Int J Comput Vis (2012) 96:162–174
DOI 10.1007/s11263-011-0457-8

Planar Motion Estimation and Linear Ground Plane Rectification using an Uncalibrated Generic Camera

Pierluigi Taddei · Ferran Espuny · Vincenzo Caglioti

Received: 28 May 2009 / Accepted: 3 May 2011 / Published online: 19 May 2011

Abstract We address and solve the self-calibration of a generic camera that performs planar motion while viewing (part of) a ground plane. Concretely, assuming initial sets of correspondences between several images of the ground plane as known, we are interested in determining both the camera motion and the geometry of the ground plane. The latter is obtained through the rectification of the image of the ground plane, which gives a bijective correspondence between pixels and points on the ground plane.

We initially propose a method to determine the camera motion by using the motion flow between pairs of images. We perform this step with no need of camera calibration. Our solution requires the fixed ground point of the camera motion to be visible on both images.

Once the camera motion is known, either by using our method or by other alternative means (e.g. GPS-based), we show that the rectification of the ground plane can be determined linearly from at least three images up to a scale factor. Experimental results on real images are presented at the end of the paper to validate the proposed methods.

This paper was written during an internship at Politecnico di Milano of Ferran Espuny, who received the financial support of the Spanish project MTM2006-14234-C02-01.

P. Taddei (✉)
Joint Research Centre of the European Commission, Ispra, Italy
e-mail: pierluigi.taddei@polimi.it

F. Espuny
Dept. d'Àlgebra i Geometria, Universitat de Barcelona, Barcelona, Spain
e-mail: fespuny@ub.edu

V. Caglioti
Politecnico di Milano, Milano, Italy
e-mail: vincenzo.caglioti@polimi.it

Keywords Self-calibration · Plane rectification · Visual odometry · Generic camera · Motion flow · Planar motion

1 Introduction

Robot localization is a fundamental process in mobile robotics applications. One way to determine the displacements and measure the movement of a mobile robot is using dead reckoning systems (such as monitoring the wheels revolutions or integrating accelerometers output). However these systems are not reliable since they provide noisy measurements and tend to diverge after few steps (Borenstein and Feng 1996).

Visual odometry, i.e. methods based on visual estimation of the motion through images captured by one or more cameras, is exploited to obtain more reliable estimates. Many approaches to visual odometry are based on perspective cameras. Due to the narrow viewing cone of this camera model, the persistence of features during an image sequence is short, increasing the error cumulation. On the other hand, visual odometry systems based on panoramic cameras require accurate calibration. These solutions are summarized in Sect. 2.

Our purpose is to work with uncalibrated general cameras, not necessarily central, so as to benefit from the possibility of panoramic viewing, leading to long feature persistence, and the simplicity of set-up, avoiding the need for calibration. The generic camera model, which associates one projection ray to each individual pixel, represents the most general mathematical model to describe a projection system (Grossberg and Nayar 2001; Sturm and Ramalingam 2004). Under this model, the relation among points on the ground plane and the image plane is not parametric and thus standard visual odometry techniques cannot be applied.

In this paper we present a method for the self-calibration of a generic camera that performs planar motions on a textured planar ground floor, while viewing this plane. We assume that the generic camera is continuous, meaning that neighbor pixels have neighbor corresponding points on the ground floor. We focus on the sensor calibration bounded to the ground plane, which can later be used to recover the camera planar motion on the same ground plane.

Calibration from pattern methods exist for the generic camera (Sturm and Ramalingam 2004). Ramalingam et al. (2005) use (motion) flow lines to self-calibrate a general central camera from pure rotations and translations of the camera. The self-calibration of a generic central camera was also solved using at least two infinitesimal rotations in Espuny (2007). They deal with central cameras and simple motions, whereas we deal with general cameras (including non-central ones) and planar motions. To our knowledge, the problems of planar motion estimation and ground plane rectification with uncalibrated generic cameras were only addressed in Caglioti and Taddei (2008), from which this paper is an extended and improved version.

The initial and final positions of the camera induced by any planar motion are always related by a particular rototranslation, i.e. the motion can be described by a center of rotation and a rotation angle, except in the case of pure translations. We estimate the motion flow induced by the rototranslation, i.e. the set of image points that matches an initial point at some stage of the motion (Ramalingam et al. 2005). For such motions, in fact, the flow curves are represented by closed paths around the imaged center of rotation. Since the camera is general, though, the curves are not given by ellipses, as in the perspective case, but have a more general form. A sampling of the motion flow is recovered from the initial set of correspondences either by fusion, when more than two images of the same motion are available, or by interpolation. We follow this latter approach since we assume that only two images are available for the same rototranslation on the ground plane.

Given the sampled motion flow, we show how to estimate with good precision both the imaged center and the angle of rotation. The center represents the only invariant point of the function describing the motion flow, i.e. its path coincides with a point. The rotation angle, instead, is recovered by counting the number of samples required to cover a closed path around the center.

Notice that our approach cannot be applied in case of images related by pure translations or displacements with small rotations. In this case, in fact, the motion flow is singular and it is not represented by closed curves.

Given multiple subsequent planar motions parallel to a ground plane, we show that it is possible to perform a rectification of the ground plane up to a scale factor. The extracted correspondences are the image of points related by a rotation whose parameters can be estimated with our first algorithm. A single set of correspondences, however, is not enough to perform a valid rectification, since it does not constrain the distance of points w.r.t. the center of rotation. We show that a second planar motion around a different center suffices to recover a valid set of constraints and to perform the rectification up to a scale factor.

If the two center positions on the ground plane are known, in fact, we can consider any valid triangle formed by these two points and a third point on the plane. The angles of this triangle can be fixed either by exploiting the motion flow (Caglioti and Taddei 2008) or directly using the feature correspondences as we show in our algorithm. With this latter approach, moreover, it is possible to exploit more than two planar motions to recover the rectification.

We then show that, given the previous estimations, it is possible to recover a continuous function representing the motion flow. This function describes the path of all image points for any rotation angle. It allows us to obtain a final estimation of the visual odometry.

Our approach differs from the one described in Caglioti and Taddei (2008) since we directly perform the rectification and then estimate the flows. This is done by conducting a single linear optimization with fewer parameters w.r.t. a set of subsequent optimization steps yielding to better results. In Caglioti and Taddei (2008), instead, the feature correspondences are used to estimate a continuous function representing the motion flow of each planar motion. These functions are exploited to recover enough constraints related to the rectification to conduct a second optimization.

We outline the components of our algorithm in Fig. 1. Notice that initially we need to select a set of image pairs related to valid distinct planar motions, but once the motion parameters are estimated and the ground plane rectified we can rectify all the image sequence.

The paper is organized as follows. In Sect. 3 we formulate the problems of the center and rotation angle estimation from two images. Section 4 describes how to register two input images and how to recover a sampling of the motion flow once the registration map is known. The problem of motion estimation (imaged center and angle) from a motion flow is solved in Sect. 5. In Sect. 6 we formulate the problem of ground plane rectification from two planar motions. Section 7 describes how to linearly restore the ground plane exploiting at least two different planar motions, i.e. given at least two sets of motion flows around different image points. Furthermore, we show how to exploit the rectification in order to recover a continuous estimation of the motion flow. In Sect. 8 we show good results on real data related to the visual odometry and to the ground plane rectification. Finally, Sect. 9 concludes the paper.

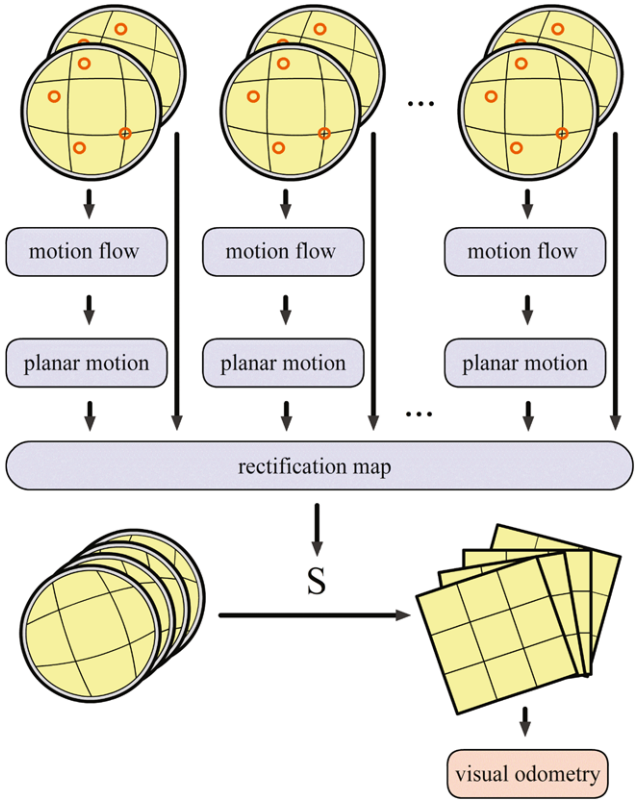

Fig. 1 Outline of the algorithm. Given a set of pixel correspondences between two images, we initially estimate the motion flow and use it to estimate the planar motion parameters. Then, given at least two motion sequences, we estimate a rectification map that can be used to rectify the overall image sequence. The output of our algorithm can be later used to perform visual odometry

2 Previous Work

In the last years methods to estimate the robot motion (ego-motion) based on visual information provided by cameras have gained attention. Early methods were based on estimation of the optical flow from image sequences in order to retrieve ego-motion. McCarthy and Barnes (2004) presented a review of the most promising methods.

Other approaches exploited stereo vision. Nister et al. (2004) proposed a method based on triangulation between stereo pairs and feature tracking in time-sequence of stereo pairs, without any prior knowledge or assumption about the motion and the environment. Comport et al. (2007) developed a visual odometry system based on a stereo pair which directly uses all gray-scale information available leading to very robust and precise results.

Agrawal and Konolige (2006) proposed an integrated, real-time system involving both stereo estimate in the disparity space and a GPS sensor in a Kalman filter framework. GPS-based systems can be sufficiently accurate for large areas but they can not be used in indoor environments and require a support framework, which prevents their use, e.g., for planetary exploration. For such application, Mars Exploration Cheng et al. (2006) employed a feature detection and tracking system in a stereo image pair; using maximum likelihood estimation the change in position and altitude for two or more pairs of stereo images is determined.

Davison (2003) proposed a real-time framework for ego-motion estimation for a single camera moving through general unknown environments. The method is based on a Bayesian framework that detects and tracks a set of features (usually corner points or lines). Assuming the rigidity in the scene, the motion of the image features allows to estimate the motion of the camera; therefore, the complete camera trajectory and a 3D map of all the observed features can be recovered. Wang et al. (2005) and Benhimane and Malis (2006) estimated the homographies between image sequences of (piece-wise) planar scenes to retrieve robot motion but both methods require camera calibration.

Usually, 3D reconstruction from images taken by a moving uncalibrated perspective camera is performed through auto-calibration. Auto-calibration from planar scenes requires either non-planar motion (Triggs 1998), or several planar motions with different altitudes of the camera w.r.t. the ground plane (Knight et al. 2003). Caglioti and Gasparini (2007), instead, performed visual odometry without going through calibration from a single planar motion. All these previous works assume a perspective camera, possibly affected by radial distortion, as the underlying camera model.

Visual odometry systems based on non perspective cameras were proposed by Gluckman and Nayar (1998), who extended standard techniques to omnidirectional cameras. Bunschoten and Krose (2003) used a central catadioptric camera to estimate the relative pose relationship from corresponding points in two panoramic images via the epipolar geometry; the scale of the movement is subsequently estimated via the homography relating planar perspective images of the ground plane. Corke et al. (2004) developed a visual odometry system for planetary rover based on a catadioptric camera; they proposed a method based on robust optical flow estimation from salient visual features tracked between pairs of images. These methods assume a calibrated central omnidirectional camera, whereas our approach instead considers generic camera models that can also deal with non central cameras. Moreover by constraining the application to planar motion we do not require calibration.

Notation. Elements on the ground plane will be represented by capital letters, whereas elements on the image plane will be represented by lower case letters.

3 Planar Motion Estimation Problem

We consider a planar motion of a generic camera, not necessarily central, viewing a ground plane Π. By planar motion

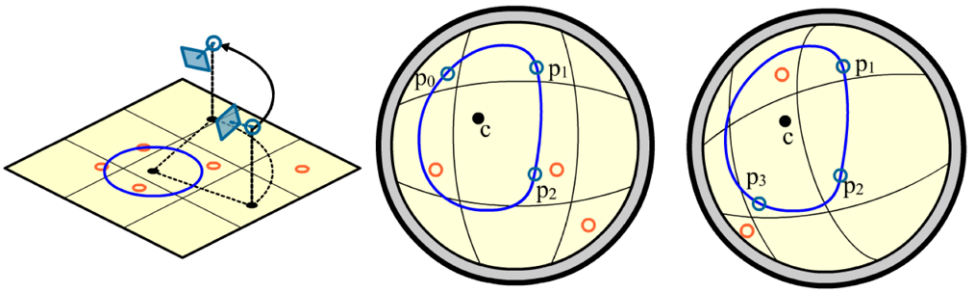

Fig. 2 The motion estimation problem setup. The camera performs a planar motion parallel to the ground plane Π. The motion parameters are the center of rotation C and the angle of rotation α. The *images* show a particular motion flow for a given point p_0 in the image plane. Refer to Sect. 4.2 for a description of the relation among p_0, p_1, p_2, and p_3

we mean that the camera translates on a plane parallel to the ground plane Π, while rotating non-trivially about an axis orthogonal to Π (see Fig. 2, left). This is common, for instance, in the case of a robot moving in indoor environments.

The camera planar motion is equivalent to a rotation about an axis orthogonal to Π. We denote by C the intersection of the rotation axis with Π and by α the angle of rotation. The center C and the angle α determine the camera motion. We aim at recovering the angle α and the image c of the center of rotation C. We denote by $\mathcal{R} \subset \mathbb{R}^2$ the image region corresponding to the ground plane.

We assume given a set of correspondences $\{p_i \leftrightarrow p_i'\}$ between two images I and I'; the correspondences are assumed to be image of points lying on the ground plane Π. We assume that the calibration map of the generic camera is continuous, i.e. that neighbor pixels of the camera have neighbor corresponding viewing rays. We assume that the image c of the center of rotation C is visible on both I and I'; in particular, we do not consider motions close to pure translations. Finally, we assume that the imaged center c is distant from the border of the image domain $\mathcal{R}$, in such a way that some of the given correspondences lie around this point.

A *motion flow* is represented by a function relating any image point p to an image curve containing all points that match p at some stage during the motion. Due to the continuity of the calibration map, in the case of a planar motion w.r.t. the ground plane, the motion flow of any imaged point is represented by a closed curve around the center of rotation (Ramalingam et al. 2005). These curves are ellipses in the case of perspective cameras, but have a more complex form in the case of a generic projection (see Fig. 2).

In order to estimate α and c we will exploit a sampling of the continuous motion flow curve through any point in the image domain $\mathcal{R}$. As we explain later, this sampling can be obtained by just iterating a matching function between the two images. Therefore, previous to the motion estimation we need to estimate a *registration* function

$$\Gamma : \mathcal{R} \to \mathcal{R}. \tag{1}$$

This registration Γ maps any pixel p from the imaged ground plane on the first image to its corresponding pixel p' on the second image.

4 Generic Registration and Motion Flow Estimation

Assume that we are given a set of correspondences $\{p_i \leftrightarrow p_i'\}$ between two images I and I', $1 \leq i \leq n$. We first need to estimate the registration Γ between the two images, which, by exploiting the continuity assumption, we model as the composition of a planar homography with a bivariate b-spline function. The registration exactly corresponds to a planar homography for images acquired by a pinhole camera, but not for images corresponding to a generic camera; we use a closed-form estimation of the homography to obtain a guess for the registration map that serves as initial value for the spline fitting.

Once we determine the registration function Γ between the two images, we can iterate it repeatedly so as to obtain an estimation of the motion flow as we explain in Sect. 4.2. In practice, the obtained motion flow does not correspond to the sampling of the real planar motion flow (closed) curve. Caglioti and Taddei (2008) describe possible improvements exploiting only the corresponding points. Instead, we will directly use this initial estimation for the computation of the center and angle of rotation avoiding, at this stage, the estimation of the motion flow.

4.1 Generic Registration between Two Images

We model the registration map Γ between the two images I and I' as the composition

$$\Gamma = \tilde{\Gamma} \circ H, \tag{2}$$

being H a planar homography and being $\tilde{\Gamma}$ a bivariate cubic b-spline function.

A cubic bivariate b-spline is defined as Prautzsch et al. (2002):

$$\tilde{\Gamma}(u,v) = \sum_{j=1}^{j=N_j} \sum_{i=1}^{i=Ni} B_i(u,v) \cdot B_j(u,v) \cdot k_{ij} \tag{3}$$

where $B_i(u,v)$ and $B_j(v,v)$ are $\mathbb{R}^2 \to \mathbb{R}$ polynomial functions that depends solely on the position of the $N_i \cdot N_j$ b-spline knots. Moreover each knot is associated to a two dimensional vector $k_{ij} = [ku_{ij}, kv_{ij}]^\top$. Notice that $\Gamma(u,v)$ depends linearly from each k_{ij} element.

We use the planar homography H in order to obtain points Hp_i that are closer to the points p'_i than the initial points p_i. H represents the linear component of the transformation whereas $\tilde{\Gamma}$ approximates the additional non linear component.

We estimate the homography as the 3×3 homogeneous matrix that minimizes the re-projection error between the points $q_i := Hp_i$ and the points p'_i. In particular, the points q_i have three homogeneous coordinates

$$q_i = Hp_i = (q_{i,x} : q_{i,y} : q_{i,z}),$$

and we consider their affine coordinates on the image

$$\tilde{p}_i = (q_{i,x}/q_{i,z}, q_{i,y}/q_{i,z});$$

the re-projection error to minimize is given by the formula

$$\sum_i \|\tilde{p}_i - p'_i\|^2. \tag{4}$$

This is a standard problem in Computer Vision; further details can be found, for example, in Hartley and Zisserman (2004).

Once we have estimated the planar homography H, we can apply it to obtain the points $\tilde{p}_i := Hp_i$, which should be mapped by the b-spline $\tilde{\Gamma}$ to the points p'_i. We aim at minimizing the reprojection error of these feature points plus an additional smoothing term, which depends on the second derivatives of the b-spline:

$$\tilde{\Gamma} = \arg\min\big(E_r[\bar{\Gamma}] + \lambda_s E_s[\bar{\Gamma}]\big), \tag{5}$$

where

$$E_r[\bar{\Gamma}] = \sum_i \big\|\bar{\Gamma}(\tilde{p}_i) - p'_i\big\|^2,$$

$$E_s[\bar{\Gamma}] = \sum_j \big\|\bar{\Gamma}_{uu}(q_j)\big\|^2 + \big\|\bar{\Gamma}_{vv}(q_j)\big\|^2 + 2\big\|\bar{\Gamma}_{uv}(q_j)\big\|^2.$$

Here $\bar{\Gamma}_{uu}$, $\bar{\Gamma}_{vv}$ and $\bar{\Gamma}_{uv}$ represent the second partial derivatives of $\bar{\Gamma}$, which are sampled at grid points $\{q_j\}$, and λ_s is a weighting constant (which was manually selected in our experiments).

In order to deal with catadioptric cameras, and in general with cameras with a large field of view, we use a b-spline $\tilde{\Gamma}_p$ whose domain is represented by polar coordinates w.r.t. the image center. Thus given a point $p(u,v)$ on the first image, we transform it in its polar representation w.r.t. the image center and then apply Γ_p to recover the corresponding point $p'(u,v)$ on the second image. This allows us to better fit the b-spline domain to the image region $\mathcal{R}$ representing the ground plane.

4.2 Motion Flow Estimation from Two Images

Let p_0 and p_1 refer to two matching image points, i.e. corresponding to the same 3D point observed in I and I'. Let p_2 be the point that in I' matches to point p_1 in I. Similarly, let p_3 be the point that in I' matches to point p_2 in I, and so forth (see Fig. 2). This sequence may be expressed using the registration function Γ:

$$p_1 = \Gamma(p_0),$$

$$p_2 = \Gamma(p_1) = \Gamma\big(\Gamma(p_0)\big) = \Gamma^2(p_0),$$

$$\ldots$$

$$p_n = \Gamma(p_{n-1}) = \ldots = \Gamma^n(p_0),$$

which reads that the nth point on the motion flow (i.e. after n rotations of angle α) corresponding to a point $p = p_0$ can be obtained by applying n times Γ to the point p.

Moreover, by exchanging the rules of I and I', we can obtain the point p_{-n} on the motion flow by p corresponding to n rotations of angle $-\alpha$:

$$p_{-n} = \Gamma^{-n}(p). \tag{6}$$

In summary, using the registration map Γ we are able to give a sequence of points $\{p_m = \Gamma^m(p)\}_{m\in\mathbb{Z}}$ on the motion flow by a given initial point p. We show that this approach is enough in order to recover the rotation angle α and the imaged center c.

Besides, the estimated sequence of points on the motion flow suffers from drifting, i.e. the points are likely to be wrong for large values of m. In fact, the recovered points $p_m = \Gamma^m(p)$ do not represent the sampling of a closed curve. In Caglioti and Taddei (2008) a solution was given in order to recover motion flows while imposing that they are closed curves. We follow an alternative approach in Sect. 7.3, consisting in the use of the rectification of the images to obtain the motion flow.

5 Planar Motion Estimation from Two Images

In this section we assume that the registration map Γ between two images I and I' is known. Therefore, given a

point p we are able to obtain a sequence of points $\{p_m = \Gamma^m(p)\}_{m\in\mathbb{Z}}$ on the motion flow curve passing through p. Using the map Γ, we show how to compute the imaged center c and the rotation angle α.

5.1 Imaged Center Estimation

The only invariant point of the function Γ is the point c. In fact

$$\forall k \in \mathbb{Z}, \quad \Gamma^k(c) = c. \tag{7}$$

All the other points are displaced according to the motion: their flows are represented by closed curves around c. In particular, these sequences of points all rotate around c either clockwise or counter-clockwise. Let's initially assume that the flows are represented by convex curves. Consider a line crossing c. In this case, all points on one half line are displaced in one half-space and all points on the other half line are displaced in the opposite half space due to the previous property (see Fig. 3).

The previous observation leads to a simple, although effective, way to compute the imaged center. We start by considering an initial segment crossing a point c_0. The segment is sampled at n sites and the displacement given by Γ is evaluated. We then detect a point c_1 which is the point on the segment that divides the sites in two groups w.r.t. the displacement orientation. We call this point the singular point of the segment. We then perform the same step on an orthogonal segment passing through c_1 to detect a point c_2 and so forth. By iterating the process the sequence of $\{c_i\}$ converges to $\bar{c}$.

Consider now the case of non-convex flow curves. Apart degenerate cases, we may assume that locally in the region around the imaged center the convex assumption is still valid. Globally there could be more than one singular point for any line on the view plane. To overcome the problem we repeat the algorithm starting from different initial positions and reducing the segment length at each step. If the current segment contains more than one singular point we randomly choose one and proceed to the next iteration. Figure 4 shows the case of non-convex motion flows and a possible sequence of iterations of the algorithm leading to the correct imaged center.

5.2 Angle Estimation

We estimate α using the following procedure. Let us consider a generic image point p, the sequence of points given by $\Gamma^m(p)$, $m \in \mathbb{Z}$, and a reference line b crossing both $\bar{c}$ and the chosen initial point p (see Fig. 5, left).

Suppose that the recovered function Γ is correct and that the rotation angle is an integral fraction f of 2π. In this case, the point $\Gamma^f(p)$ coincides with p, i.e. the sequence has a period of f. In other words, each application of Γ represents a rotation of angle $\frac{2\pi}{f}$.

In the case that the rotation angle is not an integral fraction of 2π, it is estimated by considering the least common

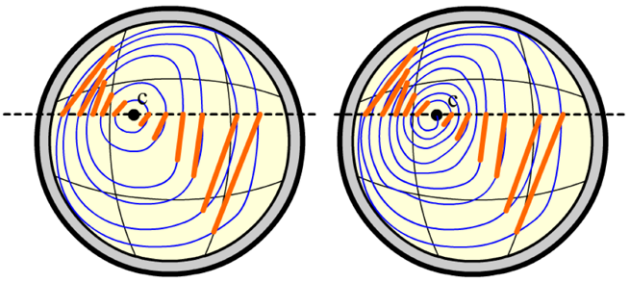

Fig. 3 (Color online) Center estimation. c divides any line crossing it in two sets of points which are displaced by the motion flow in opposite halfspaces. *Red lines* show the displacement vectors for points on the *dashed* segments. *Left image* shows a correct set of flows, whereas *right image* shows flows which suffer from drifting

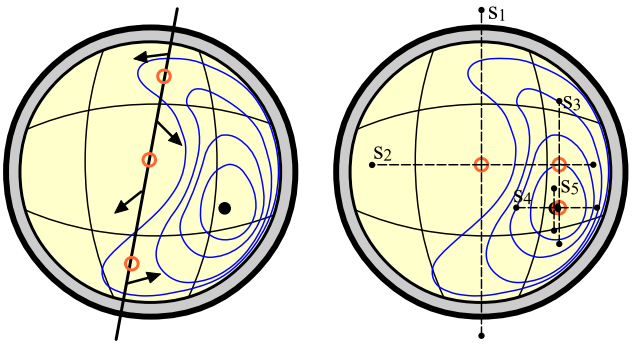

Fig. 4 *Left image*: if the motion flow is represented by non convex curves there could be an odd number of singular points for some lines on the image plane. The singular points split the segment points w.r.t. their displacement direction. *Right image* represents a possible execution of the center estimation algorithm starting from an initial position and leading to the correct imaged center. In our implementation we alternated *horizontal* and *vertical lines*, although the directions can be arbitrarily chosen

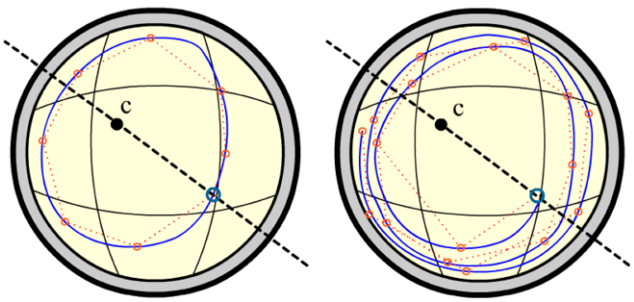

Fig. 5 Angle estimation. Given an initial point p, by counting the number of steps required to cover the closed path, we are able to calculate the roto-translation angles. *Left image* shows a correct motion flow, whereas *right image* shows a flow which suffers from drifting

multiple of 2π and α. This is practically evaluated by considering a large sequence of points which performs several loops around $\bar{c}$. By counting the number of times the sequence traverses the line b and dividing it by the number of points used we obtain a good estimate of α.

Notice, however, that our initial estimate of Γ can be inaccurate and suffer from drifting (see Fig. 5, right). Even so, we observed experimentally that this simple method accurately estimates the rotation angle α.

6 Ground Plane Rectification Problem

We assume given a set of correspondences $\{p_1^i \leftrightarrow p_2^i\}$ between two images I_{p1} and I_{p2}, and also a set of correspondences $\{q_1^i \leftrightarrow q_2^i\}$ between two images I_{q1} and I_{q2}. The two images of each pair are, generally, related by different planar motions with a common ground plane Π; these motions are equivalent to two rotations of angles α_1 and α_2 with axes intersecting the ground plane at different points C_1 and C_2. We require the images of the two rotation centers, respectively denoted by c_1 and c_2, to be visible on the images. Only in this case it is possible to detect a motion flow represented by closed curves and completely contained in the image. This is crucial for the iterative approach that we use to estimate them.

Our goal is to rectify any of the images of the ground plane. To this effect, we can consider the *rectification* map S from the image domain $\mathcal{R}$ on the ground plane Π,

$$S : \mathcal{R} \to \Pi, \tag{8}$$

that sends a pixel $p \in \mathcal{R}$ to its back-projection on the ground plane (see Fig. 6). This map S can be thought as the restriction of the generic calibration map (Ramalingam et al. 2005) to the image region corresponding to the ground plane.

The rectification map S allows us to relate the rotational motion between any two camera positions with the associated motion flow between their respective images. If we denote by Γ_j and by R_j the registration function and relative motion, respectively, between the jth camera pair (see Fig. 6), then the relation is $S\Gamma_j = C_j + R_j(S - C_j)$. This relation allows us to determine the rectification map from the initial sets of correspondences.

7 Estimating the Rectification Map

Assume that we are given a set of n_1 point correspondences $\{p_1^i \leftrightarrow p_2^i\}$, $i = 1, \ldots, n_1$, between two images I_{p_1} and I_{p_2}, and also a set of n_2 point correspondences $\{q_1^i \leftrightarrow q_2^i\}$, $i = 1, \ldots, n_2$, between two images I_{q_1} and I_{q_2}. We consider the two corresponding planar motions, described by

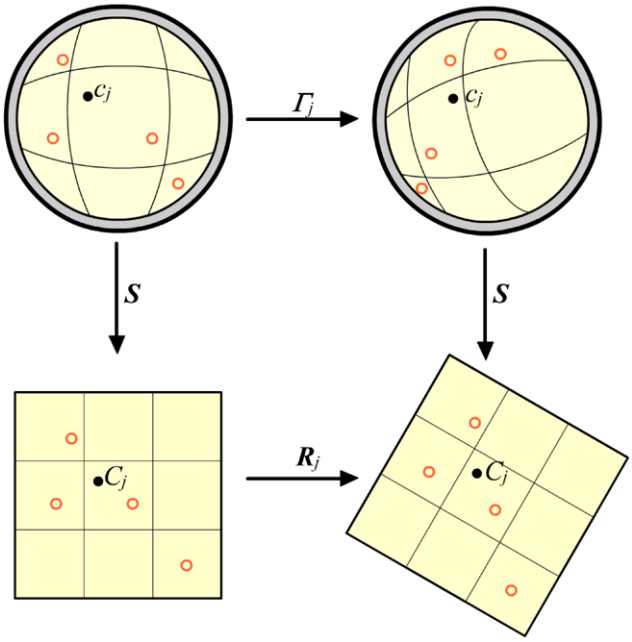

Fig. 6 Rectification of the imaged plane (*left*). The registration function Γ_j corresponds via the rectification map S with the rotation R_j with center C_j, equivalent to the planar motion between the jth pair of cameras

(α_1, C_1) and (α_2, C_2), where the images c_1, c_2 of the centers C_1, C_2 must be contained in $\mathcal{R}$. Notice that I_{p_2} and I_{q_1} might represent the same image, in which case the estimation is conducted using only three images.

We want to rectify the images of the ground plane, i.e. determine the rectification map S defined in (8), for which purpose we assume the motion parameters to be known. In practice, given the initial matching points we can estimate the motion flows Γ_1, Γ_2 corresponding to the two camera planar motions according to Sect. 4. Then, by Sect. 5, we can obtain estimations of the imaged rotation centers c_1, c_2 and the rotation angles α_1, α_2.

7.1 Constraints on the Rectification Map

Without loss of generality, we consider an orthonormal system of reference on the ground plane such that C_1 lies at the origin and C_2 is the point $(1, 0)$. Accordingly, the (known) images c_1, c_2 of the two first rotation centers satisfy:

$$S(c_1) = C_1 = (0, 0), \qquad S(c_2) = C_2 = (1, 0). \tag{9}$$

We denote by R_j the matrix of the rotation on Π with angle α_j, i.e.

$$R_j = \begin{pmatrix} \cos\alpha_j & -\sin\alpha_j \\ \sin\alpha_j & \cos\alpha_j \end{pmatrix}. \tag{10}$$

We consider P_1^i, P_2^i (resp. Q_1^i, Q_2^i) the position on the ground plane of the reconstructed image correspondences

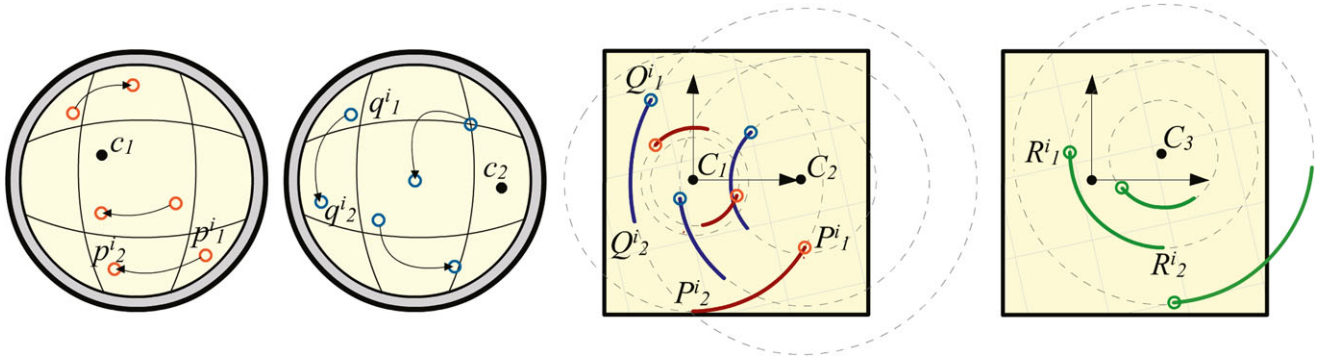

Fig. 7 Representation of the constraints for the ground plane rectification. *First* and *second image* show the feature correspondences and the imaged centers related to the first two planar motions. The *third image* shows the relation of these elements on the rectified plane: notice that the imaged center back projections are used to fix the reference frame and points are displaced according to their respective rotation angle. *Last image* show the back projections of points of an additional planar motion. In this case the position of the imaged center back projection cannot be fixed and must be considered as part of the unknowns

$p^i_1 \leftrightarrow p^i_2$, $1 \le i \le n_1$ (resp. $q^i_1 \leftrightarrow q^i_2$, $1 \le i \le n_2$). Due to the planar motion of the camera, and using that $C_1 = (0, 0)$, the reconstructed image pairs satisfy

$$P^i_2 = R_1 P^i_1, \qquad Q^i_2 = C_2 + R_2(Q^i_1 - C_2). \tag{11}$$

Since the points P^i_j (resp. Q^i_j) are the reconstruction of the points p^i_j (resp. q^i_j) we have that $S(p^i_j) = P^i_j$ (resp. $S(q^i_j) = Q^i_j$). Therefore, by (11), it holds

$$S(p^i_2) = R_1 S(p^i_1), \qquad S(q^i_2) = (C_2 - R_2 C_2) + R_2 S(q^i_1). \tag{12}$$

We show in Fig. 7 a visual representation of the constraints expressed by (9) and (12).

7.2 Determining the Rectification Map

Equations (9) and (12) give us constraints on the unknown rectification map S. The constraints (9) depend on c_1, c_2, i.e. on the imaged centers of rotation, and the constraints (12) depend on the image correspondences and the rotation angles. Since we consider that c_1, c_2, α_1 and α_2 are known, the only unknown in (9) and (12) is the map S.

Assuming that the rectification map S is smooth, we can use a bivariate cubic b-spline as the underlining parametric model for S. For each pixel p on the image domain $\mathcal{R}$, the value $S(p)$ is obtained as a linear product

$$S(p) = \begin{pmatrix} B_p^\top K_x \\ B_p^\top K_y \end{pmatrix} = \begin{pmatrix} B_p^\top & 0 \\ 0 & B_p^\top \end{pmatrix} \begin{pmatrix} K_x \\ K_y \end{pmatrix} =: A_p K, \tag{13}$$

where B_p is a constant array depending on the coordinates of pixel p and $K = (K_x, K_y)$ is the unknown coefficients of the components of the rectification spline. If we choose N knots for the spline on each direction, then the vector K has length $2N^2$ and for every $p \in \mathcal{R}$ the matrix A_p is $2 \times 2N^2$.

Equations (9) and (12) can be written as the following linear system:

$$\begin{cases} A_{c_1} K = & 0, \\ A_{c_2} K = & C_2, \\ (A_{p^i_2} - R_1 A_{p^i_1}) K = & 0, \\ (A_{q^i_2} - R_2 A_{q^i_1}) K = & C_2 - R_2 C_2 \end{cases} \tag{14}$$

which has $2(2 + n_1 + n_2)$ equations in the $2N^2$ unknowns.

The system might result rank deficient due to a bad distribution of correspondences over the image region $\mathcal{R}$ w.r.t. the knots employed for the b-spline. In this case, one can reduce the number of dots and/or use a regularization strategy similar to that explained in Sect. 4 to supply the lack of data in some "square" of the used b-spline grid.

A more robust solution can be employed in case that more than two planar motions are available, which is the typical situation for a moving robot that acquires multiple images. It is possible to extend the linear system in order to include the constraints induced by these additional motions.

Let us consider an additional pair of images I_{r_1}, I_{r_2}, with point correspondences $\{r^i_1 \leftrightarrow r^i_2\}$, $i = 1, \ldots, n_r$, and associated motion flow (α_r, c_r). We call C_r the back projection on the ground plane of c_r. As we did for the initial two planar motions we may assume α_r and c_r as known. Moreover we can evaluate the matrices R_r, $A_{r^i_1}$ and $A_{r^i_2}$.

We cannot fix the coordinates of C_r, since the ground plane reference frame has been already fixed using C_1 and C_2. As result, we can obtain $2n_r$ additional equations to those in (14) by considering $C_r = A_r K$. The corresponding

linear system is

$$\begin{cases} A_{c_1}K = & 0, \\ A_{c_2}K = & C_2, \\ (A_{p_2^i} - R_1 A_{p_1^i})K = & 0, \\ (A_{q_2^i} - R_2 A_{q_1^i})K = & C_2 - R_2 C_2, \\ (A_{r_2^i} - R_r A_{r_1^i} + (R_r - I)A_{c_r})K = & 0. \end{cases} \tag{15}$$

In general by considering $k \geq 2$ planar motions, with n_i point correspondences each, we can build a linear system with $2N^2$ unknowns and $2(2 + \sum_{i=1}^{k} n_i)$ equations. Note that once solved, the ground plane can be rectified and, moreover, the camera motion can be recovered by taking $(\alpha_i, C_i = A_{c_i}K)$.

By employing more than two planar motions it is in general possible to collect a set of well distributed correspondences so that the built linear system (15) has full rank. Otherwise, one can proceed as explained for two motions. We did not encounter rank deficiency problems in our experiments when using three or more motions.

7.3 Motion Flow Given the Rectification

We showed that the rectification can be performed directly from sets of feature correspondences related to more than two images, once the motion parameters are known. Exploiting the rectification map S, we can recover each motion flow related to the jth pair of images whose parameters have been recovered. This is done by back-projecting points on any circle centered on the rotation center C_j (remember Fig. 2). In particular, given a point $p \in \mathcal{R}$ the jth motion flow f_p^j is represented by

$$f_p^j(\beta) = S^{-1}\left(C_j + \left(R_\beta \cdot S(p) - C_j\right)\right), \tag{16}$$

where R_β is the matrix of rotation of angle $0 \leq \beta \leq 2\pi$ and C_j is represented by $C_j = S(c_j)$.

In order to estimate the function f_p^j we need to invert the mapping S. Since there is no closed-form available for S^{-1}, we estimate it using a local weighted mean approximation (Goshtasby 1988). With respect to the algorithm described in Caglioti and Taddei (2008), this approach gives consistent curves, i.e. closed not intersecting flows. With respect to the initial motion flow computed in Sect. 4.2, the motion flow obtained using (16) is fully consistent with the planar motion of the camera and, in particular, it has a fixed point.

8 Experimental Validation

We tested our algorithm on two sets of real images. For each pair of images, we manually extracted a set of correspondences and then recovered the registration function Γ.

In the first set of experiments the ground truth for the angle and center of rotation was manually measured during the camera setup. We employed a cubic b-spline in polar coordinates using 6 equidistant knots per dimension. In particular we conducted the optimizations described in (3) with $\lambda_S = 0.01$. Notice however that this value was selected after a manual tuning of the parameter and it is strictly related to the camera's geometry and to the distribution of features w.r.t. the b-spline knots.

We run the algorithm to recover the imaged center $\bar{c}$ starting from each different pixel of the image region. Apart small areas close to the region boundary the recovered value was the same, showing its robustness to the choice of initial starting point. We studied the performance with respect to the number of b-spline parameters; the observed behavior is an initial improvement of the performance as the number of parameters increases, followed by a divergence (probably due to overfitting), in correspondence to a further increase of the parameter number. In our experiments with 70 correspondences the estimation was good up to a b-spline using a grid of 35×35 knots. In all experiments the standard deviation of the recovered points was less than 6 pixel on 1024×768 images (Fig. 8 shows the recovered centers on two of such experiments).

The iterative algorithm for the angle recovery converged rapidly to the average value in all our experiments. An example of the evaluation is shown in Fig. 9. The initial point used for the angle estimation slightly influences the recovered value. We ran the algorithm several times from initial random coordinates. In all experiments, the standard deviation was less than 0.2 degree (see Fig. 10).

As for the center of rotation, we ran the algorithm several times varying the number of parameters of the b-spline. The estimation was robust since the standard deviation was always below 0.4 degree. Also, for this experiment, the accuracy increased with the number of parameters up to a given value (in our experiments with 70 correspondences we registered a value around 30–35); above this threshold the estimation diverged (see Fig. 10).

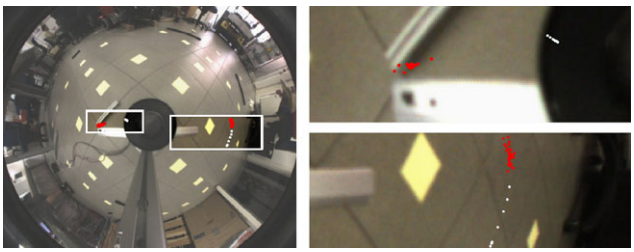

Fig. 8 (Color online) Estimated centers varying the b-spline parameters for two different planar motions. *Red points* represent inlier points, which are correctly distributed around the ground truth, while *white points* represent outliers. Notice that, in these experiments, the outliers are easily identified. *Right image* shows a close up of the two areas

Notice that, since we are modeling Γ using a regular b-spline with equally spaced knots, the distribution of the extracted features influences the estimation. In particular regions containing few correspondences are not recovered correctly since the estimation is dominated by the regularization term of (5). Since we iterate this function on the regions surrounding the imaged center of rotations it is important that, in these regions, features are well distributed w.r.t. the controlling spline knots.

We then performed the rectification by estimating the parameters of two motions given three images. These values were used to perform the ground floor rectification obtaining the result shown in Fig. 11. We first rectify each single image and then align the rectified planes exploiting the planar motion parameters. The images intensity difference is also shown in Fig. 11. In this case the errors are due to changes in the environment, such as the camera position, occlusions due to objects lying outside the plane and lightening variation. The regions containing the extracted features were correctly reconstructed. The rectification is consistent w.r.t. all the considered images, although it presents errors around the image boundary, where there is no information due to the occlusion of the camera, and at the outer regions, due to the resolution of the sensor.

Finally, we evaluated S^{-1} and estimated a set of motion flows for the two roto-translations (see Fig. 12). The resulting curves are consistent with the notion of motion flows and are more accurate than the ones presented in Caglioti and Taddei (2008). Notice that the rectification boundary errors present around the camera sensor are reflected in errors on the motion flow close to the center of the image.

In the second set of experiments we employed the outdoor sequence of the panoramic camera of the *RAWSEEDS* project (Bonarini et al. 2006). These images have been captured by a panoramic camera mounted on a moving robot that recorded odometry information using a GPS receiver and a set of linear laser scanners. We compared the angle estimation results using our algorithm with the provided ground truth, represented by the robot position along its trajectory (last row of Fig. 13 shows the ground truth of the planar motions used). In all experiments the angle error was

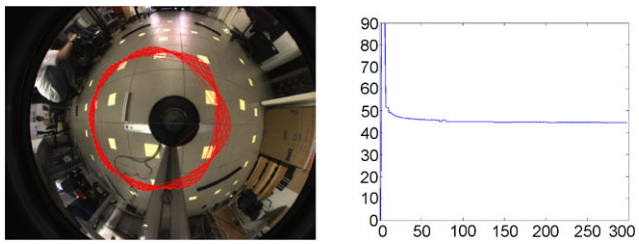

Fig. 9 (Color online) Angle estimation. The *red curve* represents the iterative application of the registration function Γ, starting from an initial point p. The segment connecting p to the estimated center is shown in *white*. It is used to count how many times the curve traverses it. *Right image* shows the estimation of the angle (in degrees) w.r.t. the number of points used (i.e. the number of applications of Γ)

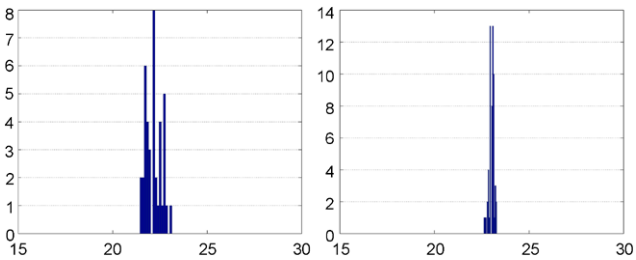

Fig. 10 Distribution of the angle estimation. The *left histogram* shows the distribution by varying the initial starting point whereas the *right histogram* shows the distribution by changing the b-spline number of parameters while maintaining the same starting position. In both cases the standard deviation is small and the values recovered are close to the measured ground truth

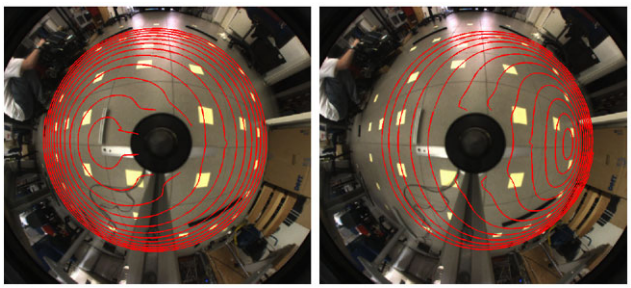

Fig. 12 Motion flow related to the planar motion used to perform the rectification in the first experiment. The curve errors close to the camera sensor are due to the lack of correspondences

Fig. 11 Rectification comparison. *Left*: rectification of the first image using the mapping S. *Center*: rectification of the second image using the mapping S followed by a rotation of angle $-\alpha_1$ about C_1. *Right*: intensity difference between the two images

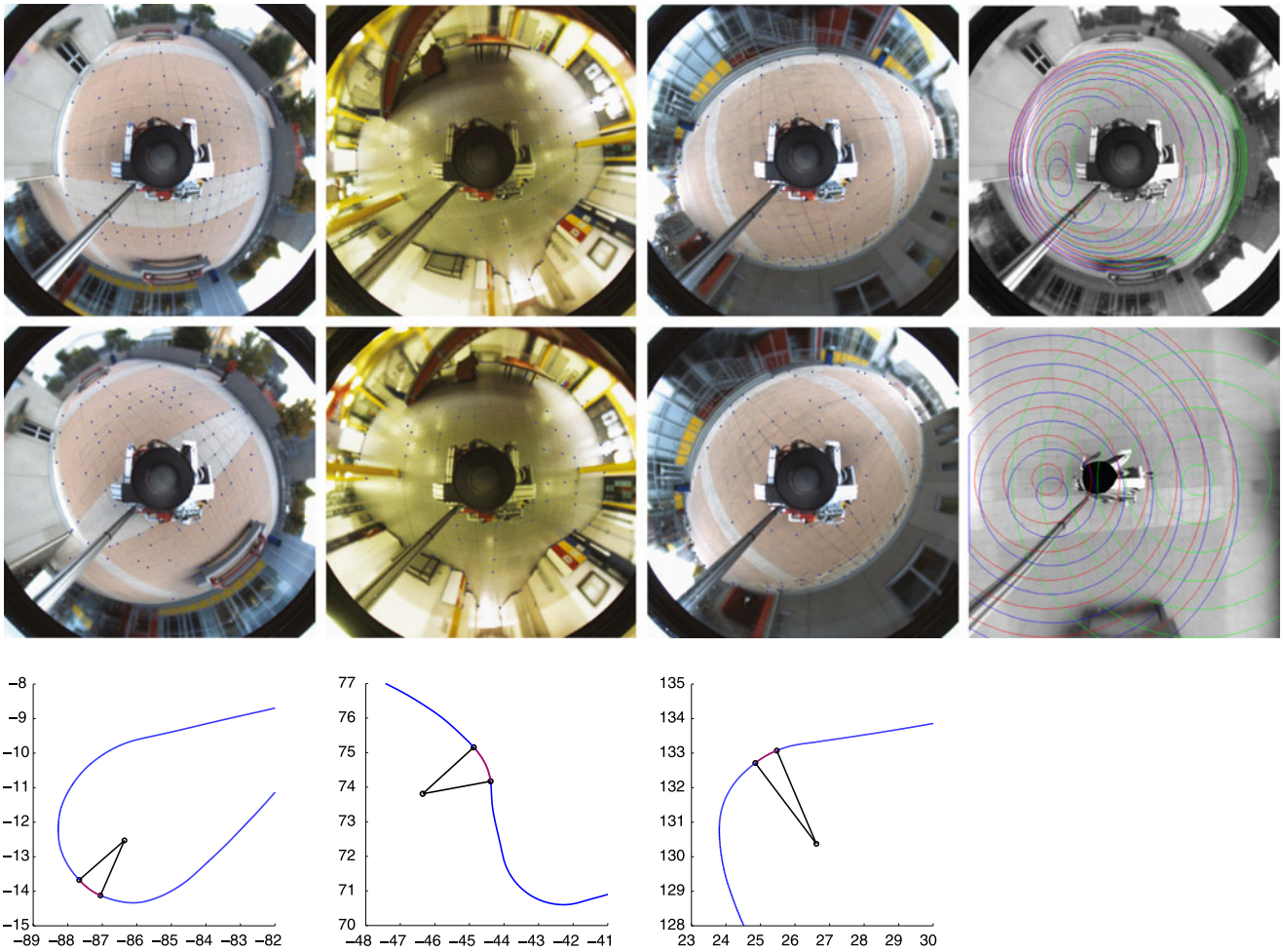

Fig. 13 The *first three columns* show the frames of the *RAWSEEDS* sequence used to recover the planar motions and then to perform the rectification. The *first two images* of each column shows the exploited features overlayed on the start and ending frames. The *third image* represents the ground truth position of the robot along its trajectory and the associated rotation to each pair of images. We use this ground truth to evaluate the quality of the planar motion estimation. The *last column* shows the motion flows related to the three planar motions. The *top image* represents all the motion flows on one image of the sequence. The *bottom image* shows the same curves on the rectified plane

within 1°, which is consistent with the angle estimation from the sampled ground truth trajectory. The motion flow and planar motion estimation took 4.5 seconds on average.

We then estimated the rectification map using three different planar motions related to different time instants (which are shown in the first three columns of Fig. 13). In this case the linear system presented 442 equations in 72 unknowns and was solved in 0.05 seconds. The recovered rectification map was, then, applied to each frame in order to generate a rectified version of the sequence (see Fig. 14). Each frame was represented by a 600×600 image generated in an average time of 27 seconds.

Notice that after applying the rectification it is possible to employ standard algorithms to recover the roto-translations relating each subsequent pair of images. This approach could be used, in principle, to recover motion information also in the case of pure translations. Finally we recovered the motion flows related to the roto-translations employed to perform the rectification. A sampling of these curves is shown in the last column of Fig. 13.

In order to quantitatively evaluate the reconstruction we considered additional frames from the sequence that were not employed to estimate the rectification and the planar motions. In particular we extracted the corners of the visible regular tiles pattern of the ground plane from the selected verification frames. We manually associated each point to its correspondent corner of a regular grid (see Fig. 15). We then apply function S and register the rectified point set to the regular grid by estimating a rigid transformation and a scale factor (assuming that the grid tiles have unitary side length). This is performed by conducting a non-linear least square minimization of the corresponding feature distances. Figure 16 shows the histogram of the optimization residual error of all employed features. The mean error is within 3%

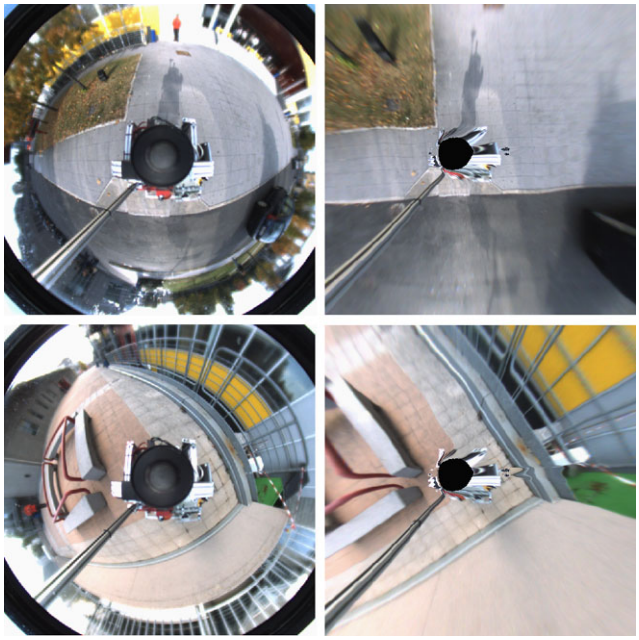

Fig. 14 *Left column*: frames from the *RAWSEEDS* sequence. *Right column*: rectified frames using the recovered rectification map

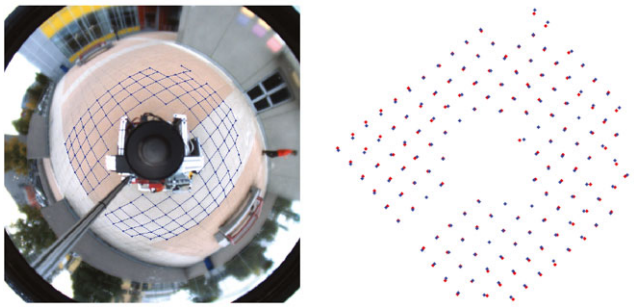

Fig. 15 (Color online) *Left*: one of *RAWSEEDS* frame used to evaluate the rectification showing the verification grid used to measure the reconstruction estimation. *Right*: the rectified position of the verification features (*red*) and the corresponding ground truth grid corners (*blue*). Notice that neither the verification frame nor the verification features were exploited to perform the rectification estimation itself

of the tile side length and we observed that the corners with the higher residual error are related to regions close to the boundary of the b-spline domain.

9 Conclusion

We presented a proof of concept to perform self-calibration bounded to the ground plane of a generic camera that performs planar motions. In particular, we showed how to recover both the imaged center and the angle of rotation given a set of correspondences between two images of the ground plane when the equivalent roto-translation center is visible. Then, given the motion parameters of at least two planar

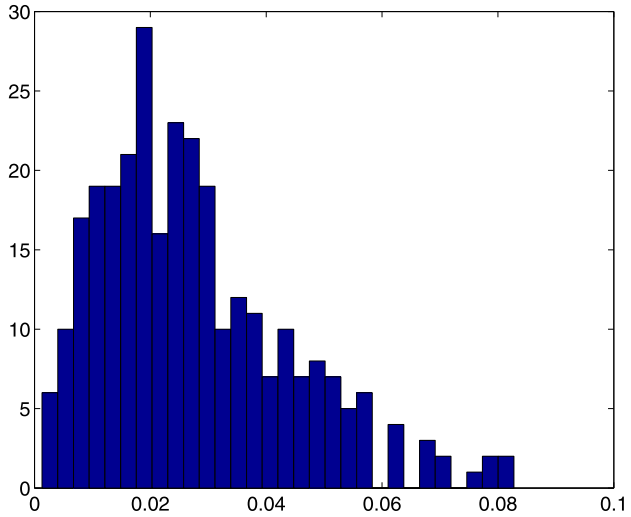

Fig. 16 Histogram of the residual errors of each grid corner. The error shown is normalized w.r.t. the grid tile side length

motions with a common ground plane we showed that it is possible to recover the plane rectification by solving a linear system that exploits the feature correspondences. Moreover, once the rectification map is known we can recover both the camera motion and the motion flow of any point on the image plane.

The angle and center estimation have been shown to be robust w.r.t. the number of degrees of freedom used in the modelization, i.e. the parameters of the b-spline function, and w.r.t. the initial point of the iteration. The rectification step yields good results both to recover a rectified image of the ground plane and to estimate the motion flow curves.

Future works will be aimed at exploiting the whole sequence of rectified images to segment the region representing points lying on the ground plane w.r.t. points not on the plane. This would allow us to build a method for the detection of outliers in the initial set of correspondences, and thus to gain robustness in the overall approach.

References

Agrawal, M., & Konolige, K. (2006). Real-time localization in outdoor environments using stereo vision and inexpensive gps. In *ICPR*.

Benhimane, S., & Malis, E. (2006). Homography-based 2D visual servoing. In *ICRA*.

Bonarini, A., Burgard, W., Fontana, G., Matteucci, M., Sorrenti, D. G., & Tardos, J. D. (2006). Rawseeds: Robotics advancement through web-publishing of sensorial and elaborated extensive data sets. In *IROS*.

Borenstein, J., & Feng, L. (1996). Measurement and correction of systematic odometry errors in mobile robots. *IEEE Transactions on Robotics and Automation*, *12*, 869–880.

Bunschoten, R., & Krose, B. (2003). Visual odometry from an omnidirectional vision system. In *ICRA*.

Caglioti, V., & Gasparini, S. (2007). Uncalibrated visual odometry for ground plane motion without auto-calibration. In *International Workshop on Robotic Vision*.

Caglioti, V., & Taddei, P. (2008). Planar motion estimation using an uncalibrated general camera. In *OMNIVIS*.

Cheng, Y., Maimone, M., & Matthies, L. (2006). Visual odometry on the mars exploration rovers—a tool to ensure accurate driving and science imaging. *IEEE Robotics & Automation Magazine*, *13*, 54–62.

Comport, A., Malis, E., & Rives, P. (2007). Accurate quadrifocal tracking for robust 3d visual odometry. In *ICRA*.

Corke, P., Strelow, D., & Singh, S. (2004). Omnidirectional visual odometry for a planetary rover. In *IROS*.

Davison, A. (2003). Real-time simultaneous localization and mapping with a single camera. In *ICCV*.

Espuny, F. (2007). A closed-form solution for the generic self-calibration of central cameras from two rotational flows. In *VISAPP*.

Gluckman, J., & Nayar, S. K. (1998). Ego-motion and omnidirectional cameras. In *ICCV*.

Goshtasby, Ardeshir (1988). Image registration by local approximation methods. *Image and Vision Computing*, *6*, 255–261.

Grossberg, M., & Nayar, S. (2001). A general imaging model and a method for finding its parameters. In *ICCV*.

Hartley, R. I., & Zisserman, A. (2004). *Multiple view geometry in computer vision* (2nd ed.). Cambridge: Cambridge University Press.

Knight, J., Zisserman, A., & Reid, I. (2003). Linear auto-calibration for ground plane motion. In *CVPR*.

McCarthy, C., & Barnes, N. (2004). Performance of optical flow techniques for indoor navigation with a mobile robot. In *ICRA*.

Nister, D., Naroditsky, O., & Bergen, J. (2004). Visual odometry. In *CVPR*.

Prautzsch, H., Boehm, W., & Peluszny, M. (2002). *Bezier and b-spline techniques*. Berlin: Springer.

Ramalingam, S., Sturm, P., & Lodha, S. K. (2005). Towards generic self-calibration of central cameras. In *OMNIVIS*.

Sturm, P., & Ramalingam, S. (2004). A generic concept for camera calibration. In *ECCV* (Vol. 2, pp. 1–13).

Triggs, B. (1998). Autocalibration from planar scenes. In *ECCV*.

Wang, H., Yuan, K., Zou, W., & Zhou, Q. (2005). Visual odometry based on locally planar ground assumption. In *ICIA*.