3D Reconstruction of Planar Patches Seen by Omnidirectional Cameras

Molnar Jozsef*, Robert Frohlich*, Dmitry Chetverikov† and Zoltan Kato*
*Institute of Informatics, University of Szeged, P.O. Box 652, H-6701 Szeged, Hungary
†Geometric Modelling and Computer Vision Laboratory, MTA SZTAKI, Kende u. 13-17, H-1111 Budapest, Hungary

***Abstract*—We propose a novel solution for reconstructing planar surface patches from omnidirectional camera images. The theoretical foundation relies on variational calculus, which yields a closed form solution for the normal vector a 3D planar surface patch, when a homography is known between the corresponding image region pairs. The method is quantitatively evaluated on a large set of synthetic data. Experimental results confirm that the method provides good reconstructions in real-time.**

I. Introduction

The importance of piecewise planar object representation in 3D stereo has been recognized by many researchers. There are various solutions in case of standard perspective cameras, many of them are making use of the plane induced homography: Habbecke and Kobbelt used a small plane, called 'disk', for surface reconstruction [1], [2]. They proved that the normal is a linear function of the camera matrix and homography. By minimizing the difference of the warped images, the surface is reconstructed. Furukawa proposed using a small patch for better correspondence [3]. The surface is then grown with the expansion of the patches. The piecewise planar stereo method of Sinha *et al.* [4] uses shape from motion to generate an initial point cloud, then a best fitting plane is estimated, and finally an energy optimization problem is solved by graph cut for plane reconstruction. Fraundorfer *et al.* [5] used MSER regions to establish corresponding regions pairs. Then a homography is calculated using SIFT detector inside the regions. Planar regions are then grown until the reprojection error is small. Although the role of planar regions in 3D reconstruction has been noticed by many researchers, the final reconstruction is still obtained via triangulation for most state-of-the-art methods. Planar objects are only used for better correspondences or camera calibration.

Homography is used in many applications including pose estimation [6], tracking [7], [8], structure from motion [9] as well as recent robotics applications with focus on navigation [10], vision and perception. Efficient homography estimation methods exist for classical perspective cameras [11], but these methods are usually not reliable in case of omnidirectional sensors. The difficulty of homography estimation with omnidirectional cameras comes from the non-linear projection model yielding shape changes in the images that make the direct use of these methods nearly impossible. Recently, the geometric formulation of central omnidirectional systems was extensively studied [12], [13], [14], [15], [16], [17]. The internal calibration of such cameras depends on these geometric models, which can be solved in a controlled environment, using special calibration patterns [16], [18], [19], [17]. When the camera is calibrated, which is typically the case in practical application, then image points can be lifted to the surface of a unit sphere providing a unified model independent of the inner non-linear projection of the camera. The big advantage of such a generic model is that many concepts from standard projective geometry (in particular homographies or stereo triangulation techniques) remain valid for central omnidirectional cameras. For example, homography can be estimated using these spherical points [7], [8]. Classical keypoint detectors, such as SIFT [20], are also widely used [9], [7] for omnidirectional images, but big variations in shape resolution and non-linear distortion challenges keypoint detectors as well as the extraction of invariant descriptors, which are key components of reliable point matching. For example, proper handling of scale-invariant feature extraction requires special considerations in case of omnidirectional sensors, yielding mathematically elegant but complex algorithms [21]. In [9], a correspondence-less algorithm is proposed to recover relative camera motion. Although matching is avoided, SIFT features are still needed because camera motion is computed by integrating over all feature pairs that satisfy the epipolar constraint. Epipolar geometry of omnidirectional camera pairs have also been studied [22], which can be used to establish dense stereo matches.

In this paper, we propose a region-based method to reconstruct planar surface patches from corresponding regions in an omnidirectional camera pair. Instead of establishing point correspondences and using triangulation, we make use of a region-based homography estimation method [23] and derive a closed form formula for computing the normal of the 3D plane from the estimated homography. Our derivation is based on variational calculus, hence we avoid any camera-specific consideration yielding a general formula for spherical cameras. While the internal parameters of the camera are assumed to be known (which is typical in real life applications), the relative pose can also be obtained from the estimated homography by classical factorization methods [6]. Therefore knowing the internal parameters and a homography induced by the 3D scene plane, we are able to efficiently recover the plane parameters. Quantitative evaluation on a large set of synthetic data confirms the real-time performence, efficiency and robustness of the proposed solution.

II. Omnidirectional camera model

A unified model for central omnidirectional cameras was proposed by Geyer and Daniilidis [14], which represents central panoramic cameras as a projection onto the surface of a unit sphere. This formalism has been adopted and models for the internal projection function have been proposed by

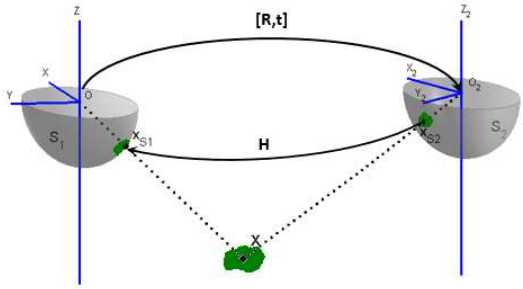

Fig. 1. Omnidirectional camera model

Micusik [15] and subsequently by Scaramuzza [24] who derived a general polynomial form of the internal projection valid for any type of omnidirectional camera.

Given a scene plane π, let us formulate the relation between its images $\mathcal{D}$ and $\mathcal{F}$ in a pair of omnidirectional cameras represented by the unit spheres $\mathcal{S}_1$ and $\mathcal{S}_2$ (see Fig. 1). Assuming that the first camera coordinate system is the reference frame, a 3D plain point $\mathbf{X} \in \pi$ is projected onto $\mathcal{S}_1$ by a simple central projection:

$$\mathbf{x}_{\mathcal{S}1} = \frac{\mathbf{X}}{\|\mathbf{X}\|} \tag{1}$$

The relative pose of the second camera is composed of a rotation $\mathbf{R}$ and translation $\mathbf{t} = (t_1, t_2, t_3)^T$, acting between the cameras $\mathcal{S}_1$ and $\mathcal{S}_2$. Thus the image in the second camera of the same 3D point $\mathbf{X}$ is

$$\mathbf{x}_{\mathcal{S}2} = \frac{\mathbf{R}\mathbf{X} + \mathbf{t}}{\|\mathbf{R}\mathbf{X} + \mathbf{t}\|} \tag{2}$$

Because of the single viewpoint, the mapping of plane points $\mathbf{X} \in \pi$ to the camera spheres $\mathcal{S}_i, i = 1, 2$ is bijective (unless π is going through the camera center, in which case π is invisible) and planar homographies stay valid for omnidirectional cameras too [7]. Denoting the normal by $\mathbf{n} = (n_1, n_2, n_3)^T$ and distance of π to the origin of $\mathcal{S}_1$ by d, the standard planar homography $\mathbf{H}$ is composed up to a scale factor as [7], [23]

$$\mathbf{H} \propto \mathbf{R} + \frac{1}{d}\mathbf{t}\mathbf{n}^T \tag{3}$$

Basically, the homography transforms the rays as $\mathbf{x}_{\mathcal{S}1} \propto \mathbf{H}\mathbf{x}_{\mathcal{S}2}$, hence the transformation induced by the planar homography $\mathbf{H}$ between the spherical points is also bijective. Thus a point $\mathbf{X}$ on the plane and its spherical images $\mathbf{x}_{\mathcal{S}1}$, $\mathbf{x}_{\mathcal{S}2}$ are related by [23]

$$\mathbf{X}_\pi = \lambda_1 \mathbf{x}_{\mathcal{S}1} = \lambda_2 \mathbf{H}\mathbf{x}_{\mathcal{S}2} \Rightarrow \mathbf{x}_{\mathcal{S}1} = \frac{\lambda_2}{\lambda_1}\mathbf{H}\mathbf{x}_{\mathcal{S}2} \tag{4}$$

Hence $\mathbf{x}_{\mathcal{S}1}$ and $\mathbf{H}\mathbf{x}_{\mathcal{S}2}$ are on the same ray yielding [23]

$$\mathbf{x}_{\mathcal{S}1} = \frac{\mathbf{H}\mathbf{x}_{\mathcal{S}2}}{\|\mathbf{H}\mathbf{x}_{\mathcal{S}2}\|} = \Psi(\mathbf{x}_{\mathcal{S}2}) \tag{5}$$

Clearly, the function Ψ is fully determined by the homography $\mathbf{H}$, hence estimating the homography parameters using *e.g.* the algorithm of [23] provides the bijective mapping Ψ between the spherical points of the omnidirectional camera pair.

III. Normal vector computation

We now derive a simple, closed form solution to reconstruct the normal vector of a 3D planar surface patch from a pair of corresponding image regions and known omnidirectional cameras. Once the normal vector $\mathbf{n}$ is determined, d can be easliy computed based on (3) as shown *e.g.* in [11]. Although differential geometric approaches were used to solve various problems in projective 3D reconstruction, the approach proposed here is unique for omnidirectional cameras to the best of our knowledge. For example, [25], [26] are about generic surface normal reconstruction using point-wise orientation- or spatial frequency disparity maps. Unlike [25], [26], which consider only projective camera and uses a parameterization dependent, non-invariant representation; we use a general omnidirectional camera model and our method avoids point correspondences and reconstructs a planar surface from the induced planar homography between image regions.

The notations in this section are widely used in classical differential geometry. For vectors and tensors we use bold letters and italics for the coordinates. Standard basis is defined by three orthonormal vectors $\mathbf{e}_1$, $\mathbf{e}_2$, and $\mathbf{e_3}$. 3D points $\mathbf{X} \in \mathbb{R}^3$ are identified with their coordinates in the standard basis $\mathbf{X} = X^1\mathbf{e}_1 + X^2\mathbf{e}_2 + X^3\mathbf{e}_3$ or $\mathbf{X} = X^k\mathbf{e}_k$ using the summation convention (repeated indices in superscript and subscript position mean summation). Considering the visible part of the scene object as a reasonably smooth surface $\mathbf{S}$ embedded into the ambient 3D space, $\mathbf{S}$ is represented by the general (Gauss) coordinates u^1 and u^2 as

$$\mathbf{S}\left(u^1, u^2\right) = X^1\left(u^1, u^2\right)\mathbf{e}_1 + X^2\left(u^1, u^2\right)\mathbf{e}_2 + \\ + X^3\left(u^1, u^2\right)\mathbf{e}_3 = X^k\left(u^l\right)\mathbf{e}_k \tag{6}$$

The tangent space to surface $\mathbf{S}$ at a surface point $\left(u^1, u^2\right)$ is spanned by the local (covariant) basis vectors $\mathbf{S}_k = \frac{\partial \mathbf{S}}{\partial u^k}$, $\mathbf{S}_k = \mathbf{S}_k\left(u^1, u^2\right)$, $k \in \{1, 2\}$. The corresponding contravariant basis vectors $\mathbf{S}^l$, $l \in \{1, 2\}$ are defined to satisfy the identity $\mathbf{S}^l \cdot \mathbf{S}_k = \delta^l_k$, where δ^l_k ,$l \in \{1, 2\}$, $k \in \{1, 2\}$ is the Cronecker delta and the scalar product is denoted by dot.

The normal vector of the surface is defined by the cross product $\mathbf{N} = \mathbf{S}_1 \times \mathbf{S}_2$. Surface area element is defined by the triple scalar product $|\mathbf{n}\mathbf{S}_1\mathbf{S}_2| \doteq \mathbf{n} \cdot (\mathbf{S_1} \times \mathbf{S_2})$ where $\mathbf{n} = \frac{\mathbf{N}}{|\mathbf{N}|}$ is the *unit normal vector* of the surface. The cross-tensor of the normal vector $\mathbf{N}_\times = \mathbf{S}_2\mathbf{S}_1 - \mathbf{S}_1\mathbf{S}_2$ is a difference of two dyadic products of the local basis vectors. Dyadic (direct) products are denoted by a simple sequence of the constituent vectors.

The dot product between dyads and vectors is defined such that $\mathbf{u}\mathbf{v} \cdot \mathbf{w} = (\mathbf{v} \cdot \mathbf{w})\,\mathbf{u}$. From this, using the triple product expansion formula $\mathbf{N}_\times \cdot \mathbf{v} = \mathbf{N} \times \mathbf{v}$ for any vector $\mathbf{v}$.

As usual, for the representation of vectors and second order tensors purely with their coordinates we use row vectors and two dimensional matrixes. The coordinate representation of a non-scalar quantity $\mathbf{Q}$ is denoted by $[\mathbf{Q}]$.

A. Camera model independent correspondence equations

Let us now have a closer look at the relation between a 3D point $\mathbf{X}$ and its 2D images (x^1_i, x^2_j) and (x^1_j, x^2_j) in a

pair of cameras i and j. This has been studied in [27] for establishing an affine transformation between the images of a known surface using known projection functions. First we briefly overview the derivation of this relation and then we will show how to use it for computing normal vectors of planar surface patches from corresponding image regions.

An image of the scene is basically a 3D→2D mapping given by two smooth projection functions, the so called coordinate functions: $x^1\left(X^1, X^2, X^3\right)$ and $x^2\left(X^1, X^2, X^3\right)$ with (x^1, x^2) being the 2D image coordinates. Herein, we don't assume any special form of these coordinate-functions except their deifferentiability w.r.t. the spatial coordinates X^1, X^2, X^3. If the projected points are on the surface (6) too, the image coordinates depend on the general parameters as well:

$$\begin{aligned} x^1 &= x^1\left(X^1\left(u^1,u^2\right), X^2\left(u^1,u^2\right), X^3\left(u^1,u^2\right)\right) \\ x^2 &= x^2\left(X^1\left(u^1,u^2\right), X^2\left(u^1,u^2\right), X^3\left(u^1,u^2\right)\right) \end{aligned} \quad (7)$$

We suppose that the mapping in (7) is bijective in a small open disk around the point $\left(u^1, u^2\right)$. Assuming that both the projection functions and the surface are smooth, these are the conditions for differentiability and local invertibility. The differential $[d\mathbf{u}] = \begin{bmatrix} du^1 & du^2 \end{bmatrix}^T$ represents a point shift on the surface with its effect on the image being $d\mathbf{x} \approx \mathbf{J} \cdot d\mathbf{u}$ where $[d\mathbf{x}] = \begin{bmatrix} dx^1 & dx^2 \end{bmatrix}^T$ and the Jacobian $\mathbf{J}$ of the mapping is invertible.

Now consider a camera pair, distinguishing them with indices i and j (note that i, j indices used in subscript position doesn't stand for "covariant" quantities). Since $\mathbf{J}_i$ is invertible, we can establish correspondences between the images taking the same point shift $d\mathbf{u} \approx \mathbf{J}_i \cdot d\mathbf{x}_i$:

$$d\mathbf{x}_j = \mathbf{J}_j \cdot \mathbf{J}_i^{-1} \cdot d\mathbf{x}_j = \mathbf{J}_{ij} \cdot d\mathbf{x}_j \quad (8)$$

where $\mathbf{J}_{ij}$ is the Jacobian of the $\mathbf{x}_i \rightarrow \mathbf{x}_j$ mapping. Now consider the derivative of a composite function $f\left(X^l\left(u^k\right)\right)$, $l \in \{1,2,3\}$, $k \in \{1,2\}$:

$$\frac{\partial f}{\partial u^k} = \frac{\partial X^l}{\partial u^k}\frac{\partial f}{\partial X^l} = \mathbf{S}_k \cdot \nabla f, \quad (9)$$

where ∇f is the gradient w.r.t. the spatial coordinates and $\mathbf{S}_k$ is the local basis vector along the parameter line u^k. Applying this result to the projection functions, the components of the Jacobians take the following form:

$$\begin{aligned} [\mathbf{J}_i] &= \begin{bmatrix} \mathbf{S}_1 \cdot \nabla x_i^1 & \mathbf{S}_2 \cdot \nabla x_i^1 \\ \mathbf{S}_1 \cdot \nabla x_i^2 & \mathbf{S}_2 \cdot \nabla x_i^2 \end{bmatrix}, \\ [\mathbf{J}_j] &= \begin{bmatrix} \mathbf{S}_1 \cdot \nabla x_j^1 & \mathbf{S}_2 \cdot \nabla x_j^1 \\ \mathbf{S}_1 \cdot \nabla x_j^2 & \mathbf{S}_2 \cdot \nabla x_j^2 \end{bmatrix} \end{aligned} \quad (10)$$

Substituting (10) into (8), the products of the components of (10) enter into $\mathbf{J}_{ij}$. For example, the determinant becomes

$$\det[\mathbf{J}_i] = \nabla x_i^1 \cdot \left(S_1 \cdot \nabla x_i^1\right)\left(S_2 \cdot \nabla x_i^2\right) - \left(S_2 \cdot \nabla x_i^1\right)\left(S_1 \cdot \nabla x_i^2\right) \quad (11)$$

which can be expressed by dyadic products equivalent to the surface normal's cross tensor as

$$\begin{aligned} \det[\mathbf{J}_i] &= \nabla x_i^1 \cdot \left(S_1 S_2 - S_2 S_1\right) \cdot \nabla x_i^2 \\ &= -\nabla x_i^1 \cdot \mathbf{N}_\times \cdot \nabla x_i^2 = -\left|\mathbf{N}\right| \left|\nabla x_i^1 \mathbf{n} \nabla x_i^2\right| \end{aligned} \quad (12)$$

where $|\mathbf{N}|$ is the absolute value (length) of the surface normal vector. The components of the Jacobian $\mathbf{J}_{ij}$ are then [27]:

$$[\mathbf{J}_{ij}] = \frac{1}{|\nabla x_i^1 \mathbf{n} \nabla x_i^2|} \begin{bmatrix} |\nabla x_j^1 \mathbf{n} \nabla x_i^2| & |\nabla x_i^1 \mathbf{n} \nabla x_j^1| \\ |\nabla x_j^2 \mathbf{n} \nabla x_i^2| & |\nabla x_i^1 \mathbf{n} \nabla x_j^2| \end{bmatrix} \quad (13)$$

The above quantities are all invariant first-order differentials: the gradients of the projections and the surface unit normal vector. Note that (13) is a general formula: neither a special form of projections, nor a specific surface is assumed here, hence it can be applied for any camera type and for any reasonably smooth surface.

Herein, we will show how to use the above formula for computing the normal vector $\mathbf{n}$, when both the projection functions and the Jacobian $\mathbf{J}_{ij}$ are known. Let us write the matrix components estimated either directly with affine estimator or taking the derivatives of an estimated planar homography[1] as:

$$[\mathbf{J}_{ij}]_{est} = \begin{bmatrix} a_1^1 & a_2^1 \\ a_1^2 & a_2^2 \end{bmatrix} \quad (14)$$

To eliminate the common denominator we can use ratios, which can be constructed using either row, column, or cross ratios. Without loss of generality, we deduce the equation for the 3D surface normal using cross ratios $\frac{a_1^1}{a_2^2}$ and $\frac{a_2^1}{a_1^2}$. After rearranging equation $[\mathbf{J}_{ij}]_{est} = [\mathbf{J}_{ij}]$ we obtain:

$$\begin{aligned} \mathbf{n} \cdot \left[a_2^2\left(\nabla x_i^2 \times \nabla x_j^1\right) - a_1^1\left(\nabla x_j^2 \times \nabla x_i^1\right)\right] &= 0 \\ \mathbf{n} \cdot \left[a_1^2\left(\nabla x_j^1 \times \nabla x_i^1\right) - a_2^1\left(\nabla x_i^2 \times \nabla x_j^2\right)\right] &= 0 \end{aligned} \quad (15)$$

Here we have two (known) vectors, both perpendicular to the normal:

$$\begin{aligned} \mathbf{p} &= \mathbf{n} \cdot \left[a_2^2\left(\nabla x_i^2 \times \nabla x_j^1\right) - a_1^1\left(\nabla x_j^2 \times \nabla x_i^1\right)\right] \\ \mathbf{q} &= \mathbf{n} \cdot \left[a_1^2\left(\nabla x_j^1 \times \nabla x_i^1\right) - a_2^1\left(\nabla x_i^2 \times \nabla x_j^2\right)\right] \end{aligned} \quad (16)$$

Thus the surface normal can readily be computed as

$$\mathbf{n} = \frac{\mathbf{p} \times \mathbf{q}}{|\mathbf{p} \times \mathbf{q}|}. \quad (17)$$

In the remaining part of this section, we will show how to compute the coordinate gradients $\nabla x_k^l, k = i,j; l = 1,2$ w.r.t. spatial coordinates and $\mathbf{J}_{ij}$ in (13) for an omnidirectional camera pair.

B. Computing coordinate gradients for the spherical camera model

The Jacobian (13) includes the coordinate gradients w.r.t. spatial coordinates. Herein, we derive these quantities for the general spherical camera model discussed in Section II. For the sake of simplicity, the calculations are done in the camera coordinate system, but coordinate gradients calculated below can be easily transformed into any world coordinate system by applying the rotation between that word coordinate frame and the camera.[2]

Following [16], [24], we assume that the camera coordinate system is in $\mathcal{S}$, the origin (which is also the center of the sphere) is the projection center of the camera and the z axis is

[1] The derivatives of a planar homography provides exact affine components.

[2] Gradients are constructed by derivation, hence the translation to any other world coordinate system cancels out from the formulae.

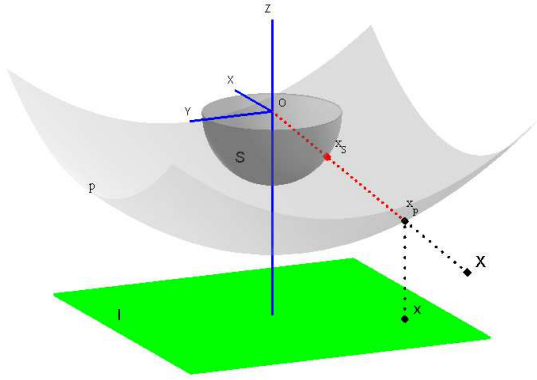

Fig. 2. Projection sphere $\mathcal{S}$ parametrized via the omni image $\mathcal{I}$.

the optical axis of the camera which intersects the image plane in the *principal point* (see Fig. 2). To represent the nonlinear (but symmetric) distortion of central omnidirectional optics, [16], [24] places a surface p between the image plane and the unit sphere $\mathcal{S}$, which is rotationally symmetric around z. The details of the derivation of p can be found in [16], [24]. Herein, as suggested by [16], we will use a fourth order polynomial $p(\|\mathbf{x}\|) = a_0 + a_2\|\mathbf{x}\|^2 + a_3\|\mathbf{x}\|^3 + a_4\|\mathbf{x}\|^4$ which has 4 parameters (a_0, a_2, a_3, a_4) representing the internal parameters of the camera (only 4 parameters as a_1 is always 0 [16]). The bijective mapping $\Phi : \mathcal{I} \to \mathcal{S}$ is composed of 1) lifting the image point $\mathbf{x} \in \mathcal{I}$ onto the p surface by an orthographic projection

$$\mathbf{x}_p = \begin{bmatrix} \mathbf{x} \\ a_0 + a_2\|\mathbf{x}\|^2 + a_3\|\mathbf{x}\|^3 + a_4\|\mathbf{x}\|^4 \end{bmatrix} \tag{18}$$

and then 2) centrally projecting the lifted point $\mathbf{x}_p$ onto the surface of the unit sphere $\mathcal{S}$:

$$\mathbf{x}_\mathcal{S} = \Phi(\mathbf{x}) = \frac{\mathbf{x}_p}{\|\mathbf{x}_p\|} \tag{19}$$

Thus the omnidirectional camera projection is fully described by means of unit vectors $\mathbf{x}_\mathcal{S}$ in the half space of $\mathbb{R}^3$ and these points correspond to the unit vectors of the projection rays. The function Φ is fully defined by the internal camera parameters (a_0, a_2, a_3, a_4), which can be determined using *e.g.* the calibration toolbox of Scaramuzza [16], [24]. Therefore the unit projection sphere $\mathcal{S}$ can be naturally parametrized by the omni image coordinates $\mathbf{x} = (x^1, x^2)$. Spatial points $\mathbf{X} \in \mathbb{R}^3$ are identified by the unit sphere points (*i.e.* the directions) denoted by $\mathbf{x}_\mathcal{S}$, $\mathbf{x}_\mathcal{S} \cdot \mathbf{x}_\mathcal{S} \equiv 1$, and their distance from the projection sphere's center denoted by $x^3 \equiv \|\mathbf{X}\|$ such that

$$\mathbf{X} = x^3\mathbf{x}_\mathcal{S}. \tag{20}$$

Note that the above equation follows from (1) and it is a non-Cartesian parameterization of $\mathbb{R}^3$ from which the gradients of the first two parameters (x^1, x^2) are required. The identity

$$\delta_k^l = \frac{\partial \mathbf{X}}{\partial x^k} \cdot \frac{\partial x^l}{\partial \mathbf{X}} = \mathbf{g}_k \cdot \nabla x^l \tag{21}$$

is the basic differential geometry relation between the covariant $\mathbf{g}_k = \frac{\partial X}{\partial x^k}$ and contravariant $\nabla x^l = \mathbf{g}^l$ basis vectors of the parameterization. Applying (21) to (20), we have:

$$\begin{aligned} \mathbf{g}_k &= \frac{\partial \mathbf{X}}{\partial x^k} = x^3 \frac{\partial \Phi}{\partial x^k},\ k \in \{1,2\} \\ \mathbf{g}_3 &= \frac{\partial \mathbf{X}}{\partial x^3} = \mathbf{x}_\mathcal{S}\,. \end{aligned} \tag{22}$$

From this the metric tensor components $g_{kl} = \mathbf{g}_k \cdot \mathbf{g}_l$, $k, l \in \{1,2,3\}$ are

$$\begin{aligned} g_{kl} = g_{lk} &= \left(x^3\right)^2 \frac{\partial \Phi}{\partial x^k} \cdot \frac{\partial \Phi}{\partial x^l},\ k,l \in \{1,2\} \\ g_{k3} = g_{3k} &= 0,\ k \in \{1,2\} \\ g_{33} &= \mathbf{x}_\mathcal{S} \cdot \mathbf{x}_\mathcal{S} = 1\,. \end{aligned} \tag{23}$$

Note that the second line of (23) follows from the derivation of the constraint $\mathbf{x}_\mathcal{S} \cdot \mathbf{x}_\mathcal{S} \equiv 1$. Using the basic result from differential geometry $\mathbf{g}^l = g^{lk}\mathbf{g}_k$, where g^{lk} are the components of the inverse metric tensor, and observing that the metric tensor has the special form $\begin{bmatrix} [g_{lk}] & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix}$, the first two contravariant basis vectors (the sought coordinate gradients) can be independently expressed from the third vector such that

$$\begin{aligned} \begin{bmatrix} \nabla x^1 \\ \nabla x^2 \end{bmatrix} &= \begin{bmatrix} g_{11} & g_{12} \\ g_{12} & g_{22} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{g}_1 \\ \mathbf{g}_2 \end{bmatrix} \\ &= \frac{1}{x^3} \begin{bmatrix} \frac{\partial \Phi}{\partial x^1} \cdot \frac{\partial \Phi}{\partial x^1} & \frac{\partial \Phi}{\partial x^1} \cdot \frac{\partial \Phi}{\partial x^2} \\ \frac{\partial \Phi}{\partial x^1} \cdot \frac{\partial \Phi}{\partial x^2} & \frac{\partial \Phi}{\partial x^2} \cdot \frac{\partial \Phi}{\partial x^2} \end{bmatrix}^{-1} \begin{bmatrix} \frac{\partial \Phi}{\partial x^1} \\ \frac{\partial \Phi}{\partial x^2} \end{bmatrix} \end{aligned} \tag{24}$$

In the above equation, coordinate gradients are expressed purely with the unit sphere's local basis vectors $\widetilde{\mathbf{g}}_k = \frac{\partial \Phi}{\partial x^k}$ induced by the image coordinates and the distance between the observed point and the center of the projection sphere x^3. Note that x^3 cancels out from the normal calculation in (17) by division. Once the normal is determined, any component of (13) provides an equation for $\frac{x_i^3}{x_j^3}$.

C. Computing the Jacobian components

Let us now see how to construct the elements a_l^k of the Jacobian matrix in (14) acting directly between the omnidirectional images. Denoting the Cartesian coordinates w.r.t. the centers of the unit spheres representing the cameras i and j by $[\mathbf{x}_i] = \begin{bmatrix} z_i^1 & z_i^2 & z_i^3 \end{bmatrix}^T$ and $[\mathbf{x}_j] = \begin{bmatrix} z_j^1 & z_j^2 & z_j^3 \end{bmatrix}^T$. These spherical points are related by the bijective mapping Ψ as derived in Section II, which can be directly estimated by estimating the homography between the cameras. Its Jacobian $\mathbf{J}_\Psi$, composed of the partial derivatives $h_l^k \doteq \frac{\partial z_j^k}{\partial z_i^l}$, associates coordinate differentials from the sphere points i to the sphere points j:

$$\begin{bmatrix} dz_j^1 \\ dz_j^2 \\ dz_j^3 \end{bmatrix} = \begin{bmatrix} h_1^1 & h_2^1 & h_3^1 \\ h_1^2 & h_2^2 & h_3^2 \\ h_1^3 & h_2^3 & h_3^3 \end{bmatrix} \begin{bmatrix} dz_i^1 \\ dz_i^2 \\ dz_i^3 \end{bmatrix} \tag{25}$$

We will translate this Jacobian to the Jacobian that acts between image coordinates x_j^k and x_i^l, $k, l \in \{1,2\}$. The condition expressing that two nearby points are constrained to a sphere can be written as

$$\begin{aligned} \left(z^1 + dz^1\right)^2 + \left(z^2 + dz^2\right)^2 + \left(z^1 + dz^3\right)^2 = \\ \left(z^1\right)^2 + \left(z^2\right)^2 + \left(z^3\right)^2, \end{aligned} \tag{26}$$

hence

$$z^1 dz^1 + z^2 dz^2 + z^3 dz^3 = 0\,. \tag{27}$$

From (27), the third differential is

$$dz^3 = -\left(\frac{z^1}{z^3} dz^1 + \frac{z^2}{z^3} dz^2\right)\,. \tag{28}$$

This differential constraint reduces the DOF of the Jacobian in (25) by one. Only two lines remain linearly independent. Choosing the first two lines and replacing dz_i^3 by the right hand side of (28), the equations between the coordinate differentials become

$$\begin{bmatrix} dz_j^1 \\ dz_j^2 \end{bmatrix} = \begin{bmatrix} h_1^1 - \frac{z_i^1}{z_i^3} h_3^1 & h_2^1 - \frac{z_i^2}{z_i^3} h_3^1 \\ h_1^2 - \frac{z_i^1}{z_i^3} h_3^2 & h_2^2 - \frac{z_i^2}{z_i^3} h_3^2 \end{bmatrix} \begin{bmatrix} dz_i^1 \\ dz_i^2 \end{bmatrix}\,. \tag{29}$$

According to (19), image points x^l, $l \in \{1,2\}$ and sphere points z^k, $k \in \{1,2\}$ are related by the bijective mapping Φ on the whole domain of estimation. Therefore the differentials are related by

$$\begin{bmatrix} dz^1 \\ dz^2 \end{bmatrix} = \begin{bmatrix} \frac{\partial z^1}{\partial x^1} & \frac{\partial z^1}{\partial x^2} \\ \frac{\partial z^2}{\partial x^1} & \frac{\partial z^2}{\partial x^2} \end{bmatrix} \begin{bmatrix} dx^1 \\ dx^2 \end{bmatrix},$$

hence the Jacobian that maps image differentials $d\mathbf{x}_j = \mathbf{J}_{ij} \cdot d\mathbf{x}_j$ is as follows:

$$[\mathbf{J}_{ij}] = \begin{bmatrix} \frac{\partial \Phi_j^1}{\partial x_j^1} & \frac{\partial \Phi_j^1}{\partial x_j^2} \\ \frac{\partial \Phi_j^2}{\partial x_j^1} & \frac{\partial \Phi_j^2}{\partial x_j^2} \end{bmatrix}^{-1} \begin{bmatrix} h_1^1 - \frac{\Phi_i^1}{\Phi_i^3} h_3^1 & h_2^1 - \frac{\Phi_i^2}{\Phi_i^3} h_3^1 \\ h_1^2 - \frac{\Phi_i^1}{\Phi_i^3} h_3^2 & h_2^2 - \frac{\Phi_i^2}{\Phi_i^3} h_3^2 \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial \Phi_i^1}{\partial x_i^1} & \frac{\partial \Phi_i^1}{\partial x_i^2} \\ \frac{\partial \Phi_i^2}{\partial x_i^1} & \frac{\partial \Phi_i^2}{\partial x_i^2} \end{bmatrix} \tag{30}$$

Like the coordinate gradients, Eq.(30) contains only the components of unit spheres' local basis vectors $\frac{\partial \Phi_i}{\partial x_i^k}$ $k \in \{1,2\}$ and $\frac{\partial \Phi_j}{\partial x_j^l}$ $l \in \{1,2\}$. Since both cameras are calibrated, Φ_i and Φ_j are known. Furthermore, the homography $\mathbf{H}$ acting between the (spherical) regions $\mathcal{D}$ and $\mathcal{F}$ corresponding to the scene plane π has been computed, Ψ is also know, hence $\mathbf{J}_{ij}$ is fully determined.

In summary, given a pair of corresponding regions F and D in a pair of calibrated omnidirectional cameras with known projection functions Φ_i, Φ_j, the 3D scene plane π can be reconstructed through the following steps:

1) Estimate the homography $\mathbf{H}$ acting between the corresponding spherical regions $\mathcal{F}$ and $\mathcal{D}$ (using *e.g.* [23]), which gives Ψ.
2) Estimate the relative pose $(\mathbf{R}, \mathbf{t})$ between the cameras. Given $\mathbf{H}$, this can be done by a standard homography factorization method, *e.g.* [6].
3) Compute the normal $\mathbf{n}$ of π using the direct formula (17), and then d by a standard method based on (3) [11].

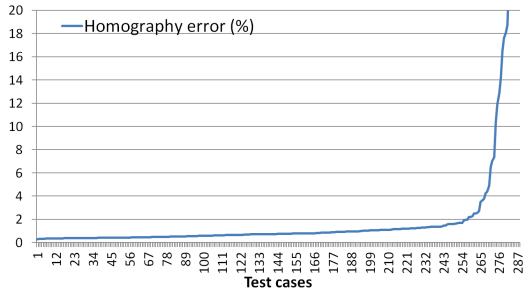

Fig. 3. Homography error for the synthetic datasets (test cases sorted on the x axis).

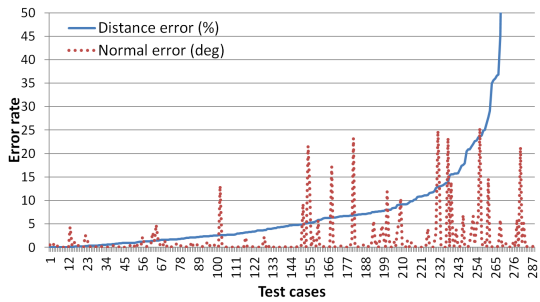

Fig. 4. Distance error and normal error plot for the synthetic datasets (test cases sorted on the x axis based on distance error, normal error values are scaled with the factor of 0.3 for better visualization).

IV. Experimental Results

The proposed method was tested on 3 datasets, each having approximately 100 image pairs. Images of 24 different shapes were used as scene planes and a pair of virtual omnidirectional cameras with random pose were used to generate the omni image pairs. Assuming that a 800×800 pixels scene corresponds to a 5×5m patch, we positioned the virtual cameras at distances from the [45cm-55cm], [100cm-200cm], and [200cm-500cm] intervals respectively, resulting in 3 datasets with different camera base distances. The first step of our algorithm is estimating a homography between the omnidirectional cameras. For this purpose, we use the correspondence-less method proposed in [23]. For a detailed evaluation of the method, see [23]. For reference, we show the homography error on our synthetic dataset in terms of the percentage of non overlapping area sorted in increasing order in Fig. 3. The produced homographies has less than 2% error for about 256 examples. This is important as it directly affects the reconstruction accuracy of our method.

Once the planar homography between the corresponding region pair is estimated, we can compute the 3D surface normal and distance using the proposed closed form formula. Sample 3D reconstructions for synthetic data is shown in Fig. 5. The red surface is the ground truth surface and the green one is the recovered surface. Fig. 4 shows the error plots for the whole synthetic dataset. It is clear that distance error plot runs together with the normal error, hence our method provides reliable reconstructions for most test cases, giving low error rates for both surface parameters.

It is worth mentioning that the reconstruction algorithm's runtime is only 8ms running in Matlab on an Intel i7 3.4 GHz

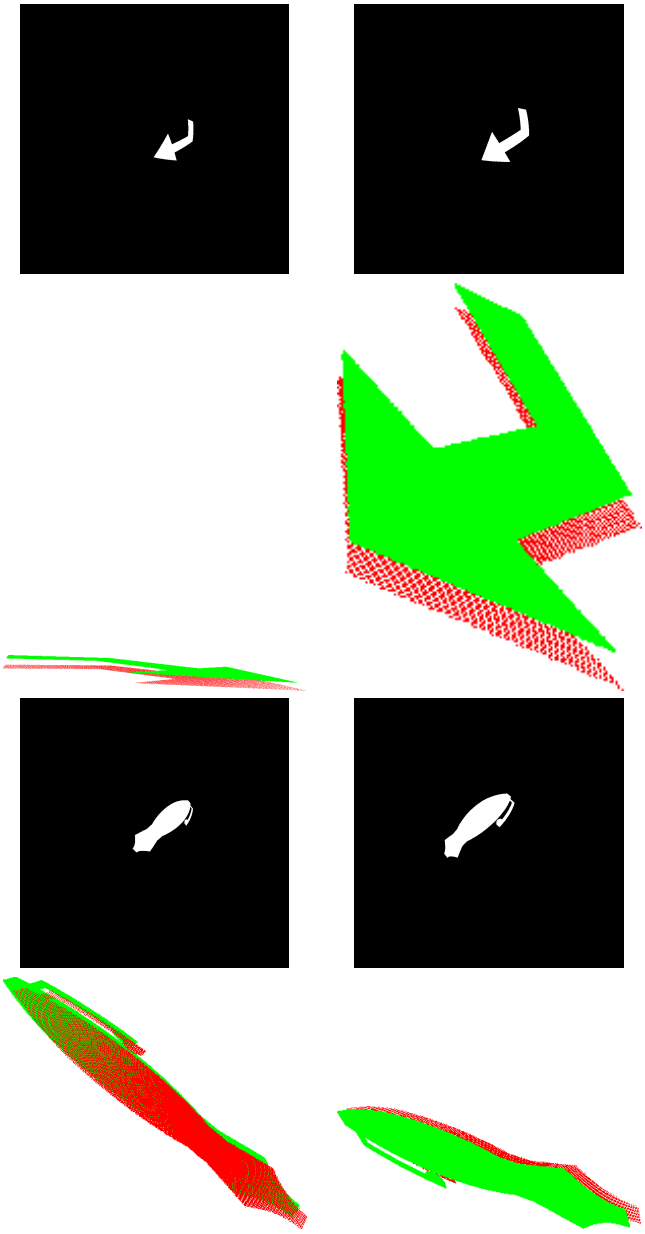

Fig. 5. Reconstruction results from a pair of synthetic omni images (red: reconstructed, green: original 3D planar patch

CPU with 8GB memory. This means it can reach real time speed due to the closed form solution adopted.

A. Comparison with a classical solution

We have performed an experimental comparison of our method with a well known classical plane from homography method described by Hartley and Zisserman [11] (the Matlab code is available from http://www.robots.ox.ac.uk/~vgg/hzbook/code/codevgg_plane_from_2P_H.m) and quantitatively demonstrated the performance of our method with respect to that algorithm. The purpose of this experiment is to compare our direct method derived via differential geometric considerations with a classical direct methods derived via projective geometric considerations, as a basis. Results show that our method is significantly better in determining the

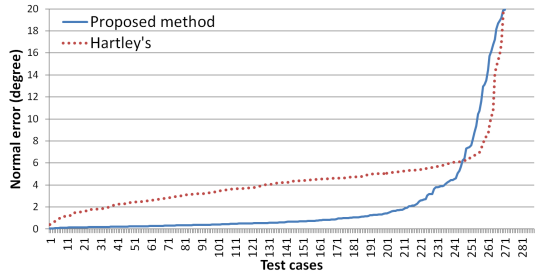

Fig. 6. Comparative normal error plot on our synthetic dataset with the method from [11] (test cases sorted independently for the two methods)

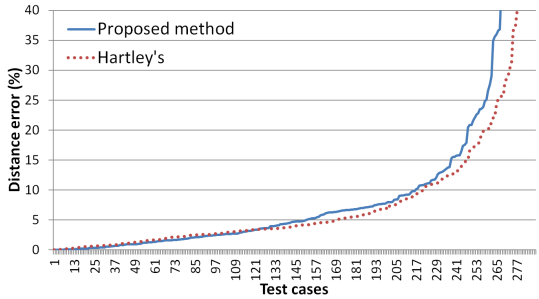

Fig. 7. Comparative distance error plot on our synthetic dataset with the method from [11] (test cases sorted independently for the two methods).

correct normal vector. The error shown in Fig. 6 is computed as the angle in degrees between the calculated and the ground truth normal vectors: mean value of our method was only 0.66^o, while the classical plane from homography method produced 4.32^o error on average. We remark that an error above 5 degrees can be considered a completely wrong result. The relative distance error of the reconstructed plane is shown in Fig. 7. On these plots we can see that the precision of the two methods is almost identical, because both approaches uses a similar way to compute d.

B. Robustness

As we mentioned before, the precision of the estimated homography is crucial for 3D reconstruction. As we can see in Fig. 8 the distance error of the reconstruction is low, until the homography error is below $2-3\%$ but than with bigger homography error it increases exponentially. We can observe the same behavior in the normal vector calculation as shown in Fig. 9.

TABLE I. NORMAL ERROR(DEG) W.R.T. ROTATION ERROR IN DIFFERENT AXIS

Noise(deg)	0	0.5	1	2	4
x	0.55	0.85	1.46	1.89	4.14
y	0.55	0.78	1.21	1.80	3.36
z	0.55	1.23	1.66	3.09	5.59

TABLE II. DISTANCE ERROR(%) W.R.T. ROTATION ERROR IN DIFFERENT AXIS

Noise(deg)	0	0.5	1	2	4
x	2.59	2.71	4.56	4.92	7.71
y	2.59	2.73	2.98	3.01	3.36
z	2.59	2.94	3.11	3.36	4.67

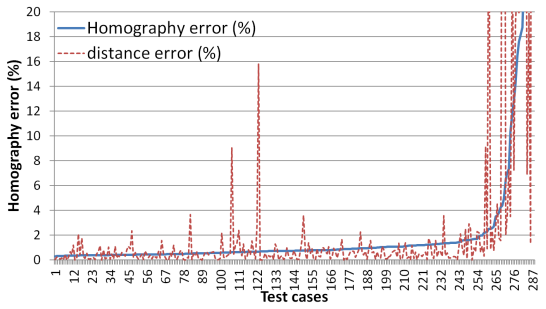

Fig. 8. Distance error rates (scaled with a factor of 0.1 for better visualization) corresponding to the homography error (test cases sorted by the homography error)

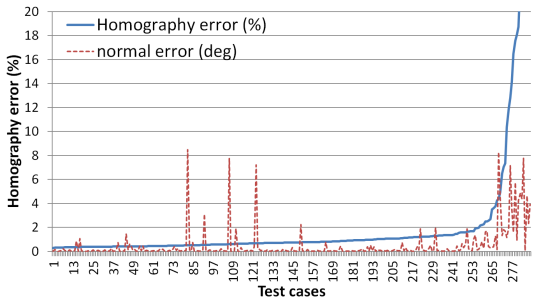

Fig. 9. Normal error rates (scaled with a factor of 0.1 for better visualization) corresponding to the homography error (test cases sorted by the homography error)

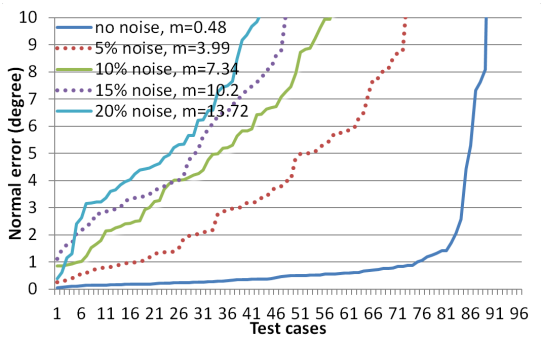

Fig. 10. Normal errors for the noisy omni image datasets (test cases sorted independently, m is the median of errors)

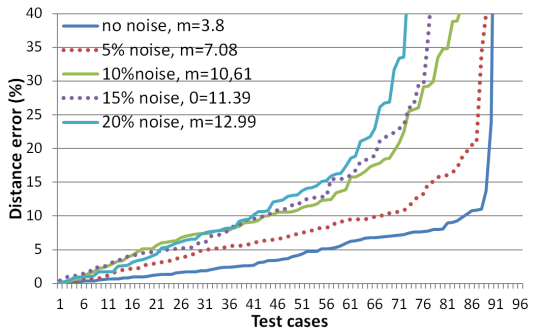

Fig. 11. Distance errors for the noisy omni image datasets (test cases sorted independently, m is the median of errors)

TABLE III. DISTANCE ERROR(%) W.R.T. TRANSLATION ERROR

Noise(%)	0	2	5	10	15
	2.59	3.24	5.41	8.73	14.97

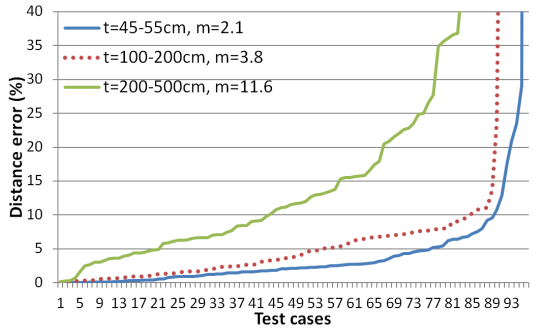

Fig. 12. Distance error plots w.r.t. different baselines (test cases sorted independently, m is the median of errors).

The accuracy of the proposed method depends not only on the quality of the homography estimation, but also on the determined camera pose parameters. Obviously, normal estimation is only affected by the rotation matrix, while distance calculation depends on both rotation and translation. To characterize the robustness of our method against errors in these parameters, we added various percent of noise to the original values and quantitatively evaluated the reconstruction error on our synthetic dataset (see Fig. 10 and Fig. 11). Table I and Table II show that both distance and normal estimation are sensitive to rotation errors in the camera pose, being robust up to 2 degree of rotation error, and distance estimation can tolerate up to 5% translation error as well (see Table III). Normal estimation is more sensitive to rotation error around the z axis, while distance errors increase more with rotation errors around the x axis.

Baseline is another important parameter of 3D reconstruction. Three different datasets (as described at the beginning of this section) were used to test the effect of short, medium and large baselines on reconstruction precision. Fig. 12 shows the distance error while Fig. 13 shows the normal error with respect to each baseline. Of course, shorter baseline has higher error rate, which is a well known fact for stereo reconstruction. However, homography errors are smaller in case of short and medium base distances (see Fig. 14), hence

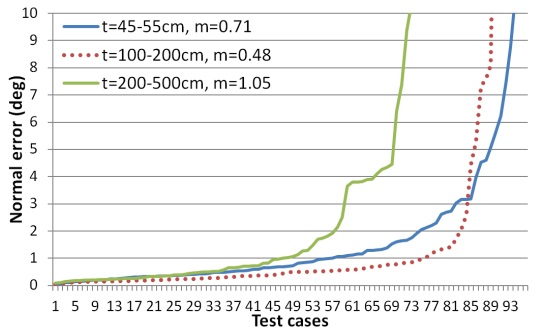

Fig. 13. Normal error plots w.r.t. different baselines (test cases sorted independently, m is the median of errors).

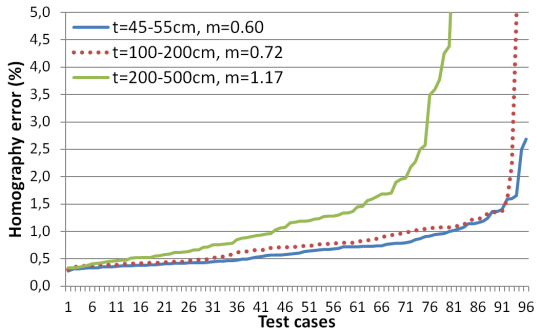

Fig. 14. Homography error w.r.t. different baselines (test cases sorted independently, m is the median of errors).

overall reconstruction performence is better for these datasets.

V. Conclusion

We proposed an efficient 3D reconstruction method, which allows the reconstruction of complete planar surface patches from a homography map between corresponding image regions and calibrated omnidirectional cameras. The theoretical foundation relies on variational calculus, which leads to a closed form solution for the surface normal, while relative pose and distance can be computed from the homography using classical methods. Being a closed form solution, our reconstruction algorithm runs in real-time which can be particularly useful for mobile and embedded vision systems. Quantitative experiments on a large synthetic dataset confirm the superior performance w.r.t. a classical plane reconstruction algorithm.

Acknowledgment

This research was partially supported by the European Union and the State of Hungary, co-financed by the European Social Fund through projects TAMOP-4.2.4.A/2-11-1-2012-0001 National Excellence Program and FuturICT.hu (grant no.: TAMOP-4.2.2.C-11/1/KONV-2012-0013).

References

[1] M. Habbecke and L. Kobbelt, "Iterative multi-view plane fitting," in *In VMV06*, 2006, pp. 73–80.

[2] ——, "A surface-growing approach to multi-view stereo reconstruction," *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pp. 1–8, 2007.

[3] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multi-view stereopsis," in *CVPR*, 2007, pp. 1362–1376.

[4] S. Sinha, D. Steedly, and R. Szeliski, "Piecewise planar stereo for image-based rendering," *Computer Vision, 2009 IEEE 12th International Conference on*, pp. 1881–1888, 2009.

[5] F. Fraundorfer, K. Schindler, and H. Bischof, "Piecewise planar scene reconstruction from sparse correspondences," *Image Vision Comput.*, vol. 24, no. 4, pp. 395–406, Apr. 2006.

[6] P. Sturm, "Algorithms for plane-based pose estimation," in *Proceedings of International Conference on Computer Vision and Pattern Recognition*, vol. 1, Jun. 2000, pp. 706–711.

[7] C. Mei, S. Benhimane, E. Malis, and P. Rives, "Efficient homography-based tracking and 3-D reconstruction for single-viewpoint sensors," *Robotics, IEEE Transactions on*, vol. 24, no. 6, pp. 1352–1364, Dec. 2008.

[8] G. Caron, r. Marchand, and E. M. Mouaddib, "Tracking planes in omnidirectional stereovision." in *ICRA*. IEEE, 2011, pp. 6306–6311.

[9] A. Makadia, C. Geyer, and K. Daniilidis, "Correspondence-free structure from motion," *International Journal of Computer Vision*, vol. 75, no. 3, pp. 311–327, Dec. 2007. [Online]. Available: http://dx.doi.org/10.1007/s11263-007-0035-2

[10] O. Saurer, F. Fraundorfer, and M. Pollefeys, "Homography based visual odometry with known vertical direction and weak Manhattan world assumption," in *IEEE/IROS Workshop on Visual Control of Mobile Robots (ViCoMoR)*, 2012.

[11] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, ISBN: 0521540518, 2004.

[12] S. K. Nayar, "Catadioptric omnidirectional camera," in *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, ser. CVPR '97. Washington, USA: IEEE Computer Society, 1997, pp. 482–. [Online]. Available: http://dl.acm.org/citation.cfm?id=794189.794460

[13] S. Baker and S. K. Nayar, "A theory of single-viewpoint catadioptric image formation," *International Journal of Computer Vision*, vol. 35, no. 2, pp. 175–196, 1999.

[14] C. Geyer and K. Daniilidis, "A unifying theory for central panoramic systems," in *European Conference on Computer Vision (ECCV)*, 2000, pp. 445–462.

[15] B. Mičušík and T. Pajdla, "Para-catadioptric camera auto-calibration from epipolar geometry," in *Proc. of the Asian Conference on Computer Vision (ACCV)*, K.-S. Hong and Z. Zhang, Eds., vol. 2. Seoul, Korea South: Asian Federation of Computer Vision Societies, January 2004, pp. 748–753.

[16] D. Scaramuzza, A. Martinelli, and R. Siegwart, "A toolbox for easily calibrating omnidirectional cameras." in *IEEE/RSJ International Conference on Intelligent Robots*. Bejing: IEEE, October 9–15 2006, pp. 5695–5701.

[17] L. Puig and J. J. Guerrero, *Omnidirectional Vision Systems: Calibration, Feature Extraction and 3D Information*. Springer, 2013.

[18] J. Kannala and S. S. Brandt, "A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 8, pp. 1335–1340, 2006.

[19] C. Mei and P. Rives, "Single view point omnidirectional camera calibration from planar grids," in *IEEE International Conference on Robotics and Automation (ICRA)*, Roma, Italy, April 2007.

[20] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[21] L. Puig and J. J. Guerrero, "Scale space for central catadioptric systems: Towards a generic camera feature extractor," in *Proceedings of International Conference on Computer Vision*. IEEE, 2011, pp. 1599–1606.

[22] T. Svoboda and T. Pajdla, "Epipolar geometry for central catadioptric cameras," *International Journal of Computer Vision*, vol. 49, no. 1, pp. 23–37, 2002.

[23] R. Frohlich, L. Tamas, and Z. Kato, "Homography estimation between omnidirectional cameras without point correspondences," in *Proceedings of ICRA Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras*, IEEE. Hong Kong: IEEE, Jun. 2014.

[24] D. Scaramuzza, A. Martinelli, and R. Siegwart, "A flexible technique for accurate omnidirectional camera calibration and structure from motion," in *Proceedings of the Fourth IEEE International Conference on Computer Vision Systems*, ser. ICVS-06. Washington, USA: IEEE Computer Society, 2006, pp. 45–51.

[25] F. Devernay and O. Faugeras, "Computing differential properties of 3-D shapes from stereoscopic images without 3-D models," in *Proceedings of International Conference on Computer Vision and Pattern Recognition*, Jun. 1994, pp. 208–213.

[26] D. G. Jones and J. Malik, "Determining three-dimensional shape from orientation and spatial frequency disparities," in *Proceedings of European Conference on Computer Vision*, ser. Lecture Notes in Computer Science, G. Sandini, Ed., vol. 588. Springer, 1992, pp. 661–669.

[27] J. Molnár and D. Chetverikov, "Quadratic transformation for planar mapping of implicit surfaces," *Journal of Mathematical Imaging and Vision*, vol. 48, no. 1, pp. 176–184, 2014.