sum_{j=1}^{N_2} \lambda_j \left(1 - \sum_{i=1}^{N_1} \mu_{ij} \right),$$

dans laquelle les λ_j sont des multiplicateurs de Lagrange. La minimisation de $L(\hat{\mathbf{t}}, \hat{\omega})$ prend également en compte la contrainte (11). Ainsi, μ_{ij} est calculé comme la solution du système d'équations suivant :

$$\frac{\partial L(\hat{\mathbf{t}}, \hat{\omega})}{\partial \mu_{ij}} = 0, \frac{\partial L(\hat{\mathbf{t}}, \hat{\omega})}{\partial \lambda_j} = 0.$$

Donc, nous avons :

$$\mu_{ij} = \frac{1}{\sum_{k=1}^{N_1} \left(\frac{D(\mathbf{d}_i, \mathbf{d}_j, \hat{\mathbf{t}}, \hat{\omega})}{D(\mathbf{d}_k, \mathbf{d}_j, \hat{\mathbf{t}}, \hat{\omega})} \right)^{\frac{1}{m-1}}}. \quad (12)$$

La contrainte (11) étant toujours respectée et D étant positif, alors $0 \leq \mu_{ij} \leq 1$. L'équation (12) définit μ_{ij} comme étant une mesure de correspondance de chaque droite j de la deuxième image avec toutes les droites de la première image. Si une droite j de la deuxième image n'a pas de vraie correspondante dans la première image, μ_{ij} sera plus proche de 1,0 pour la droite i qui donne le $D(\mathbf{d}_i, \mathbf{d}_j, \hat{\mathbf{t}}, \hat{\omega})$ minimum. Si la droite j donne à peu près la même valeur de D pour tous les droites i de la première image, μ_{ij} sera proche de $1/N_1$. La solution que nous proposons pour lever

cette ambiguïté est de calculer deux mesures de correspondance u_{ij} et v_{ij} telles que :

$$\sum_{i=1}^{N_1} u_{ij} = 1, \sum_{j=1}^{N_2} v_{ij} = 1. \quad (13)$$

Ces deux coefficients u_{ij} et v_{ij} sont calculés en se basant sur l'équation (12), de la façon suivante :

$$u_{ij} = \frac{1}{\sum_{k=1}^{N_1} \left(\frac{D(\mathbf{d}_i, \mathbf{d}_j, \hat{\mathbf{t}}, \hat{\omega})}{D(\mathbf{d}_k, \mathbf{d}_j, \hat{\mathbf{t}}, \hat{\omega})} \right)^{\frac{1}{m-1}}}, \quad (14)$$

$$v_{ij} = \frac{1}{\sum_{k=1}^{N_2} \left(\frac{D(\mathbf{d}_i, \mathbf{d}_j, \hat{\mathbf{t}}, \hat{\omega})}{D(\mathbf{d}_i, \mathbf{d}_k, \hat{\mathbf{t}}, \hat{\omega})} \right)^{\frac{1}{m-1}}}. \quad (15)$$

Le coefficient u_{ij} (rsp. v_{ij}) mesure la degré d'appartenance de la droite i de l'image I_1 (rsp. j de l'image I_2) à la classe définie par toutes les droites j de l'image I_2 (rsp. i de l'image I_1). Nous cherchons à obtenir une mesure de correspondance entre les droites i et j . C'est pourquoi nous définissons μ_{ij} ainsi :

$$\mu_{ij} = u_{ij} \star v_{ij}, \quad (16)$$

où \star représente une t-norme [11]. En effet, soit C_1^i (C_2^j) l'ensemble des droites de l'image I_2 (I_1) qui correspondent à la droite i (j) dans l'image I_1 (I_2). La proposition "la droite i et la droite j se correspondent" implique $i \in C_1^j$ et $j \in C_2^i$. Ceci se traduit par l'équation (16). La t-norme choisie est l'opérateur minimum: $\mu_{ij} = \min(u_{ij}, v_{ij})$.

Il faut souligner l'importance de la fonction D pour le calcul de u_{ij} et v_{ij} . Cette fonction doit permettre de bien distinguer les droites i et j qui ne sont pas en correspondance. Néanmoins, des simulations ont montré que la fonction D donnée par l'équation (6) n'était pas assez sélective. Nous avons donc choisi une autre mesure de distance F formulée ainsi :

$$F(\mathbf{d}_i, \mathbf{d}_j, \hat{\mathbf{t}}, \hat{\omega}) = e_j^2 + \hat{e}_j^2, \quad (17)$$

avec

$$e_j = \frac{\hat{\rho}_j}{\cos(\hat{\alpha}_j - \alpha_j)} - \rho_j, \quad (18)$$

$$\hat{e}_j = \frac{\rho_j}{\cos(\hat{\alpha}_j - \alpha_j)} - \hat{\rho}_j. \quad (19)$$

Comme le montre l'équation (17), F est composée de deux termes e_j et \hat{e}_j . e_j et \hat{e}_j sont des distances entre les droites $\mathbf{d}_j = \{\rho_j, \alpha_j\}$ et $\hat{\mathbf{d}}_j = \{\hat{\rho}_j, \hat{\alpha}_j\}$ (Figure 3), où les paramètres de $\hat{\mathbf{d}}_j$ sont donnés par les équations (7) et (8). Cependant, la fonction D utilisée pour calculer u_{ij} et v_{ij} dans les équations (14) et (15) doit être substituée par la fonction F .

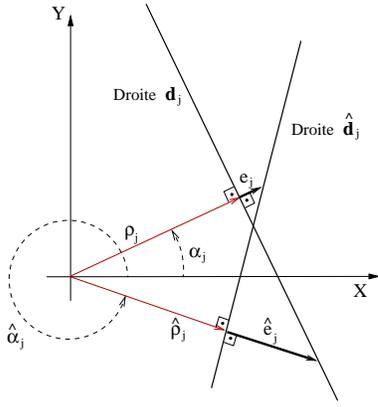


FIG. 3: Distances e_j et \hat{e}_j entre deux droites.

Les coefficients μ_{ij} représentent les appariements possibles entre les droites des images I_1 et I_2 . Mais, comme ces mesures sont relatives, il est encore possible que μ_{ij} indique un appariement (i, j) qui n'existe pas. Ce problème peut être résolu par un mécanisme d'élimination de droites des listes L_1 et L_2 . Nous proposons le mécanisme de sélection suivant basé sur les statistiques robustes [12]:

1. En premier lieu, on extrait les appariements les plus importants entre L_1 et L_2 , qui sont tels que $\mu_{ij} \geq \mu_{\min}$, et on obtient une liste G de n_v éléments $\mathbf{g}_k = \{(i, j)^T \mid \mu_{ij} \geq \mu_{\min}\}$, $k = 1, \dots, n_v$;
2. Pour la liste G , on crée un vecteur \mathbf{s} de n_v éléments où chaque élément contient la valeur de la fonction F (éq. (17) avec les estimations courantes $\hat{\mathbf{t}}$ et $\hat{\omega}$) pour l'élément correspondant \mathbf{g}_k de la liste G ;
3. On ordonne le vecteur \mathbf{s} par éléments croissants, et on calcule les statistiques suivantes:
$$\sigma_s = \frac{1}{1,654} (\mathbf{s}_{0,72n_v} - \mathbf{s}_{0,28n_v}), \quad (20)$$

$$\bar{s} = \mathbf{s}_{0,5n_v}, \quad (21)$$
où \mathbf{s}_n est le $n^{\text{ème}}$ élément du vecteur \mathbf{s} ordonné;
4. On marque dans la liste G tous les appariements $\mathbf{g}_k = (i, j)^T$ tels que $F(\mathbf{d}_i, \mathbf{d}_j, \hat{\mathbf{t}}, \hat{\omega}) \geq \bar{s} + \eta\sigma_s$, où η est une constante.
5. Si $N_1 > N_2$ (rsp. $N_2 \geq N_1$) on élimine de la liste L_1 (L_2) toutes les droites dont les appariements ont été marqués dans l'étape précédente.

Ce mécanisme permet d'éliminer des appariements aberrants.

L'algorithme d'estimation de mouvement et d'appariement de droites peut se résumer ainsi :

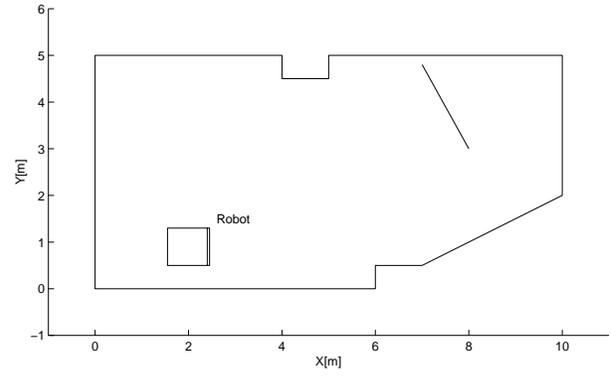


FIG. 4: Modèle de l'environnement simulé et robot dans sa position initiale.

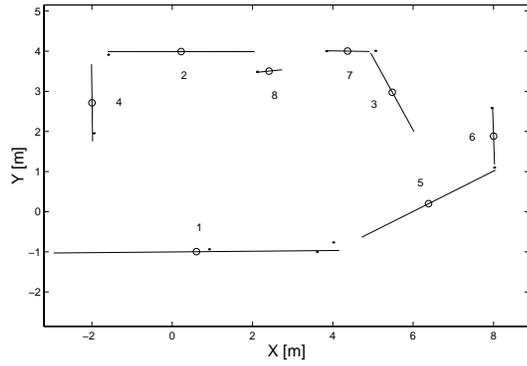
1. Variables d'entrée: listes L_1 et L_2 , et estimations initiales de \mathbf{t} et ω telles que $\hat{\mathbf{t}}_0 = \begin{pmatrix} 0 & 0 \end{pmatrix}^T$ et $\hat{\omega}_0$ a été fourni par un gyromètre;
2. Initialiser $k := 1$ et $m := 1, 5$;
3. Calculer u_{ij} et v_{ij} à partir des eqs. (14) et (15), avec D substitué par F (éq. (17)), et $\mu_{ij} = \min(u_{ij}, v_{ij})$;
4. Mettre à jour $\hat{\mathbf{t}}_k$ et $\hat{\omega}_k$ avec les équations (9) et (10);
5. Si $\|\hat{\mathbf{t}}_k - \hat{\mathbf{t}}_{k-1}\| \geq \epsilon_t$, faire $k := k + 1$ et retourner à 3;
6. Appliquer le mécanisme de sélection de droites. S'il existe des exclusions, retourner au pas 2.
7. Fin.

Un test arrête cet algorithme s'il y a un problème d'inversion matricielle dans le calcul du mouvement ou si le nombre d'itérations atteint un seuil prédéfini. Dans ces cas, l'algorithme ne peut pas donner d'estimation de mouvement.

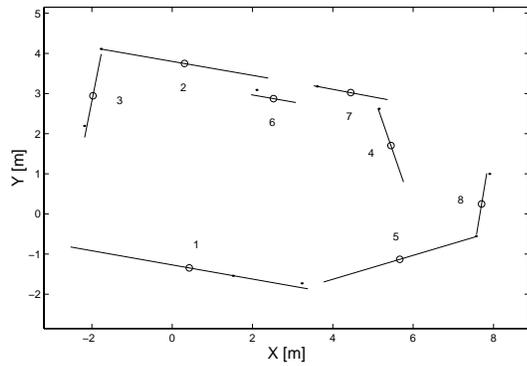
4 Simulation

Une simulation a été réalisée pour évaluer la performance de l'algorithme proposé. Nous avons utilisé l'environnement décrit par la Figure 4 dans lequel ont été générées 16 images télémétriques obtenues pour différentes positions du robot. Dans le simulateur, les mesures de distance du télémètre ont été contaminées par un bruit gaussien d'écart-type $\sigma_r = 5 \text{ cm}$. Le capteur balaye un angle de 270° avec un pas de $0,6^\circ$.

Les droites extraites des images 1 et 2 (qui correspondent aux deux premières positions) sont présentées dans les Figures 5(a) et 5(b). Chaque droite est identifiée par une étiquette. L'ordre de numérotation est attribué en fonction du nombre de points de la droite. Nous avons limité le nombre de droites de chaque image



(a) Droites extraites de l'image 1



(b) Droites extraites de l'image 2

FIG. 5: *Appariement de droites simulées.*

à 8, afin de conserver uniquement celles qui ont un nombre maximum de points.

L'algorithme d'estimation de mouvement a été exécuté avec les paramètres $\mu_{\min} = 0,5$ et $\eta = 10,0$ pour le mécanisme de sélection de droites. Pour simuler la mesure gyrométrique, ω a été initialisé par sa valeur réelle augmentée d'une variable aléatoire gaussienne d'écart-type 5° . Le nombre d'itérations de l'algorithme a varié en moyenne de 2 à 4. Pour les images 1 et 2 (Fig. 5), les valeurs de μ_{ij} sont données dans la Table 1. Nous pouvons vérifier que l'algorithme a trouvé les bons appariements sans éliminer de droite. Néanmoins, il y a eu des images où le mécanisme de sélection est intervenu. Les estimations de mouvement obtenues sont présentées dans la Figure 6. Dans cette figure, l'axe des abscisses porte les numéros des images où le mouvement a été estimé. Les écarts-types $\sigma_{\varepsilon_{i_x}}$, $\sigma_{\varepsilon_{i_y}}$ et $\sigma_{\varepsilon_{\omega}}$ obtenus sur les erreurs des estimations de t_x , t_y et ω sont $\sigma_{\varepsilon_{i_x}} = 2,39 \text{ cm}$, $\sigma_{\varepsilon_{i_y}} = 1,28 \text{ cm}$ et $\sigma_{\varepsilon_{\omega}} = 0,33^\circ$.

TAB. 1: *Simulation: μ_{ij} pour les images 1 et 2.*

$i \setminus j$	1	2	3	4	5	6	7	8
1	1,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0
2	0,0	1,0	0,0	0,0	0,0	0,0	0,0	0,0
3	0,0	0,0	0,0	1,0	0,0	0,0	0,0	0,0
4	0,0	0,0	1,0	0,0	0,0	0,0	0,0	0,0
5	0,0	0,0	0,0	0,0	1,0	0,0	0,0	0,0
6	0,0	0,0	0,0	0,0	0,0	0,0	0,0	1,0
7	0,0	0,0	0,0	0,0	0,0	0,0	1,0	0,0
8	0,0	0,0	0,0	0,0	0,0	1,0	0,0	0,0

Les résultats de cette simulation sont très satisfaisants. Néanmoins, l'environnement simulé était simple et il n'y avait aucun obstacle en mouvement. Dans la prochaine section, nous allons présenter les résultats de cette méthode pour deux expérimentations dans des environnements réels.

5 Validation expérimentale

La première expérience (expérience A) a été réalisée dans une salle vide dont un des murs est constitué par une baie vitrée. La deuxième expérience (expérience B) a été réalisée dans un hall où la présence d'obstacles en mouvement (i.e. personnes), et d'objets de forme irrégulière (plantes, ...) est fréquente. Les acquisitions d'images de distance ont été réalisées avec un robot mobile équipé d'un Ladar 2D *IBEO Laser-technik*. Les mesures ont été effectuées en arrêtant le robot. La précision dans la mesure de distance de ce capteur est de l'ordre $\pm 5 \text{ cm}$, avec un biais maximum de $2,3 \text{ cm}$ pour une distance d'environ 5 m . Son angle de balayage est 270° et son pas de mesure est $0,6^\circ$. Les mesures de positionnement réel du véhicule supposées exactes ont été effectuées de façon manuelle.

L'algorithme de segmentation utilisé a fourni uniquement des droites composées de plus de 10 points, afin d'avoir suffisamment de précision sur leurs paramètres polaires. Le nombre de droites de chaque liste a été ici encore limité à 8 au maximum. Le mécanisme de sélection emploie les mêmes paramètres que dans la simulation précédente.

5.1 Expérience A

Dans cette expérience, nous avons acquis 7 images successives. Les Figures 7(a) et 7(b) montrent les images 1 et 2, pour lesquelles l'algorithme a donné les plus grandes erreurs. Nous remarquons que les droites 3 et 6 de l'image 1 et les droites 4, 5 et 6 de l'image 2 ont été produites par un même obstacle: une grande baie vitrée. Dans ce cas, la précision du télémètre change en fonction de l'angle d'incidence. En conséquence, cet objet plan a été déformé entre les images 1 et 2,

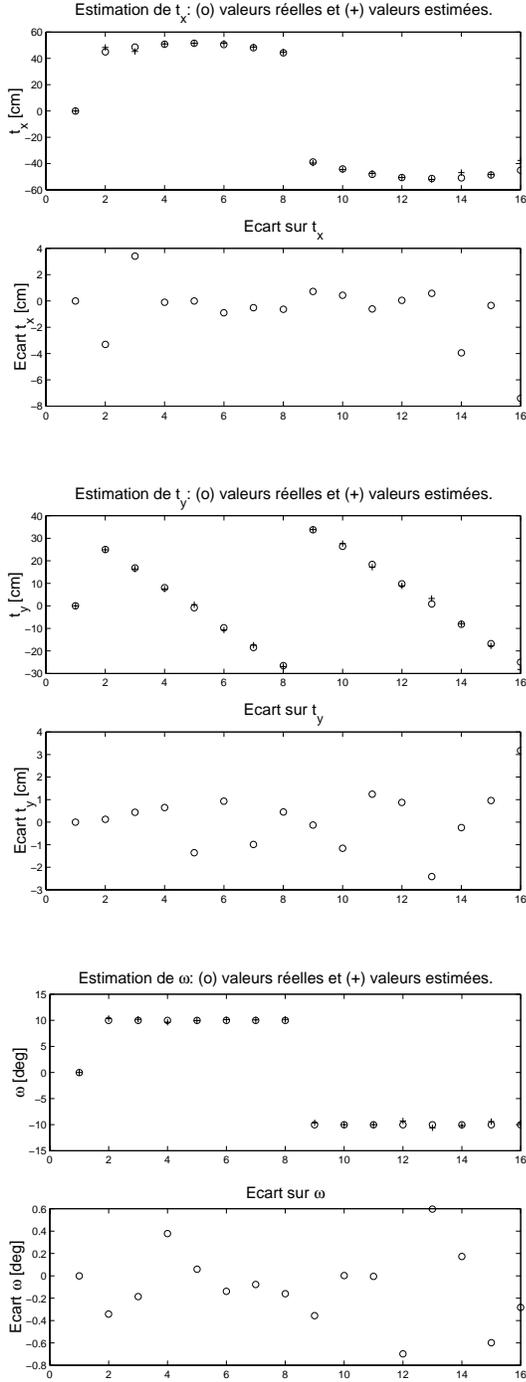


FIG. 6: *Simulation: estimation de mouvement (en abscisse: numéro des images de la séquence).*

TAB. 2: *Expérience A: μ_{ij} pour les images 1 et 2.*

$i \setminus j$	1	2	3	4	5	6	7
1	0,99	0,0	0,0	0,0	0,0	0,0	0,0
2	0,0	0,0	0,99	0,0	0,0	0,0	0,0
3	0,0	0,0	0,0	0,65	0,35	0,0	0,0
4	0,0	0,99	0,0	0,0	0,0	0,0	0,0
5	0,0	0,0	0,0	0,0	0,0	0,0	1,0
6	0,0	0,0	0,0	0,01	0,0	0,98	0,0

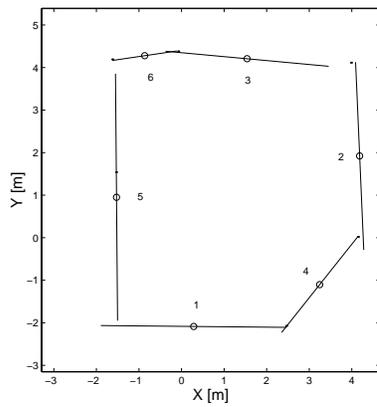
ainsi que dans les images suivantes. Ses mesures de correspondance évaluées par la fonction F étant bien distribuées, le mécanisme de sélection n'a pas pu rejeter les mauvais appariements. Cependant, la mesure μ_{ij} doit permettre d'atténuer l'effet de ces mauvais appariements sur l'estimation du mouvement. Cette mesure est donnée dans la table 2. Nous pouvons constater que la droite $i = 3$ a été appariée avec les droites $j = 4$ et $j = 5$, mais avec des mesures de correspondance différentes.

Les Figures 7(b) et 7(c) donnent un exemple de l'action du mécanisme de sélection sur les images 2 et 3. Nous pouvons voir que la droite $i = 7$ de l'image 2 et la droite $j = 6$ de l'image 3 n'ont pas de correspondante. Néanmoins, lors de la stabilisation de la première estimation de mouvement, ces droites ont été appariées car nous avons obtenu $\mu_{76} = 1,0$. Cette mesure a été attribuée à la correspondance la plus plausible. Le mécanisme de sélection a ensuite détecté cette faute et a éliminé la droite $i = 7$ de l'image 2. Ainsi, dans le nouveau cycle d'estimation de mouvement qui a suivi, la droite $j = 6$ n'a pas eu de correspondante. Toutes les bonnes correspondances ont été trouvées avec une erreur sur les paramètres polaires de 3 cm pour ρ et $0,3^\circ$ pour α .

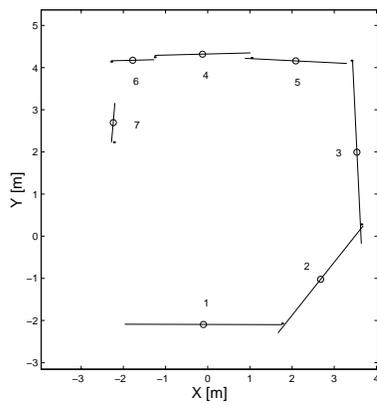
Les estimations \hat{t} et $\hat{\omega}$ sont données dans la Figure 8. Nous pouvons encore vérifier que la précision de l'estimation est satisfaisante. Les écarts-types sur les erreurs des estimations de t_x , t_y et ω pour l'expérience A sont: $\sigma_{\varepsilon_{t_x}} = 7,26 \text{ cm}$, $\sigma_{\varepsilon_{t_y}} = 4,87 \text{ cm}$ et $\sigma_{\varepsilon_{\omega}} = 0,74^\circ$.

5.2 Expérience B

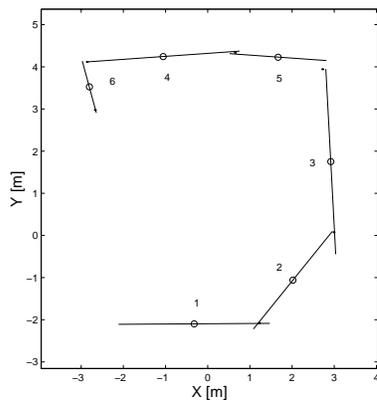
Dans cette expérience, nous avons obtenu 9 images successives. La Figure 9 montre un exemple d'appariement de droites réalisé par cet algorithme pour les images 3 et 4. Comme nous pouvons le vérifier, l'appariement n'est pas évident et il existe une quantité importante de petits obstacles. En effet, l'algorithme de segmentation a trouvé des droites très courtes, et certaines droites qui apparaissent dans l'image 3 ne sont pas vues dans l'image 4. Ici encore, le mécanisme de sélection a été exécuté une fois: la droite $j = 7$ de



(a) Droites extraites de l'image 1



(b) Droites extraites de l'image 2



(c) Droites extraites de l'image 3

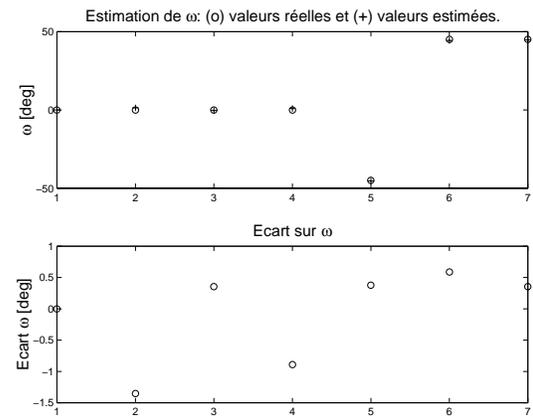
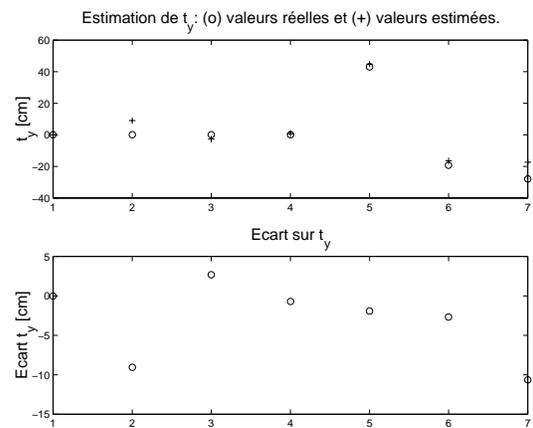
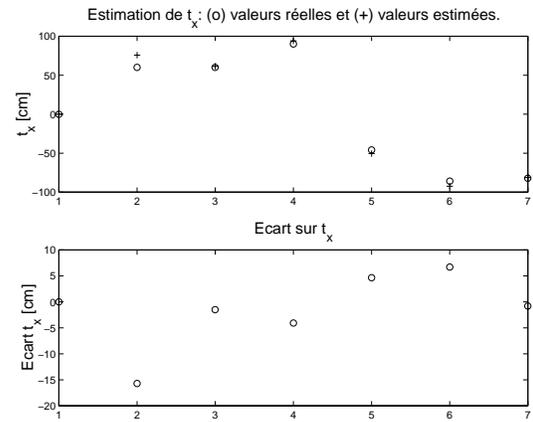


FIG. 8: *Expérience A*: estimation de t_x , t_y et ω (en abscisse: numéro des images de la séquence).

FIG. 7: *Expérience A*: droites des images 1, 2 et 3.

TAB. 3: *Expérience B: μ_{ij} pour les images 3 et 4.*

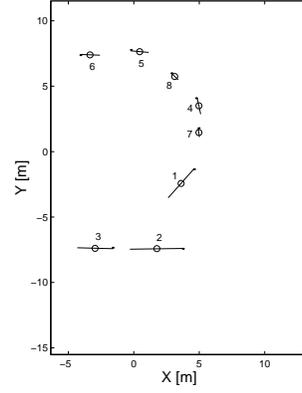
$i \setminus j$	1	2	3	4	5	6	7	8
1	1,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0
2	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,02
3	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,01
4	0,0	0,0	0,0	1,0	0,0	0,0	0,0	0,0
5	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0
6	0,0	0,0	0,0	0,0	0,0	1,0	0,0	0,0
7	0,0	0,0	0,19	0,0	0,01	0,0	0,0	0,0
8	0,0	0,99	0,0	0,0	0,0	0,0	0,0	0,0

l'image 4 a été rejetée. La valeur finale des coefficients μ_{ij} est donnée dans la table 3.

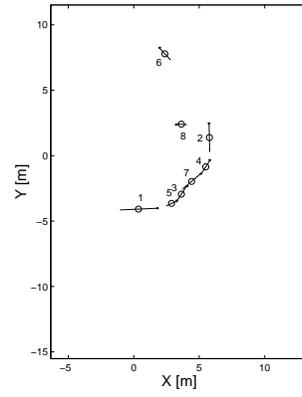
Les résultats des estimations $\hat{\mathbf{t}}$ et $\hat{\omega}$ sont montrés dans la Figure 10. Comme l'algorithme n'a pas pu estimer le mouvement entre les images 8 et 9, ces résultats ne sont affichés que pour les positions 1 à 8. L'échec d'appariement entre les deux dernières images peut s'expliquer par une grande amplitude de rotation (90°) qui se traduit par un petit nombre de droites en correspondance. Néanmoins, nous pouvons vérifier que la précision d'estimation de mouvement est satisfaisante, même avec de fortes variations d'orientation (par exemple, 65° entre les images 6 et 7). Les écarts-types sur les erreurs des estimations de t_x , t_y et ω ont été de: $\sigma_{\varepsilon_{t_x}} = 7,30 \text{ cm}$, $\sigma_{\varepsilon_{t_y}} = 7,73 \text{ cm}$ et $\sigma_{\varepsilon_{\omega}} = 1,03^\circ$.

6 Conclusions

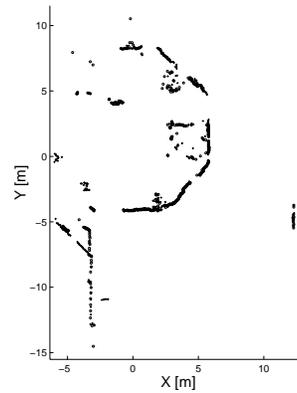
Nous avons présenté une méthode pour réaliser l'estimation du mouvement relatif d'un véhicule à partir de la mise en correspondance de droites extraites d'images télémétriques 2D. Cette méthode ne traite pas la mise en correspondance et l'estimation de mouvement comme des processus séparés. La mise en correspondance est réalisée de façon implicite dans une procédure d'optimisation et d'estimation de mouvement. Pour être plus insensible aux mauvais appariements, l'algorithme utilise un mécanisme de sélection basé sur les statistiques robustes. Des résultats de simulation et d'expérimentation ont montré la robustesse de cette méthode aux mauvais appariements et aussi son comportement dans certains cas particuliers. Nous cherchons maintenant à étendre cette approche pour mettre en correspondance d'autres types d'indices.



(a) Droites extraites de l'image 3



(b) Droites extraites de l'image 4



(c) Images 3 et 4 superposées

FIG. 9: *Expérience B: appariement des images 3 et 4.*

Références

- [1] J. Borenstein, H. R. Everett et L. Feng, *Navigating Mobile Robots, Systems and Techniques*, A. K. Peters Ltd, 1996.
- [2] I. J. Cox, *Blanche: An Experiment in Guidance and Navigation of an Autonomous Mobile Robot*. IEEE Transactions on Robotics and Automation, vol. 7, n. 3, 1991, pp. 193-204.
- [3] L. Le Bris, *Localisation d'un robot mobile par fusion de données proprioceptives et extéroceptives*. Thèse de Doctorat, Université Montpellier II, juillet 1995.
- [4] Tobias Einsele, *Real-Time Self-Localization in Unknown Indoor Environments Using a Panorama Laser Range Finder*, IROS 1997, pp. 697-702.
- [5] H. Bulata, *Modélisation d'un environnement structuré et localisation sur amers pour la navigation d'un robot mobile autonome*. Thèse de Doctorat, Université Paul Sabatier, Toulouse, mai 1996.
- [6] John J. Leonard, Hugh F. Durrant-Whyte et Ingemar J. Cox, *Dynamic Map Building for an Autonomous Mobile Robot*, The International Journal of Robotics Research, vol. 11, n. 4, august 1992, pp. 286-298.
- [7] J. Gonzalez, R. Gutierrez, *Mobile Robot Motion Estimation from Range Scan Sequence*, IEEE International Conference on Robotics and Automation, Albuquerque, New Mexico, USA, april 1997.
- [8] Feng Lu et Evangelos Miliotis, *Robot Pose Estimation in Unknown Environments by Matching 2D Range Scans*, Journal of Intelligent and Robotic Systems, vol. 18, n. 3, 1997, pp. 249-275.
- [9] J. Borenstein et L. Feng, *Measurement and Correction of Systematic Odometry Errors in Mobile Robots*, IEEE Transactions on Robotics and Automation, vol. 12, n. 6, december 1996, pp. 869-880.
- [10] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, 1981.
- [11] Didier Dubois et Henri Prade, *Possibility Theory: An approach to Computerized Processing of Uncertainty*, Plenum Press, 1986.
- [12] Jean-Pierre Lecoutre et Philippe Tassi, *Statistique non paramétrique et robustesse*, Ed. Economica, 1987.

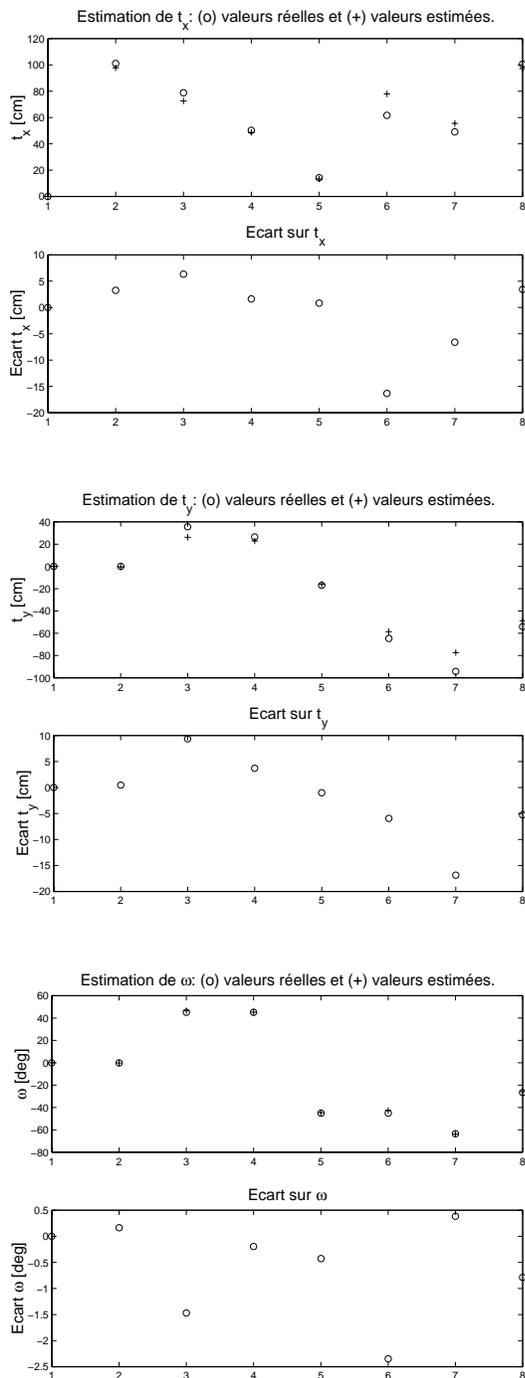


FIG. 10: Expérience B: estimation de t_x , t_y et ω (en abscisse: numéro des images de la séquence).