The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

Flexible Extrinsic Calibration of Non-Overlapping Cameras Using a Planar Mirror: Application to Vision-Based Robotics

Pierre Lébraly[1,3], Clément Deymier[1,3], Omar Ait-Aider[1,3], Eric Royer[2,3] and Michel Dhome[1,3]

Abstract— **Multi-camera systems are used in many domains such as vision-based robotics or video surveillance. An accurate extrinsic calibration is usually required. In most of cases, this task is done by matching features through different views of the same scene. However, if the cameras' fields of view do not overlap, such a matching procedure is not feasible anymore.**

Despite this constraint, this article deals with a simple and flexible extrinsic calibration method. The main contribution is the use of an unknown geometry scene and a planar mirror to create an overlap between views of the different cameras. Furthermore, the impact of the mirror refraction is also studied. The aim is the calibration of two non-overlapping cameras embedded on a vehicle, for visual navigation purpose in urban environment. The proposed approaches have been validated with both synthetic and real data in a metrological experimental framework.

I. Introduction

Several approaches have been proposed in order to calibrate a set of cameras with non-overlapping field of view. Depending on application, a multi-camera system can be static or embedded on a vehicle. In this paper, both cases are considered. In the first case, it's possible to use an additional camera moving around the static cameras to calibrate [1]. Calibration is obtained thanks to a 3D reconstruction up to a scale factor. In contrast, Lamprecht [2] calibrates a moving multi-camera system. A fixed calibration object is tracked and a prior knowledge about the system speed is required. Another approach, first proposed in [3] and improved in [4], consists in using the rigidity constraint between the coupled cameras. First, each camera trajectory is computed. Second, the relative poses are deduced since they do not change over time. Then, this rigidity constraint can be used during a localization process of the multi-camera system [5], [6].

An alternative is to use a mirror to create an overlap between views. Martins [7] puts an object between the camera and planar mirrors to calibrate the system and reconstruct the object. Another way to use planar mirrors is shown by Gluckman [8] in order to capture stereo images just from one camera. Sturm and Bonfort [9] use planar mirrors to estimate the pose of a calibration object with respect to one camera. Similar works have been presented by Hesch [10], [11] to analytically solve this problem. Kumar [12] develops an extrinsic calibration method for a multi-camera system, where a planar mirror and a known geometry calibration pattern are used.

Authors are with (1) Clermont Université, Université Blaise Pascal, LASMEA, BP 10448, F-63000 CLERMONT-FERRAND (2) Clermont Université, , LASMEA, BP 10448, F-63000 CLERMONT-FERRAND (3) CNRS, UMR 6602, LASMEA, F-63177 AUBIERE Firstname.Lastname@lasmea.univ-bpclermont.fr

In this paper, we propose a calibration strategy which does not require knowing the scene geometry. Our approach uses visual markers stuck on the mirror surface in order to initialize nonlinear equations. After a mathematical formalization of the problem II, we introduce two distinctive methods to estimate the relative pose of rigidly linked cameras. Their fields of view can be totally non-overlapping. How to apply this method in the context of mobile robots is presented. In Section IV, the impact of the mirror's refraction is studied. Finally (Section V), the results validate our approach with both synthetic and real data. Appendix VII is dedicated to an automatic feature detection and an identification method.

II. Mathematical formulation

The goal is to calculate the rigid transformation between different cameras from 2D/2D matchings.

A. System overview

We consider a scene S_i of 3D points (Fig. 1), and at least two rigidly linked cameras C^{1r} and C^{2r}, with known intrinsic parameters.The camera C^{2r} can see directly the scene. The camera C^{1r} watches several views of this scene thanks to the mobile planar mirror. Let ${}^{1r}T_{2r}$ be the homogeneous transformation from C^{1r} to C^{2r} coordinate systems. In the same way, ${}^{1r}T_{\Pi_j}$ represents the homogeneous transformation from C^{1r} and Π_j coordinate system, where Π_j denotes the planar mirror observed in the j^{th} image from C^{1r}.

We call a *virtual camera*, the camera obtained by a mirror plane symmetry. C^{1v_j} is the j^{th} virtual camera obtained from C^{1r} relative to Π_j. It is assumed that one image of the scene is acquired by each C^{1v_j} and one other image by C^{2r}. $s_i^{1v_j}$, s_i^{2r} are respectively the projected points of the scene S_i, in the image plane of C^{1v_j} and C^{2r}. Fig. 1 illustrates the whole system. Due to symmetry, real cameras are represented by a right-handed basis whereas virtual cameras are represented by a left-handed basis.

A set of 9 markers M_k — blue circles in Fig. 1 — are stuck on the mirror's surface to enable each mirror pose Π_j computation (see III-A). The automatic and subpixel markers detection is detailed in appendix VII. In the sequel, $m_{k_j}^{1r}$ denotes the j^{th} projection of M_k points in the C^{1r} image plane. Such markers are used for several reasons:

- Geometric constraints are added.
- The mirror pose estimate enables to deduce the link between a real camera and its virtual camera. Moreover, opposite to [12] where degenerate configurations are present (for example: the mirror cannot be orthogonal to the camera's optical axis), our methods don't forbid any pose for the mirror. So the user's task is more flexible.

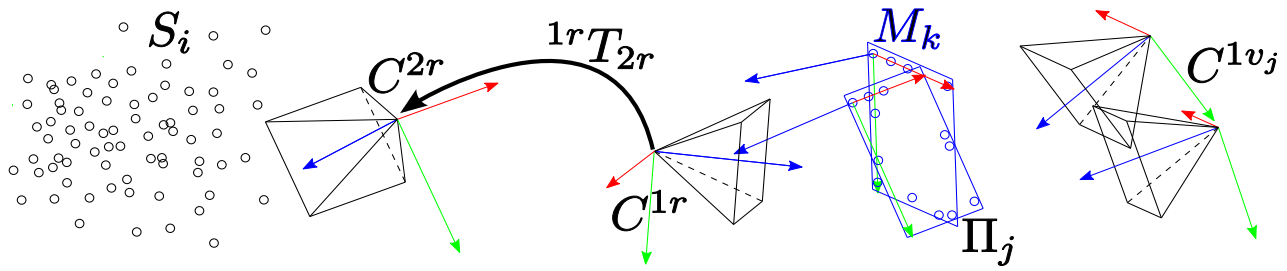

Fig. 1. Schema of the mirror and the multi-camera system

B. Intrinsic parameters of a virtual camera

The intrinsic calibration matrix concept can be extended for the virtual cameras. Let's consider the intrinsic calibration matrix K_r of a pinhole real camera (2). With $f_x = f/d_x$ and $f_y = f/d_y$, where f is the focal length, d_x and d_y are the pixel sizes of the CCD array, and (u_0, v_0) are the principal point pixel coordinates.

The virtual camera's image plane is obtained by a reflection with respect to an axis of our choice. As we want to get a simple expression for the intrinsic calibration matrix, we choose the vertical axis through the principal point[1]. (x, y) are the real camera image coordinate system (the origin is the principal point). The pixel coordinates of the real and virtual camera are respectively (u_r, v_r) and (u_v, v_v), expressed by:

$$\left\{ \begin{array}{l} u_r = \frac{x}{dx} + u_0 \\ v_r = \frac{y}{dy} + v_0 \end{array} \right. \Leftrightarrow \left\{ \begin{array}{l} u_v = 2u_0 - u_r = -\frac{x}{dx} + u_0 \\ v_v = \frac{y}{dy} + v_0 \end{array} \right. \quad (1)$$

The virtual intrinsic calibration matrix K_v then simplifies to:

$$K_r = \begin{pmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{pmatrix} \Rightarrow K_v = \begin{pmatrix} -f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{pmatrix} \quad (2)$$

III. Non-overlapping camera calibration

First, we propose two ways to calibrate the system: one use a known calibration pattern, the other doesn't require such a pattern. Second, the calibration procedure for a mobile robot is discussed.

A. Mirror pose computation

Our calibration method starts by computing an estimation of the mirror pose Π_j relative to C^{1r}. Starting from an approximate knowledge about the markers M_k (measured with a ruler), a homography is computed using a 2D/3D registration technique [14]. Therefore, a planar object pose is first obtained using a decomposition of the homography matrix H relating the object and the camera image plane (see (3) and [15]). The homography is estimated linearly, and then refined using a nonlinear optimization (refer to [16] for more details). Once H is obtained, the pose parameters (R, T) are computed as follows :

$$\left\{ \begin{array}{l} \lambda = \|K^{-1}\mathbf{h}_1\|^{-1} \\ \quad = \|K^{-1}\mathbf{h}_2\|^{-1} \\ T = \lambda K^{-1}\mathbf{h}_3 \end{array} \right. \text{and} \left\{ \begin{array}{l} \mathbf{r}_1 = \lambda K^{-1}\mathbf{h}_1 \\ \mathbf{r}_2 = \lambda K^{-1}\mathbf{h}_2 \\ \mathbf{r}_3 = \mathbf{r}_1 \wedge \mathbf{r}_2 \end{array} \right. \quad (3)$$

[1]This problem is mentioned in [13].

Where $R = [\mathbf{r}_1\, \mathbf{r}_2\, \mathbf{r}_3]$, $H = [\mathbf{h}_1\, \mathbf{h}_2\, \mathbf{h}_3]$ and K is the intrinsic matrix of the current camera. A last step consists in refining both poses and markers geometry in a bundle adjustment process [17].

B. Calibration with a known pattern

We suppose that the scene geometry (*i.e.* the points S_i) is known. The calibration process is detailed in Algorithm 1.

Algorithm 1: Extrinsic calibration with a known calibration pattern

Inputs : s_i^{2r}, $s_i^{1v_j}$, $m_{k_j}^{1r}$, M_k and S_i
Outputs: ${}^{1r}T_{2r}$, ${}^{1r}T_{\Pi_j}$, and M_k

Step 1: Initialization

For each *image* j **:**
  Calculate the mirror Π_j pose relative to C^{1r}.
Refine the markers M_k geometry and the mirror Π_j poses by bundle adjustment.
For each *image* j **:**
  Deduce the virtual camera C^{1v_j} pose by symmetry with respect to Π_j.
  Calculate the pattern S_i pose relative to C^{1v_j}.
Calculate the transformation ${}^{2r}T_S$ between camera C^{2r} and the pattern S_i.
Deduce the initial transformation ${}^{1r}T_{2r}$ by averaging all the several estimations [18], [19], [20].

Step 2: Global nonlinear optimization

$$\underset{({}^{1r}T_{2r}, {}^{1r}T_{\Pi_j}, {}^{2r}T_S, M_k)}{\operatorname{argmin}} \Big(\sum_k \sum_j \| m_{k_j}^{1r} - \underset{C^{1r}}{\operatorname{proj}}(M_k) \|^2 + \sum_i \left[\| s_i^{2r} - \underset{C^{2r}}{\operatorname{proj}}(S_i) \|^2 + \sum_j \| s_i^{1v_j} - \underset{C^{1v_j}}{\operatorname{proj}}(S_i) \|^2 \right] \Big) \quad (4)$$

Where $\underset{C}{\operatorname{proj}}(M)$ is the projection function of the point M in the camera C. With $p([XYZ]^T) = \frac{1}{Z}(X\ Y)^T$, we get:

$$\begin{array}{l} \underset{C^{1r}}{\operatorname{proj}}(M_k) = p(K_{1r}\, {}^{1r}T_{\Pi_j} M_k) \\ \underset{C^{2r}}{\operatorname{proj}}(S_i) = p(K_{2r}\, {}^{2r}T_S S_i) \\ \underset{C^{1v_j}}{\operatorname{proj}}(S_i) = p(K_{1v_j}\, {}^{1v_j}T_{1r}\, {}^{1r}T_{2r}\, {}^{2r}T_S S_i) \end{array} \quad (5)$$

where the symmetry transformation between C^{1v_j} and C^{1r} is ${}^{1v_j}T_{1r} = \begin{pmatrix} I_3 - 2n_j n_j^T & 2d_j n_j \\ 0\ \ 0\ \ 0 & 1 \end{pmatrix}$. Here, n_j is the mirror

normal vector expressed in C^{1r} coordinate system, and d_j is the distance between C^{1r} and Π_j.

The calibration pattern can be either planar or not. During the initialization step, a planar object pose is obtained by using (3). If the calibration pattern isn't planar, 3D pose estimation is first initialized by the method described by Dementhon [14], then by nonlinear optimization[2]. In the second step, all system parameters (cameras and mirror poses) are optimized by minimizing every reprojection error using the Levemberg Marquardt algorithm.

From a practical point of view, the user has to move the mirror until a suffisant number of pattern points are seen by both C^{1v_j} and C^{2r}.

C. Calibration without a known pattern

We now propose a calibration process (algorithm 2) which uses an unknown geometry scene S_i : no calibration pattern is required. Thus, the field of view is not restricted to a pattern anymore: the whole field of view covered by the mirror can be used. From user point of view, this algorithm is easier and more flexible than the first one, because the entire scene hasn't to be observable through the mirror in all images (scene's occlusions are handled). It relies on an accurate and automatic feature detector which is detailed in Appendix VII.

Algorithm 2: Extrinsic calibration without any known calibration pattern

Inputs : s_i^{2r}, $s_i^{1v_j}$, $m_{k_j}^{1r}$ and M_k
Outputs: ${}^{1r}T_{2r}$,${}^{1r}T_{\Pi_j}$, M_k and S_i

Step 1: Initialization

For each *image* j **:**
  Calculate the mirror Π_j pose relative to C^{1r}.

Refine the markers M_k geometry and the mirror Π_j poses by bundle adjustment.

For each *image* j **:**
  Deduce the virtual camera C^{1v_j} pose by symmetry with respect to Π_j.

Reconstruct the scene points S_i relative to C^{1r}, using C^{1v_j} and $s_i^{1v_j}$: a projective factorization [16, p 444] and a bundle adjustment are used.
Calculate the camera C^{2r} pose ${}^{2r}T_S^{-1}$ relative to the scene S_i, express in C^{1r} coordinate system. It's the initial guess of ${}^{1r}T_{2r} = {}^{2r}T_S^{-1}$.

Step 2: Global nonlinear optimization

$$\underset{\left({}^{1r}T_{2r},{}^{1r}T_{\Pi_j},S_i,M_k\right)}{\operatorname{argmin}} \Big(\sum_k \sum_j \| m_{k_j}^{1r} - \underset{C^{1r}}{\operatorname{proj}}(M_k) \|^2 + \sum_i \left[\| s_i^{2r} - \underset{C^{2r}}{\operatorname{proj}}(S_i) \|^2 + \sum_j \| s_i^{1v_j} - \underset{C^{1v_j}}{\operatorname{proj}}(S_i) \|^2 \right] \Big) \quad (6)$$

In contrast to algorithm 1, every scene point is reconstructed during the last optimization step. The scale factor of the whole system (common to every calibration process) is obtained using at least one accurate known distance between two scene points. In the proposed schema, C^{1r} and C^{2r} don't have a symmetric function. Increasing the number of mirror poses Π_j yields to a larger amount of data extracted from C^{1r}. In contrast, just one image is extracted from C^{2r}. The proposed methods are so described for clearness purpose. However, they can easily be converted to symmetric methods by adding — during the optimization process — new images of the mirror in front of C^{2r}.

Moreover, an outlier rejection process is useless since feature's false detections are rejected, during the subpixel refinement or the label identification (see appendix VII).

[2]Levenberg Marquardt algorithm minimizes reprojection error. Every rotation is represented by Euler angles.

Fig. 2. Two non-overlapping cameras embedded on a vehicle

D. Calibration for a mobile robot

Recently, some navigation methods have been proposed for mobile robots using a single camera and natural landmarks. However, those approaches are very sensitive to outdoor light conditions. To alleviate this problem, a solution consists in mounting two cameras on the vehicle: one at the front, another at the back (see Fig. 2).

The main advantages are the vehicle symmetry, and the localization robustness against outdoor illumination problems, e.g., overexposure. If the sun is in front of one camera, the other gives some useful information. Then, the localization process requires calibrating the multi-camera system. Contrary to previous cases, the multi-camera system is not static which slightly changes Algorithm 2. The proposed calibration scheme, which takes advantage of the robot motion, is described in the Procedure.

Procedure: Calibration setup for a mobile robot

Move the vehicle and acquire N images of the scene.
Stop the vehicle. A camera should observe the scene. In our case, it's the front camera, referred as C^{2r}.
Move the mirror N times in front of the other camera, referred as C^{1r}.
Apply the calibration without a known pattern (Algorithm 2).

Contrarily to Algorithm 2: after the reconstruction stage, a pose is estimated for all the N views of the scene. Then, they are taken into account during the nonlinear optimization.

This additional set of scene views enables to:

- get a better 3D scene reconstruction.
- have the same number of images for each camera.

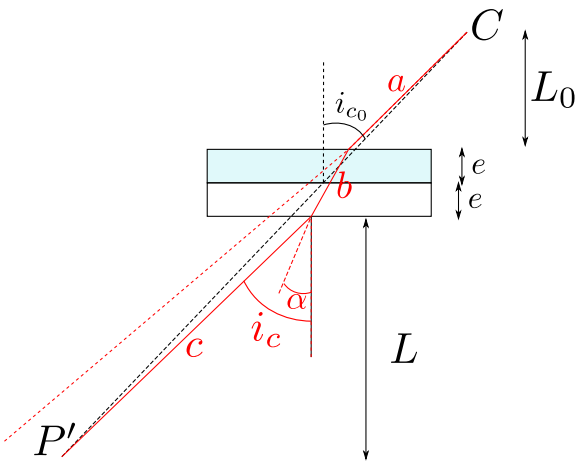

Fig. 3. Refraction caused by a planar mirror watched by a central camera

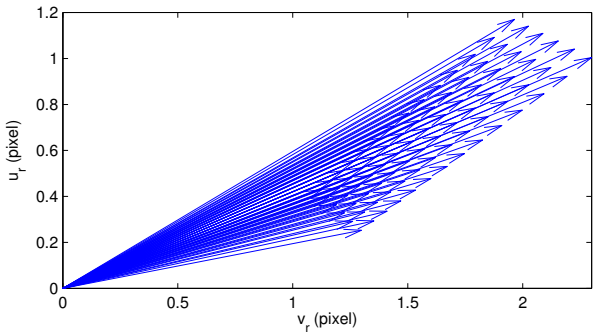

Fig. 4. Image plane error due to the refraction. A planar checkerboard pattern is observed at a distance of about 1.2 m with a mean angle of incidence of 29°

IV. Refraction influence

This section is dedicated to the study of the impact of the refractive effect — caused by the glass layer over a second surface mirror — on the pose estimation of a mirrored object. To the best of our knowledge, this problem has never been studied. Indeed, if the refraction is ignored, then the calibration algorithms can converge to biased results. Let's now see how to model the refraction and take it into account in the calibration process.

A. Perspective camera and refraction model

Instead of studying the view of an object through a mirror, we consider an equivalent problem: the view of a virtual object through a glass layer. Therefore, this equivalent glass layer is twice thicker as the mirror glass layer. According to Snell-Descartes' first law, we can restrict the problem to the plane defined with the optical center C of the camera, the observed point P' and the mirror normal. Fig. 3 illustrates the refraction phenomena. In the mirror case, the virtual point P' is the symmetry of the real point P with respect to the reflective surface. e refers to the mirror thickness, i_c is the angle of incidence when refraction is considered and i_{c_0} is the angle defined by the line (CP') and the surface normal of the mirror.

The goal is to get the minimum optical path length between P' and C (Fermat's principle). In our case, the optical path length can be expressed as:

$$\mathcal{C} = n_{air}(a + c) + n_{glass}b \tag{7}$$

with $n_{air} = 1$ and $n_{glass} = 1.5$. Geometrically, we obtain the equation (8):

$$\begin{aligned} &2e\tan\left(\arcsin\left(\frac{n_{air}}{n_{glass}}\sin(i_c)\right)\right) \\ =&(L_0 + L + 2e)\tan(i_{c_0}) - (L_0 + L)\tan(i_c) \end{aligned} \tag{8}$$

i_{c_0} is first computed, by ignoring refraction. Then, a Gauss-Newton method is used to solve for i_c in the equation (8).

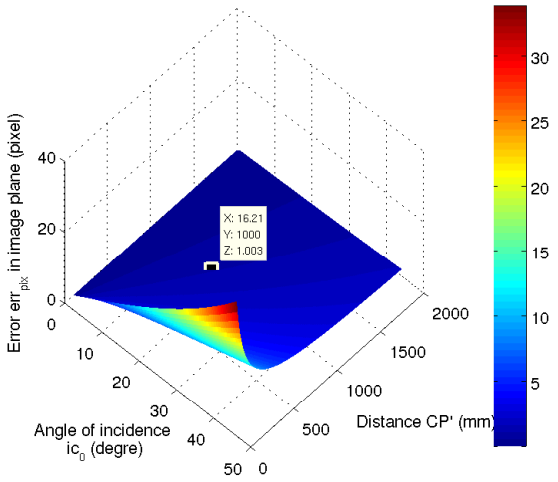

Fig. 5. Pixel error if the refraction were ignored

B. Impact of the refraction on the image plane

If the refraction was not considered, we would try to observe the point P' on the red dotted line emanated from camera center (see Fig. 3), whereas P' is in fact on the black dotted line. It involves the angular error $\delta i_c = i_c - i_{c_0}$. For a set of close points, it would lead to biased projections in the image plane of the camera (see Fig. 4 which shows the pixel error for a planar pattern. The grid size is about 2 cm). In this section, we consider real world values for the mirror thickness $e = 2.80$ mm, an 8 mm focal length for the camera and the pixel width: 4.4 μm.

Thus, we're able to quantify the impact of the angular error δi_c with respect to angles of incidence i_{c_0} and distances CP' between the camera and the point. However, this quantitative effect is meaningful on the camera's image plane. Fig. 5 shows the theoretical pixel error err_{pix} if the refraction were ignored. For each i_{c_0} and CP', we computed $L + L_0 = CP'cos(i_{c_0}) - 2e$ and then solve (8) to get δi_c. The pixel error is lastly obtained with $err_{pix} = f_y sin(\delta i_c)$, by assuming that P' is on the optical axis of the camera.

Some intuitive results are found back: refraction is all the more significant that the angle of incidence is high and that the observed point is near. So, do we really need to consider refraction? The answer isn't boolean, but depends on the desired accuracy, the focal length, the mirror thickness, the angle of incidence and the distance of the observed point.

For instance, if the observed point is about $1\ m$ distant, with an angle of incidence of $16°$, then image error is 1 pixel (see Fig. 5). As a consequence, refraction can easily induce an error around about one pixel. However, features are detected with subpixel accuracy: round about a tenth or a hundredth of pixel. In this case, refraction should not be ignored.

C. Pose estimation and refraction

The pose of an object viewed through a refractive medium can be calculated in two steps:

- First, the object pose is estimated while refraction is ignored (with for instance Dementhon's algorithm [14]).
- Second, the reprojection error is minimized by taking into account the refraction phenomena and the optimal optical path length computation, cf IV-A.

For the calibration algorithms, the refractive effect is just included in the projection function $\underset{C^{1v_j}}{\text{proj}}()$ of the virtual cameras.

V. Results

In this Section, the proposed methods are validated with both synthetic and real data.

A. Results with synthetic data

The experimental protocol is the following. First of all, scene points S_i are created. We synthesize all camera poses, and several random mirror poses at distance about $150\ mm$ from the camera C^{1r}. The mirror is assumed to be first surface. This ground truth is used to calculate the image points s_i^{2r}, $s_i^{1v_j}$ and $m_{k_j}^{1r}$. Then, they are all subject to an additive white Gaussian noise with standard deviation σ. Finally, the relative pose between the two cameras is estimated with noisy data. Six images of the mobile mirror are acquired. Moreover, all scene points are supposed to be in the virtual camera's field of view. Intrinsic parameters are also supposed to be known.

During this experiment, the transformation ${}^{1r}T_{2r}$ corresponds to the translation $(0.3m\ -0.5m\ -1m)^T$ and the rotation from yaw, pitch and roll angles (15°, 180°, -20°). For every σ value, 20 measures have been performed. The calibration accuracy is given by both the norm $\|dT\|$ of the translational error vector and the angular error dR whose expression is given by:

$$dR = d(\hat{R}, R^*) = \arccos\left(\frac{trace(\hat{R}^T R^*) - 1}{2}\right) \quad (9)$$

where R^* is the ground truth rotation and $\hat{R}$ is the estimated rotation.

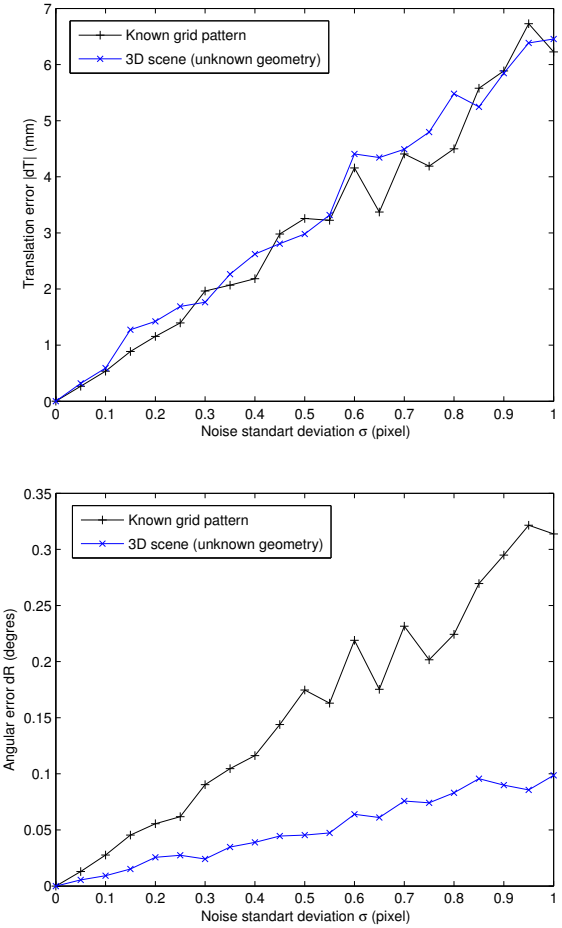

Fig. 6. Results with synthetic data: mean value of the pose estimation error with respect to noise level

Fig. 6 shows the performances of the Algorithm 1, using a known planar grid pattern with 84 points (about $12\ cm$ by $30\ cm$) at distance $1\ m$ from C^{2r}. It can be compared with the performances of Algorithm 2, for which an 84 points unknown 3D scene is randomly created at about one meter from C^{2r}.

For the same point number, both methods are equivalent according to translation estimation. Whereas in presence of noise, rotation estimation is three times more accurate with the Algorithm 2 as compared to Algorithm 1. This is because the points of a known grid pattern are restricted to a quite tiny area and our experiments show that the more scattered the features are, the more accurate is the extrinsic calibration. Nevertheless, the feature detection noise could counterbalance the accuracy due to scattering. As a result, the accuracy depends on a compromise between the number, the scattering, and the noise of features.

B. Results with real data

We use an $8\ mm$ focal length for the camera, with a pixel width of $4.4\ \mu m$. During our first experiments, we used a low cost mirror of unknown quality. The results weren't satisfying (the standard deviation of reprojection error was $0.6\ pix$) because of the inaccuracy of the planarity,

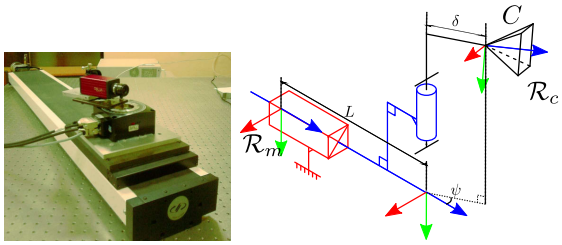

Fig. 7. Metrological system and its mechanical linkage

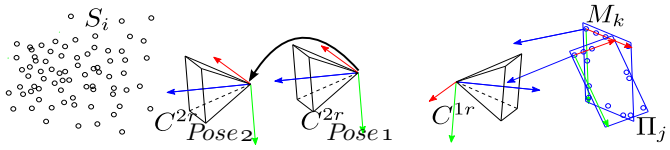

Fig. 8. Relative poses experiment - pure translations

and too much bending deformation (due to low thickness $e = 2.8\ mm$). So we use a $6mm$ thick mirror with a surface accuracy of 4λ to 6λ per inch, where λ is the wavelength. Such mirrors are front surface.

In order to have an accurate ground truth — unavailable if our system were directly mounted on a vehicle — a camera is mounted onto a metrological motion system. Its mechanical linkage is shown in Fig. 7. It is assumed that the axis of rotation and the axis of translation are perfectly orthogonal. A motorized rotation stage, with $0.01°$ accuracy, is mounted on a linear stage. A micrometric translation is reachable. L refers to measured translation vector norm, and ψ is the measured angle of rotation. A camera is rigidly mounted on the rotor. The camera optical center is set along the axis of rotation by manually minimizing the parallax.

Thanks to the metrological system, a multi-camera system is recreated with high accuracy. Each camera is obtained at a fixed rotation/translation pair (ψ, L).

The criteria for the rotation accuracy between cameras C^{1r} and C^{2r}, is defined as follows:

$$dR = |d(I_3, \hat{R}) - (\psi_1 - \psi_2)| \tag{10}$$

with the function $d()$ defined by (9). Some unknown parameters are added by the experiment system: the parallax error δ and the rotation matrix between the rotor and the camera coordinate system $\mathcal{R}_c$. We don't make any assumption on those parameters, hence only valid and accurate metrological measures are used. Therefore, the calibration is done thanks to the Procedure of III-D with pure translations (see Fig. 8). We acquire several pictures of the mirror moving in front of the fixed camera C^{1r}, and pure translations are applied to camera C^{2r}. A picture is acquired every $5\ cm$, for L varying from 0 through $100\ cm$. This set of poses of camera C^{2r} represents the ground truth.

The line Δ fits their aligned optical centers. The algorithm accuracy is measured with the following criteria:

- The distance d between the line Δ and optical centers,
- The angular error ϕ between each camera C^{2r} and the mean camera [18], [19], [20],

TABLE I

EXPERIMENT RESULTS OF THE PROCEDURE OF III-D

	Mean	Standard deviation
d (mm)	0.04	0.03
ϕ (°)	0.009	0.005
dR (°)	0.02	0.002
τ (mm)	0.02	0.02
$\|dT\| + 2\delta$ (mm)	4.7	0.07

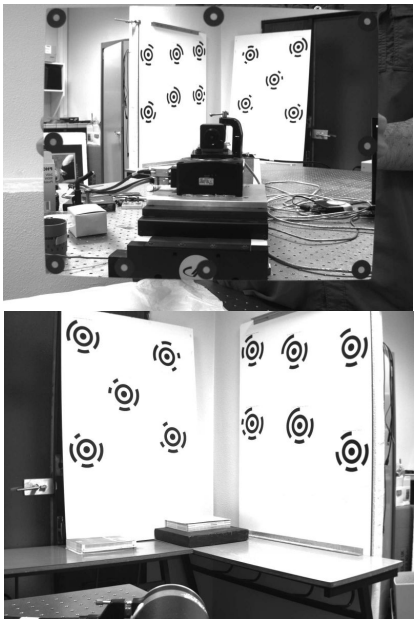

Fig. 9. Experiment with an unknown scene geometry, indirect view (top), and direct view (bottom). The image resolution is 1600x1200.

- The angular error dR between the pose of C^{1r} and C^{2r},
- The error τ of the translation vector norm between two successive poses.

Table I shows the results when a $180°$ rotation and a $1\ m$ translation are applied between C^{1r} and C^{2r} (for $N = 21$) (see Fig. 9).

As detailed in Table I, all the aligned C^{2r} poses are reconstructed very precisely (d is under $0.1\ mm$ and ϕ is about $0.01°$). The rotation between C^{1r} and C^{2r} is found back with an accuracy less than $0.03°$ (dR). The translation error τ is about $0.02\ mm$ added up to each successive poses; *i.e.* for a $1\ m$ translation it leads to an error of $0.3\ mm$. This error is due to the estimation of the scale factor. The error $\|dT\|$ of the norm of translation between C^{1r} and C^{2r} is biased, which is mostly due to the parallax error δ. After the minimization, the standard deviation of reprojection error is $0.14\ pix$. These results are satisfying for mobile robot applications.

We compared the previous results with Algorithm 2. The angular difference is less than $0.02°$. The norm of the translation difference is 2.6 mm. Indeed, additional poses of camera C^{2r} lead to a best depth estimation of the scene, and thus to an accurate translation estimation.

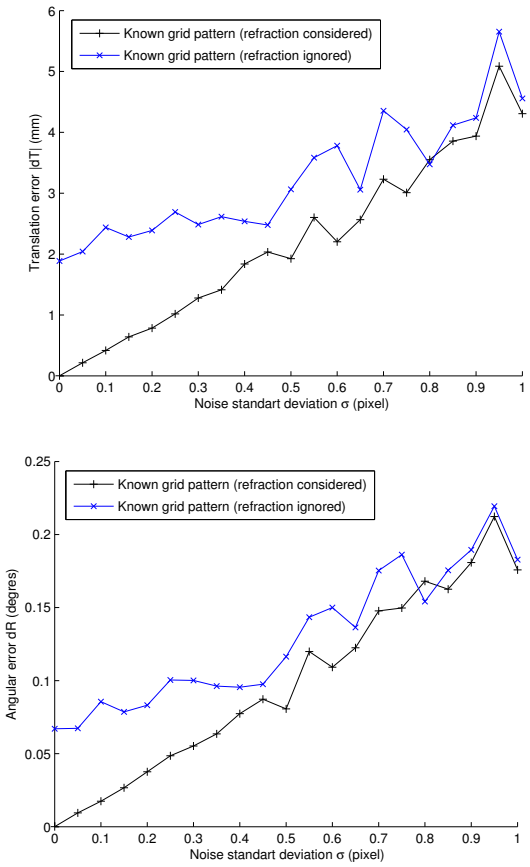

Fig. 10. Impact of the refraction on the accuracy of the Algorithm 1 (synthetic data)

C. Refraction and calibration algorithm

Let's consider a simulated ground truth created with the refractive effect, the experimental protocol is similar to the one described previously in V-A. Fig. 10 shows the effect of the refraction on Algorithm 1, called twice; with or without modeling the refraction. During these simulations, the transformation ${}^{1r}T_{2r}$ corresponds to the translation $(0m\ 0m\ -1m)^T$ and the rotation from yaw, pitch and roll angles are (5°, 40°, -20°). When the refraction is ignored for low noise level, Fig. 10 shows that estimation of translation and rotation can be biased. For mobile robot localization, this translation bias ($2\ mm$) is acceptable but the rotation bias (0.07°) is greater than the real achievable accuracy (see V-B).

VI. Conclusion

A flexible method for calibration of a static or embedded multi-camera system, with non-overlapping fields of view, has been proposed and validated with both synthetic and real data. The analysis of the refraction allows to know whether it can be ignored or not, basing on several experimental conditions: desired accuracy, focal length, mirror thickness, angle of incidence and distance of the observed points. Moreover, the results show that the calibration method without a known pattern — which is more flexible in practice — leads to similar performances, assuming that scene features are scattered enough. Finally, the metrological experiment shows that the accuracy obtained is satisfying for localization of a mobile robot.

VII. Appendix - Markers and Landmarks

A flexible calibration procedure should be as automatic as possible. To do so, we have developed an automatic feature detection and label identification. For scene features, we don't use interest points, but rather circular landmarks similar to [21], with a black bullseye and a circular code (see Fig. 9). Mirror markers are white reflecting disks on black disks. Such colors allow to easily classifying markers from landmarks.

A. Automatic detection

In order to get the most user friendly protocol, we have developed an automatic ellipses detector. The only input is an image, and the outputs are the characteristics (location, major and minor axis length) of every black or white elliptical region in the image. The detector is described by the following:

- Regional extrema in image's gray levels thanks to morphological operations.
- Watershed and blob analysis.
- Classify elliptical regions.

A region is classified as elliptical if: $\frac{\mathcal{A}}{\pi ab} > 0.99$, where $\mathcal{A}$, a and b are respectively the region's area, the major and the minor semi-axis' length of the ellipse that has the same normalized second central moments as the region.

B. Automatic subpixel refinement

For scene landmarks we use concentric circles, because the perspective projection of a circle's center can be recovered. We have developed an automatic subpixel detector based on a pattern matching. Fig. 11 shows the detector initialization on the first line, and result on the second line. First column is the image, second is the pattern and third is the absolute difference between image and pattern. The images histograms of last column are equalized. When a false detection occurs, the candidate is rejected if the root mean square error is greater than 0.07 (for an image's gray level from 0 to 255).

A constrained nonlinear optimization fits image content with respect to a homography transformation and the landmark's photonic response (see Fig. 12). The photonic response is defined by 5 parameters: a high value, a low value, and for the transition: the slope p of the line and the two arcs' radius r_1 and r_2. Such a pattern matching algorithm was first developed by Lavest [22], with just an affine transformation and a Levenberg Marquardt optimization. Our method is more adapted to real data because divergent cases can be avoided. At this stage, for a calibrated camera, the localization of a landmark could be found back thanks to (3).

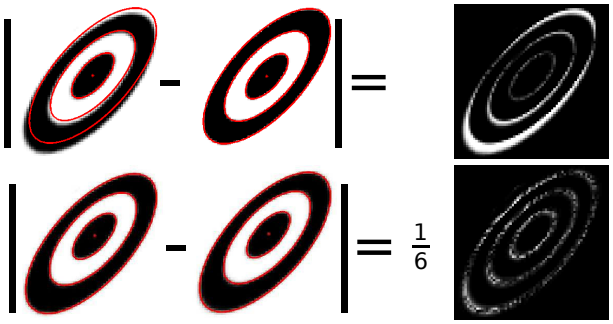

Fig. 11. Subpixel detection of concentric ellipses, before and after convergence (homographic transformation)

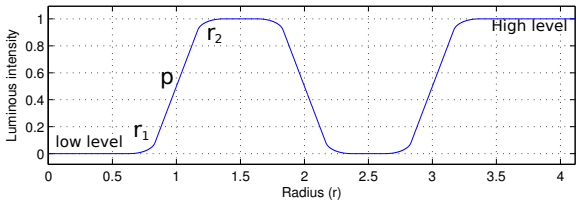

Fig. 12. Photonic response of a landmark

C. Automatic label identification

First, mirror markers are automatically labeled thanks to their asymmetrical geometry. Second, the circular code allows to automatically label landmarks, no matter how they are seen (directly or through the mirror). A circular code is composed of 3 parts: a header H common to every landmark, a label L, and a parity bit P (see Fig. 13). Here we choose 7 bits for the header and 8 bits for the label, expressed in Gray code. The code must be read counterclockwise for a direct view, and clockwise for an mirrored landmark (see Fig. 14). The header's choice is not random. To find it, an exhaustive exploration of the circular codes is done. A circular code is valid, for a direct view, if and only if:

- The header is found back only one time in counterclockwise direction.
- The header is never found back in clockwise direction.

The best header is achieved when a maximum number of messages are valid. In this case, several maxima were possible, so we choose one with the most of transitions: 104 in decimal base code (*i.e.* 1011100 in Gray code). The number of allowed labels is 212 over $2^8 = 256$.

Thanks to this header choice, the circular code is readable even if the landmark is mirrored. Note that orientation of the target is now found back. Thus, the whole pose (localization and orientation) can be computed.

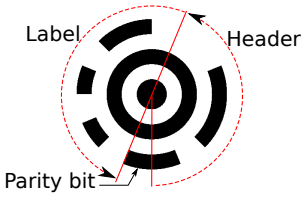

Fig. 13. Circular code's landmark, H = 104, L = 76 and P = 1

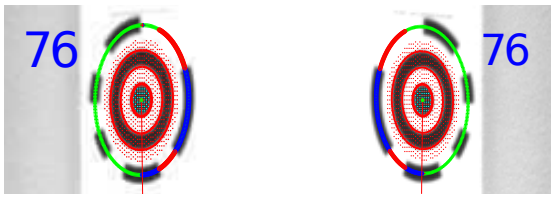

Fig. 14. Circular code identified (L = 76) for a direct view (left) and through a mirror (right)

REFERENCES

[1] E. Mouragnon, "Reconstruction 3d et localisation simultannée de caméras mobiles : une approche temps-réel par ajustement de faisceaux local," Ph.D. dissertation, Ecole Doctorale Sciences Pour l'Ingénieur de Clermont-Ferrand, 2007.

[2] B. Lamprecht, S. Rass, S. Fuchs, and O. Kyamakya, "Extrinsic camera calibration for an on-board two-camera system without overlapping field of view," in *ITSC07*, 2007.

[3] S. Esquivel, F. Woelk, and R. Koch, "Calibration of a multi-camera rig from non-overlapping views," in *DAGM07*, 2007, pp. 82–91.

[4] P. Lébraly, E. Royer, O. Ait-Aider, and M. Dhome, "Calibration of non-overlapping cameras - application to vision-based robotics," in *BMVC*. British Machine Vision Association, 2010.

[5] B. Clipp, J. Kim, J. Frahm, M. Pollefeys, and R. Hartley, "Robust 6dof motion estimation for non-overlapping, multi-camera systems," in *WACV08*, 2008, pp. 1–8.

[6] J. Frahm, K. Koeser, and R. Koch, "Pose estimation for multi-camera systems," in *DAGM04*, 2004, pp. 286–293.

[7] M. N. and D. J., "Camera calibration using reflections in planar mirrors and object reconstruction using volume carving method," *Imaging Science Journal, The*, vol. 52, pp. 117–130(14), 2004.

[8] J. Gluckman and S. K. Nayar, "Catadioptric stereo using planar mirrors," *Int. J. Comput. Vision*, vol. 44, no. 1, pp. 65–79, 2001.

[9] P. Sturm and T. Bonfort, "How to compute the pose of an object without a direct view?" 2006, pp. II:21–31.

[10] J. A. Hesch, A. I. Mourikis, and S. I. Roumeliotis, "Determining the camera to robot-body transformation from planar mirror reflections," in *IROS08*, vol. 1, Nice, France, Sept 22 – Sept 26 2008.

[11] ——, "Mirror-based extrinsic camera calibration," in *In The Eighth International Workshop on the Algorithmic Foundations of Robotics (WAFR'08)*, Mexico, Dec 22 – Dec 26 2008.

[12] R. Kumar, A. Ilie, J.-M. Frahm, and M. Pollefeys, "Simple calibration of non-overlapping cameras with a mirror," *CVPR08*.

[13] A. R. François, G. G. Medioni, and R. Waupotitsch, "Mirror symmetry 2-view stereo geometry," *Image and Vision Computing*, vol. 21, no. 2, pp. 137–143, February 2003.

[14] D. F. Dementhon and L. S. Davis, "Model-based object pose in 25 lines of code," *IJCV*, vol. 15, pp. 123–141, 1995.

[15] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on PAMI*, vol. 22, no. 11, pp. 1330–1334, 2000.

[16] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004.

[17] J.-M. Lavest, M. Viala, and M. Dhome, "Do we really need an accurate calibration pattern to achieve a reliable camera calibration?" in *ECCV '98*. London, UK: Springer-Verlag, 1998, pp. 158–174.

[18] C. Gramkow, "On averaging rotations," *J. Math. Imaging Vis.*, vol. 15, no. 1-2, pp. 7–16, 2001.

[19] M. Moakher, "Means and averaging in the group of rotations," *SIAM J. Matrix Anal. Appl.*, vol. 24, no. 1, pp. 1–16, 2002.

[20] X. Pennec, "Computing the Mean of Geometric Features Application to the Mean Rotation," INRIA, Tech. Rep. RR-3371, 03 1998. [Online]. Available: http://hal.inria.fr/inria-00073318/en/

[21] D. López de Ipi na, P. R. S. Mendonça, and A. Hopper, "Trip: A low-cost vision-based location system for ubiquitous computing," *Personal Ubiquitous Comput.*, vol. 6, no. 3, pp. 206–219, 2002.

[22] J. M. Lavest, M. Viala, and M. Dhome, "Quelle précision pour une mire d'étalonnage," *Traitement du Signal*, vol. 16, no. 3, 1999.