SCHOOL OF ENGINEERING - STI
ELECTRICAL ENGINEERING INSTITUTE
SIGNAL PROCESSING LABORATORY

Luigi Bagnato

EPFL - FSTI - IEL - LTS
Station 11
Switzerland-1015 LAUSANNE

Phone: *+41 21 69 36874*
Fax: *+41 21 69 37600*
e-mail: `luigi.bagnato@epfl.ch`

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

A VARIATIONAL FRAMEWORK FOR STRUCTURE FROM MOTION IN OMNIDIRECTIONAL IMAGE SEQUENCES

Luigi Bagnato, Pierre Vandergheynst, Pascal Frossard

Ecole Polytechnique Fédérale de Lausanne (EPFL)
Signal Processing Laboratory

Technical Report LTS-2009-013

November 17, 2009

Part of this work has been submitted to IEEE Transaction on Image Processing

A Variational Framework for Structure from Motion in Omnidirectional Image Sequences

Luigi Bagnato, Pierre Vandergheynst, Pascal Frossard

Abstract

We address the problem of depth and ego-motion estimation from omnidirectional images. We propose a correspondence-free structure from motion problem for images mapped on the 2-sphere. A novel graph-based variational framework is proposed for depth estimation. The problem is cast into a $TV - L^1$ optimization problem that is solved by fast graph-based optimization techniques. The ego-motion is then estimated directly from the depth information without computation of the optical flow. Both problems are addressed jointly in an iterative algorithm that alternates between depth and ego-motion estimation for fast computation of the 3D information. Experimental results demonstrate the effective performance of the proposed algorithm for 3D reconstruction from synthetic and natural omnidirectional images.

Index Terms

Optical Flow, Manifold, Ego-Motion, Depth Estimation

I. Introduction

RECENTLY, omnidirectional imagers such as catadioptric cameras, have sparked tremendous interest in image processing and computer vision. These sensors are particularly attractive due to their (nearly) full field of view. The visual information coming from a sequence of omnidirectional images can be used to perform a 3D reconstruction of a scene. This type of problem is usually referred to as *Structure from Motion* [1] in the literature. Let us imagine a monocular observer that moves in a rigid unknown world, then the structure from motion problem consists in estimating the 3D rigid self-motion parameters, i.e., rotation and direction of translation, and the structure of the scene that is usually represented as a depth map with respect to the observer position. Structure from motion has attracted considerable attention in the research community over the years since it has direct applications in diverse applications such as autonomous navigation, mixed reality, or 3D video.

In this paper we introduce a novel structure from motion (SFM) framework for omnidirectional images mapped on the 2-sphere, which permits to unify various models of single effective viewpoint cameras. Our SFM algorithm uses only differential motion between two consecutive frames of a video sequence through brightness derivatives and does not attempt to establish correspondences between the images. We propose a novel variational framework to solve the ill-posed depth estimation problem on the 2-sphere. Variational techniques are among the most successful approaches to solve under-determined inverse problems and efficient algorithms have been proposed recently so that their use becomes appealing [2].

We show that it is possible to extend very efficient variational approaches, while naturally handling the geometry of omnidirectional images. We embed a discrete image in a weighted graph whose connections are given by the topology of the manifold and the geodesic distances between connected pixels. We then cast our depth estimation problem as a $TV - L^1$ optimization problem, and we solve the resulting variational problem with fast graph-based optimization techniques similar to [3], [4], [5]. To the best of our knowledge, this is the first time that graph-based variational techniques are applied to obtain a dense depth map from omnidirectional video sequences.

We estimate the 3D motion parameters related to the motion of the camera using the computed depth map. We use a direct approach to estimate the ego-motion, formulating a simple least square optimization problem, which advantageously permits to avoid the computation of the optical flow field.

We finally combine both solutions in an iterative algorithm that jointly estimates depth and ego-motion. Experimental results with synthetic spherical images and natural images from a catadioptric sensor confirm the validity of our approach for 3D reconstruction. The proposed algorithm hence provides an efficient and low-complexity solution to the SFM problem.

The depth and ego-motion estimation problems have been quite widely studied in the last couple of decades and we describe here the most relevant papers that present correspondence-free techniques. Most of the literature in depth estimation is dedicated to stereo depth estimation [6]. In the stereo depth estimation problem cameras are usually separated by a large distance in order to efficiently capture the geometry of the scene. Registration techniques are often used to find a disparity map between the two image views, and the disparity is eventually translated into a depth map. In our problem, we rather assume that the displacement between two consecutive frames in the sequence is small. Correspondence-free depth estimation has been studied in the case of omnidirectional images in [7]. Our method, based on a variational framework, is however expected to provide a higher robustness to quantization errors, noise or illumination gradients.

Ego-motion estimation approaches usually proceed by first estimating the image displacement field, the so-called optical flow. The optical flow field can be related to the global motion parameters by a mapping that depends on the specific imaging surface of the camera. The mapping typically defines the space of solutions for the motion parameters, and specific techniques can eventually be used to obtain an estimate of the ego-motion [8], [9], [10], [11]. Most techniques reveal sensitivity to noisy estimation of the optical flow. The optical flow estimation is a highly ill-posed inverse problem that needs some sort of regularization in order to obtain displacement fields that are physically meaningful; a common approach is to impose a smoothness constraint on the field [12], [13]. In order to avoid the computation of the optical flow, one can use the so-called ”direct approach” where image derivatives are directly related to the motion parameters. Without any assumption on the scene, the search space of the ego-motion parameters is limited by the *depth positivity constraint*. For example, the works in [14], [15] estimate the motion parameters that result into the smallest amount of negative values in the depth map. Several algorithms originally proposed for planar cameras have later been adapted to cope with the geometrical distortion introduced by omnidirectional imaging systems. For example, an omnidirectional ego-motion algorithm has been presented by Gluckman in [16], where the optical flow field is estimated in the catadioptric image plane and then back-projected onto a spherical surface. In our work, we directly estimate the ego-motion from the depth map by solving a simple least square optimization problem.

The rest of the paper is structured as follows. We describe in Section II the framework used in this paper for motion and depth estimation and the corresponding discrete operators in graph-based representations. The variational depth estimation problem is presented in Section III, and the ego-motion estimation is discussed in Section IV. Section V presents the joint depth and ego-motion estimation algorithm, while Section VI presents experiments of 3D reconstruction from synthetic and natural omnidirectional image sequences.

II. Framework Description

A. *Motion in spherical images*

In this section, we introduce the framework and the notation that will be used in the paper. We derive the equations that relate global motion parameters and depth map to the brightness derivatives on the sphere. Finally, we show how we embed our spherical framework on a weighted graph structure and define differential operators in this representation.

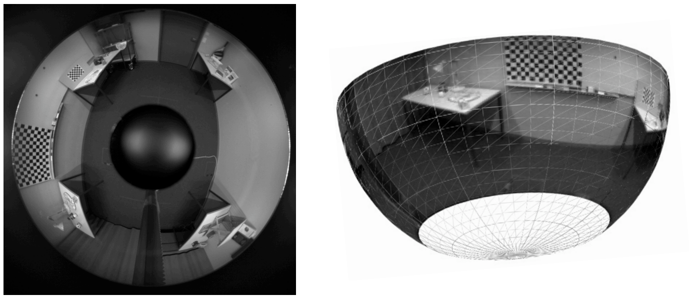

Fig. 1. Left: the original catadioptric image. Right: projection on the sphere

We choose to work on the 2-sphere S^2, which is a natural spatial domain to perform processing of omnidirectional images as shown in [17] and references therein. For example, catadioptric camera systems with a single effective viewpoint permit a one-to-one mapping of the catadioptric plane onto a sphere via inverse stereographic projection [18]. The centre of that sphere is co-located with the focal point of the parabolic mirror and each direction represents a light ray incident to that point. We assume then that a pre-processing step transforms the original omnidirectional images into spherical ones as depicted in Fig. 1.

The starting point of our analysis is the *brightness consistency equation*, which assumes that pixel intensity values do not change during motion between successive frames. Let us denote $I(t, \mathbf{y})$ an image sequence, where t is time and $\mathbf{y} = (y^1, y^2, y^3)$ describes a spatial position in 3-dimensional space. If we consider only two consecutive frames in the image sequence, we can drop the time variable t an use I_0 and I_1 to refer to the first and the second frame respectively. The brightness consistency assumption then reads: $I_0(\mathbf{y}) - I_1(\mathbf{y} + \mathbf{u}) = 0$ where $\mathbf{u}$ is the displacement field between the frames. We can linearize the brightness consistency constraint around $\mathbf{y} + \mathbf{u}_0$ as:

$$I_1(\mathbf{y} + \mathbf{u}_0) + (\nabla I_1(\mathbf{y} + \mathbf{u}_0))^T(\mathbf{u} - \mathbf{u}_0) - I_0(\mathbf{y}) = 0, \tag{1}$$

with an obvious abuse of notation for the equality. This equation relates the motion field $\mathbf{u}$ (also known as optical flow field) to the (spatial and temporal) image derivatives. It is probably worth stressing that, for this simple linear model to hold, we assume that the displacement $\mathbf{u} - \mathbf{u}_0$ between the two scene views I_0 and I_1 is sufficiently small.

When data live on S^2 we can express the gradient operator ∇ from Eq. (1) in spherical coordinates as :

$$\nabla I(\phi, \theta) = \frac{1}{\sin\theta}\partial_\phi I(\phi, \theta)\hat{\phi} + \partial_\theta I(\phi, \theta)\hat{\theta}, \tag{2}$$

where $\theta \in [0, \pi]$ is the colatitude angle, $\phi \in [0, 2\pi[$ is the azimuthal angle and $\hat{\phi}, \hat{\theta}$ are the unit vectors on the tangent plane corresponding to infinitesimal displacements in ϕ and θ respectively (see Fig. 2). Note also that by construction the optical flow field $\mathbf{u}$ is defined on the tangent bundle $TS = \bigcup_{\omega \in S^2} T_\omega S^2$, i.e. $\mathbf{u} : S^2 \subset \mathbb{R}^3 \rightarrow TS$.

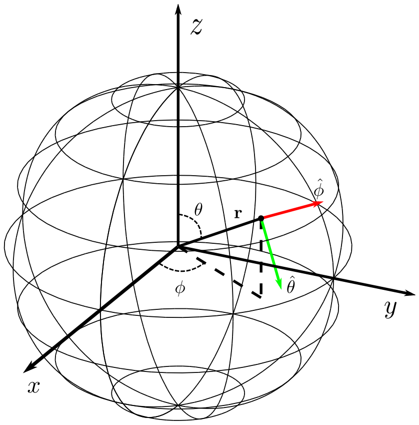

Fig. 2. The representation and coordinate on the 2-sphere S^2.

B. Global motion and optical flow

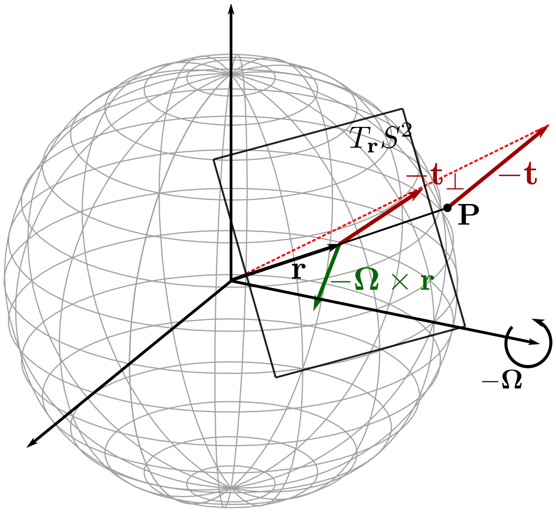

Fig. 3. The sphere and the motion parameters.

Under the assumption that the motion is slow between frames, we have derived above a linear relationship between the apparent motion $\mathbf{u}$ on the spherical retina and the brightness derivatives. If the camera undergoes rigid translation $\mathbf{t}$ and rotation around the axis $\mathbf{\Omega}$, then we can derive a geometrical constraint between $\mathbf{u}$ and the parameters of the 3D motion of the camera. Let us consider a point $\mathbf{P}$ in the scene, with respect to a coordinate system fixed at the center of the camera. We can express $\mathbf{P}$ as: $\mathbf{P} = D(\mathbf{r})\mathbf{r}$ where $\mathbf{r}$ is the unit vector giving the direction to $\mathbf{P}$ and $D(\mathbf{r})$ is the distance of the scene point from the center of the camera. During camera motion the scene point moves with respect to the camera by a quantity given as :

$$\delta\mathbf{P} = -\mathbf{t} - \mathbf{\Omega} \times \mathbf{r}. \tag{3}$$

This motion is illustrated in Fig. 3. We can now build the geometric relationship that relates the motion field $\mathbf{u}$ to the global motion parameters $\mathbf{t}$ and $\mathbf{\Omega}$. It reads

$$\mathbf{u}(\mathbf{r}) = -\frac{\mathbf{t}_\perp}{D(\mathbf{r})} - \mathbf{\Omega} \times \mathbf{r} = -Z(\mathbf{r})\mathbf{t}_\perp - \mathbf{\Omega} \times \mathbf{r}, \tag{4}$$

where $\mathbf{t}_\perp(\mathbf{r}) = (\mathbf{t} \cdot \mathbf{r})\mathbf{r} - \mathbf{t}$ is the projection of the global vector $\mathbf{t}$ onto the tangent plane of the spherical retina. The function $Z(\mathbf{r})$ is defined as the multiplicative inverse of the distance function $D(\mathbf{r})$. In the following we will refer to Z as the *depth map*. In Eq. (4) we find all the unknowns of our structure from motion problem: the depth map $Z(\mathbf{r})$ describing the structure of the scene and the 3D motion parameters $\mathbf{t}$ and $\mathbf{\Omega}$. The first thing to note is that, due to the multiplication between $Z(\mathbf{r})$ and $\mathbf{t}$ both quantities can be estimated only up to a scale factor. So in the following we will consider that $\mathbf{t}$ has unitary norm.

We can finally combine Eq. (1) and Eq. (4) in a single equation:

$$\begin{aligned} I_1(\mathbf{y} + \mathbf{u}_0) + (\nabla I_1(\mathbf{y} + \mathbf{u}_0))^T(-Z(\mathbf{r})\mathbf{t}_\perp - \\ \mathbf{\Omega} \times \mathbf{r} - \mathbf{u}_0) - I_0(\mathbf{y}) = 0. \end{aligned} \tag{5}$$

Eq. (5) relates image derivatives directly to 3D motion parameters. The equation is not linear in the unknowns and it defines an under-constrained system (i.e. more unknown than equations). We will use this equation as constraint in the optimization problem proposed in the next section.

C. Discrete differential operators on the 2-Sphere

We have developed our previous equations in a continuous spatial domain, but we have to remember that our images are digital. Although the 2-sphere is a simple manifold with constant curvature and a simple topology, a special attention has to be paid to the definition of the differential operators that will be used later in the variational framework.

We assume that the omnidirectional images recorded by the sensor are interpolated onto a spherical equiangular grid : $\{\theta_m = m\pi/M, \phi_n = n2\pi/N\}$, with $M \cdot N$ the total number of samples. In spherical coordinates, a simple discretization of the gradient obtained from finite differences reads:

$$\begin{aligned} &\nabla_\theta f(\theta_{i,j}, \phi_{i,j}) = \frac{f(\theta_{i+1,j}, \phi_{i,j}) - f(\theta_i, \phi_j)}{\Delta\theta}, \\ &\nabla_\phi f(\theta_{i,j}, \phi_{i,j}) = \\ &\qquad \frac{1}{\sin\theta_{i,j}} \left(\frac{f(\theta_{i,j}, \phi_{i,j+1}) - f(\theta_{i,j}, \phi_{i,j})}{\Delta\phi} \right). \end{aligned} \tag{6}$$

The discrete divergence, by analogy with the continuous setting, is defined by $div = -\nabla^*$ (∇^* is the adjoint of ∇). It is then easy to verify that the divergence is given by:

$$\begin{aligned} div\mathbf{p}(\theta_{i,j}, \phi_{i,j}) = &\frac{p^\phi(\theta_{i,j}, \phi_{i,j}) - p^\phi(\theta_{i,j}, \phi_{i,j-1})}{\sin\theta_{i,j}\Delta\phi} + \\ &\frac{\sin\theta_{i,j} p^\theta(\theta_{i,j}, \phi_{i,j}) - \sin\theta_{i,j} p^\theta(\theta_{i-1,j}, \phi_{i,j})}{\sin\theta_{i,j}\Delta\theta}. \end{aligned} \tag{7}$$

Both Eq. (6) and Eq. (7) contain a $(\sin\theta)^{-1}$ term that induces very high values around the poles (i.e., $\theta \simeq 0$ and $\theta \simeq \pi$) and can cause numerical instability. We therefore propose to define discrete differential operators on weighted graphs (i.e., discrete manifold) as a general way to deal with geometry in a coordinate-free fashion.

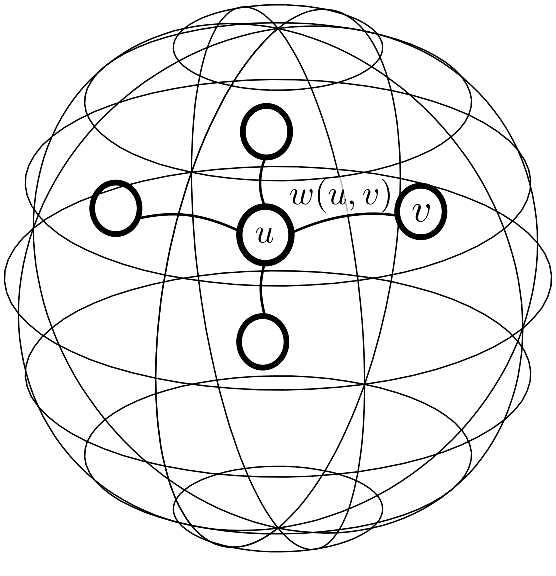

Fig. 4. Embedding of discrete sphere on a graph structure. The pixels u and v in the spherical image represent vertices of the graph, and the edge weight $w(u, v)$ typically captures the geodesic distance between the vertices.

We represent our discretized imaging surface (a sphere in our problem) as a weighted graph, where the vertices represent image pixels and edges define connections between pixels (i.e., the topology of the surface) as represented in Fig. 4. A weighted undirected graph $\Gamma = (V, E, w)$ consists of a set of vertices V, a set of vertices pairs $E \subseteq V \times V$, and a weight function $w : E \mapsto \mathbb{R}$ satisfying $w(u, v) > 0$ and $w(u, v) = w(v, u)$, $\forall(u, v) \in E$. Following Zhou et al [5], we now define the gradient and divergence over Γ as :

$$(\nabla^w f)(u, v) = \sqrt{\frac{w(u,v)}{d(u)}} f(u) - \sqrt{\frac{w(u,v)}{d(v)}} f(v) \tag{8}$$

and

$$(div^w F)(u) = \sum_{u \sim v} \sqrt{\frac{w(u,v)}{d(v)}} \left(F(v,u) - F(u,v) \right), \tag{9}$$

where $u \sim v$ stands for all vertices v connected to u and $d : V \mapsto \mathbb{R}$ is the degree function defined as:

$$d(v) = \sum_{u \sim v} w(u,v). \tag{10}$$

The weight $w(u,v)$ is typically defined as the geodesic distance between the vertices u and v.

The main advantages of the graph-based representation are that the differential operators are directly defined on a discrete domain. They reveal a much more stable behavior than their counterparts from Eq. (6) and Eq. (7). Indeed, it is easy to see that, using a simple 4-connected topology, the factor $w(u,v)/d(u)$ is of order $1/4$ at each vertex and can easily be pre-computed. Hence there is no more source of instability in the numerical scheme. It should finally be noted that this generic framework provides flexibility in the choice of the discrete grid points, whose density can vary locally on the sphere.

III. Variational Depth Estimation

Equipped with the above formalism, we now propose a new variational framework to estimate a depth map from two consecutive frames of an omnidirectional image sequence. We assume at this point that the parameters $\mathbf{t}, \mathbf{\Omega}$ that describe the 3D motion of the camera are known. In addition, we might have an estimate of the optical flow field $\mathbf{u}_0$.

Let us consider again Eq. (5) that relates image derivatives to motion parameters. Since the image gradient ∇I_1 is usually sparse, Eq. (5) does not provide enough information to recover a dense depth map. Hence, we formulate the depth estimation problem as a regularized inverse problem using the L^1 norm to penalize deviation from the brightness constraint and the TV-norm to obtain a regular depth map with sharp transitions.

We build the following error functional:

$$J(Z) = \int_\Omega \psi(\nabla Z) \, \mathsf{d}\Omega + \lambda \int_\Omega |\rho(I_0, I_1, Z)| \, \mathsf{d}\Omega, \tag{11}$$

and we look for the depth map Z that minimizes it. In Eq. (11) the function ρ is the data fidelity term that describes the residual image error after motion compensation:

$$\begin{aligned} &\rho(I_0, I_1, Z) = I_1(\mathbf{y} + \mathbf{u}_0) + \\ &(\nabla I_1(\mathbf{y} + \mathbf{u}_0))^T(-Z(\mathbf{r})\mathbf{t}_\perp - \mathbf{\Omega} \times \mathbf{r} - \mathbf{u}_0) - I_0(\mathbf{y}), \end{aligned} \tag{12}$$

where we use our assumption that $\mathbf{t}, \mathbf{\Omega}$ and $\mathbf{u}_0$ are known. The regularization function ψ is given by:

$$\psi(\nabla Z) = |\nabla Z(\mathbf{r})|. \tag{13}$$

With such a choice of the functional J we define a $TV - L^1$ inverse problem. Several advantages come from this choice. First the $TV - L^1$ model is very efficient in removing noise and robust against illumination changes : it inherits these properties from the Rudin-Osher-Fatemi (ROF) model [19] and the L^1 norm fidelity term ensures robustness to outliers and also non-erosion of edges [20]. Furthermore the TV regularization is a very efficient prior to preserve sharp edges. The total variation model then suits the geometrical features of a real scene structure where the depth map is typically piecewise linear with sharp transitions on objects boundaries.

The functional in Eq. (11) is written in terms of continuous variables, while we actually work with discrete images in practice. Inspired by the continuous formulation, we now propose to solve a similar, though purely discrete, problem. With the graph described in the previous section, we define the local isotropic variation of Z at vertex (pixel) v by :

$$\|\nabla_v^w Z\| = \sqrt{\sum_{u \sim v} \left[\Big(\nabla^w Z \Big)(u,v) \right]^2}. \tag{14}$$

The discrete optimization problem can then be written as :

$$J(Z) = \sum_v \|\nabla_v^w Z\| + \lambda \sum_v |\rho(I_0, I_1, Z)|. \tag{15}$$

The definition of ρ is the same as in Eq. (12), where we however substitute the naive finite difference approximation of the gradient given in Eq. (6). Note that the discrete problem now uses two different discretizations for differential operators on S^2. The reason for this choice will be made clear below.

We now discuss the solution of the depth estimation problem in Eq. 15. Even though the resulting functional J is convex, it poses severe computational difficulties. Following [21], we propose a convex relaxation into a sum of two simpler sub-problems:

$$\begin{aligned} J(Z) = &\sum_v \|\nabla_v^w Z\| + \\ &\frac{1}{2\theta}\sum_u (V(u) - Z(u))^2 + \lambda \sum_u |\rho(I_0, I_1, V)|, \end{aligned} \tag{16}$$

where V is an auxiliary variable that should be as close as possible to Z. If θ is small then V converges to Z and the functional defined in Eq. (16) converges to the one defined in Eq. (15). The minimization must now be performed with respect to both the variables V, Z. Since the functional is convex the solution can be then obtained by an iterative two-step procedure:

1) For Z fixed, solve:

$$\min_V \left\{ \frac{1}{2\theta} \sum_u (V(u) - Z(u))^2 + \lambda |\rho(V(u))| \right\}. \tag{17}$$

2) For V fixed, solve:

$$\min_Z \left\{ \sum_u \|\nabla_u^w Z\| + \frac{1}{2\theta} \sum_u (V(u) - Z(u))^2 \right\}. \tag{18}$$

The minimization in the first step is straightforward : the problem is completely decoupled in all coordinates and the solution can be found in a point-wise manner using this thresholding scheme:

$$V = Z + \begin{cases} \theta\lambda \nabla I_1^T \mathbf{t} & \text{if } \rho(Z) < -\theta\lambda(\nabla I_1^T \mathbf{t})^2 \\ -\theta\lambda \nabla I_1^T \mathbf{t} & \text{if } \rho(Z) > \theta\lambda(\nabla I_1^T \mathbf{t})^2 \\ -\dfrac{\rho(Z)}{\nabla I_1^T \mathbf{t}} & \text{if } |\rho(Z)| \leqslant \theta\lambda(\nabla I_1^T \mathbf{t})^2. \end{cases} \tag{19}$$

The previous result can be easily obtained by writing the Euler-Lagrange condition for Eq. (17)

$$\frac{1}{\theta}(Z - V) + \lambda \nabla I_1^T \mathbf{t} \frac{\rho(V)}{|\rho(V)|} = 0, \tag{20}$$

and then analyzing the three different cases: $\rho(Z) > 0$, $\rho(V) < 0$ and $\rho(V) = 0$. Using the relationship $\rho(V) = \rho(Z) + \nabla I_1^T \mathbf{t}(V - Z)$ we have:

- $\rho > 0$:
  $(Z - V) = \theta\lambda\nabla I_1^T \mathbf{t} \Rightarrow \rho(Z) > \nabla I_1^T \mathbf{t}(Z - V) = \theta\lambda(\nabla I_1^T)^2$
- $\rho < 0$:
  $(Z - V) = -\theta\lambda\nabla I_1^T \mathbf{t} \Rightarrow \rho(Z) < -\nabla I_1^T \mathbf{t}(Z - V) = \theta\lambda(\nabla I_1^T)^2$
- $\rho = 0$:
  $\rho(Z) = -\nabla I_1^T \mathbf{t}(V - Z)$

Notice that this computation relies on evaluating the scalar product $\nabla I_1^T \mathbf{t}$, which can not be evaluated if we use a graph-based gradient since the vector $\mathbf{t}$ is unconstrained (in particular it does not correspond necessarily to an edge of the graph). However, this part of the algorithm is not iterative and the gradient can be pre-computed, therefore avoiding severe numerical instabilities as we move closer to the poles.

The minimization in Eq. (18) corresponds to the total variation image denoising model, for which Chambolle proposed an efficient fixed point algorithm [22]. As most TV denoising algorithms, it is iterative and both gradient and divergence will be computed at each iteration; this is the primary reason for using the graph-based operators in this part of the variational problem. Chambolle's iterations read explicitely:

$$\begin{aligned} Z &= V - \theta div\mathbf{p}, \\ \mathbf{p}^{n+1} &= \frac{\mathbf{p}^n + \tau\nabla(div\mathbf{p}^n - V/\theta)}{1 + \tau|\nabla(div\mathbf{p}^n - V/\theta)|}. \end{aligned} \tag{21}$$

Finally, it should be noted that the algorithm is formally the same whatever discretization is chosen, i.e., the discrete operator can be given either by Eq. (9) or Eq. (7) . Experimental results however show that the graph-based operators unsurprisingly lead to the best performance.

IV. Least Square Ego-Motion Estimation

We discuss in this section a direct approach for the estimation of the ego-motion parameters $\mathbf{t}, \mathbf{\Omega}$ from the depth map Z. We propose a very simple formulation based on least mean square optimization.

When we have an estimate of $Z(\mathbf{r})$ in Eq. (5), we have a set of linear constraints in the motion parameters $\mathbf{t}$,$\mathbf{\Omega}$ that can be written as :

$$Z(\nabla I_1)^T \mathbf{t} + (\mathbf{r} \times (\nabla I_1))^T \mathbf{\Omega} = I_0 - I_1. \tag{22}$$

For each direction in space $\mathbf{r}$ we can rewrite Eq. (22) in a matrix form:

$$A(\mathbf{r})\mathbf{b} = C(\mathbf{r}), \tag{23}$$

where $A(\mathbf{r}) = [(Z(\mathbf{r})\nabla I_1(\mathbf{r}))^T \;\; (\mathbf{r} \times \nabla I_1(\mathbf{r}))^T]$, $C(\mathbf{r}) = I_0(\mathbf{r}) - I_1(\mathbf{r})$ and $\mathbf{b} = [\mathbf{t}; \mathbf{\Omega}]$.

We can formulate the ego-motion estimation problem as follows:

find $\mathbf{b}$ that minimize the following error functional:

$$\mathbf{b}^* = \underset{\mathbf{b}}{\operatorname{argmin}} \sum_{\mathbf{r}} (A(\mathbf{r})\mathbf{b} - C(\mathbf{r}))^2. \tag{24}$$

The solution to this linear least square problem is simply:

$$\mathbf{b} = \frac{\sum_{\mathbf{r}} A^T C}{\sum_{\mathbf{r}} A A^T}. \tag{25}$$

There are several aspects that are important for the existence and the unicity of the solution of the ego-motion estimation problem. First, the images must present enough structure. In other words, the image gradient ∇I_1 should carry enough information on the structure on the scene. In particular, since the gradient only gives information on motion that is perpendicular to image edges, the gradient itself will not help recovering motion parameters if the projection of the motion parameters on the spherical retina is everywhere parallel to the gradient direction. This situation is however highly unlikely for a real scene and a wide field of view camera.

Then, there is a possibility of confusion for certain combinations of the motion parameters. In Eq. (22) we compute the scalar product between the image gradient and the vector $Z(\mathbf{r})\mathbf{t}_\perp + \mathbf{\Omega} \times \mathbf{r}$, i.e., the spherical projection of 3D motion. For a small field of view, $\mathbf{r}$ does not change much and the two terms $Z(\mathbf{r})\mathbf{t}_\perp$ and $\mathbf{\Omega} \times \mathbf{r}$ could be parallel, meaning that we cannot recover them univocally. This happens for example with a rotation around vertical axis and a displacement in the perpendicular direction to both viewing direction and rotation axis. Such a confusion however disappear in the spherical framework, since it is has been shown that a full field of view disambiguate motion.

V. Joint Ego-Motion and Depth Map Estimation

We have described in the previous sections the separate estimation of a dense depth map and the 3D motion parameters. The purpose of this section is to combine the proposed solutions in a dyadic multi-resolution framework.

We jointly solve the depth and ego-motion estimation problems by alternating minimization steps. For each resolution level, we compute a solution to Eq. (5) by performing two minimization steps:

1) We have an estimate $\bar{Z}(\mathbf{r})$ of $Z(\mathbf{r})$ from the previous iterations of the algorithm. By least square minimization from Eq.(25) we can compute an estimate of the motion parameters $\bar{\mathbf{t}}, \bar{\mathbf{\Omega}}$.
2) If we have an estimate of the motion parameters $\bar{\mathbf{t}}, \bar{\mathbf{\Omega}}$ we can find an estimate $\bar{Z}(\mathbf{r})$ of the depth map by solving Eq. 15 using the variational framework described in Sec. III.

Since we perform a coarse to fine approach we only need to assign an initial value to our parameter at the coarsest scale. We choose the following values: $Z_0 = const$ $\mathbf{t}_0 = [0; 0; 0]$ $\mathbf{\Omega}_0 = [0; 0; 0]$. At coarse scale, the approximation that we introduce by flattening the depth map Z is well posed since all edges are smoothed away at low resolution. We also find that the estimation of motion parameters is very accurate at this level, and it is completely independent on the constant value we choose for Z. At the next resolution level i we initialize our algorithm with the estimates of previous resolution level Z_{i-1},$\mathbf{t}_{i-1}$,$\mathbf{\Omega}_{i-1}$. Furthermore we can obtain an estimate of the optical flow $\mathbf{u}_0 = -Z_i(\mathbf{r})\mathbf{t}_{i\perp} - \mathbf{\Omega}_i \times \mathbf{r}$, and use it to warp image I_1, i.e., to estimate $I_1(\mathbf{r} + \mathbf{u}_i)$. The joint depth and ego-motion estimation algorithm is summarized in Algorithm 1.

Similar ideas of alternating minimization steps have been proposed in [23], [24]. In these works, however, the authors assume to have an initial rough estimate of the depth map. On top of that they propose a simple locally constant depth model, so that they can obtain a estimation using the Lukas-Kanade algorithm [25]. In our experiments we show that this model is an oversimplification of the real world, which cannot be applied to scenes with complex structure.

Algorithm 1 Computation of $Z, \mathbf{t}, \mathbf{\Omega}$

```
1) Initialize with $Z_0 = const$ $\mathbf{t}_0 = [0;0;0]$ $\mathbf{\Omega}_0 = [0;0;0]$
2) For each resolution level $i$:
    a) Initialize $Z_i$ $\mathbf{t}_i$ $\mathbf{\Omega}_i$ with solution at previous resolution level.
    b) Estimate optical flow $\mathbf{u}_0 = -Z_i(\mathbf{r})\mathbf{t}_{i\perp} - \mathbf{\Omega}_i \times \mathbf{r}$
    c) Estimate $\mathbf{t}$ and $\mathbf{\Omega}$ in $\mathbf{b}$ using $Z_i$:
           $$\mathbf{b} = \frac{\sum_{\mathbf{r}} A^T C}{\sum_{\mathbf{r}} A A^T}.$$
    d) Estimate $Z$ using the depth estimation algorithm described in Sec. III with the current estimates $\mathbf{t}$ and $\mathbf{\Omega}$.
```

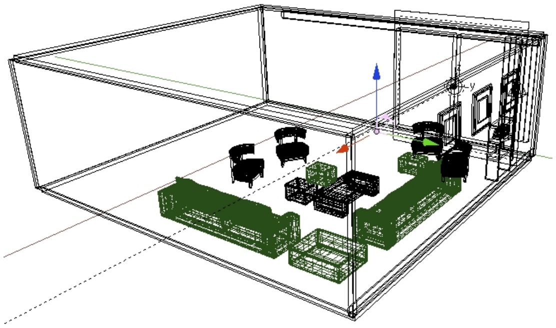

Fig. 5. The 3D model of the scene

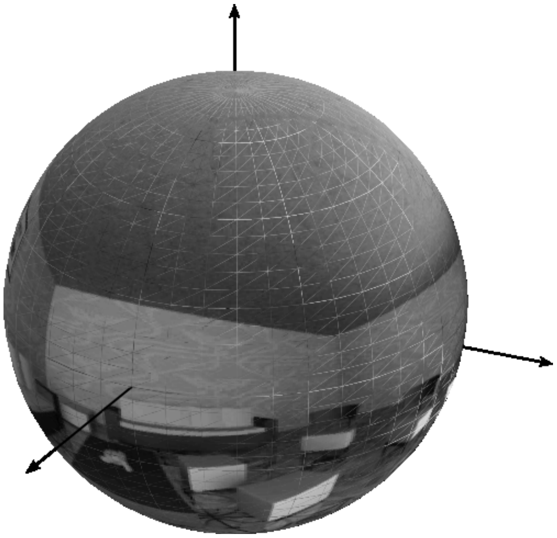

Fig. 6. Synthetic omnidirectional images in spherical representation.

VI. Experimental results

A. Synthetic omnidirectional images

We analyze in this section the performance of the proposed algorithms for two sets of omnidirectional images, namely a synthetic and a natural sequence. The first set of images represents the spherical rendering of the 3D model of a living room shown in Fig. 5. The resulting images are illustrated in Fig. 6.

A planar representation of this image can be obtained by a Mercator projection, as shown in Figure 7. In this image plane, the vertical and horizontal coordinates correspond respectively to the θ and ϕ angles. The images are represented such that the top of the image correspond to the north pole and the bottom to the south pole. We further have the ground truth depth information for this synthetic image set, which permits to evaluate the performance of our depth estimation algorithm. The ground truth depth map is illustrated in Fig. 7.

We first study the influence of the discretization scheme in the variational depth estimation algorithm. As discussed in

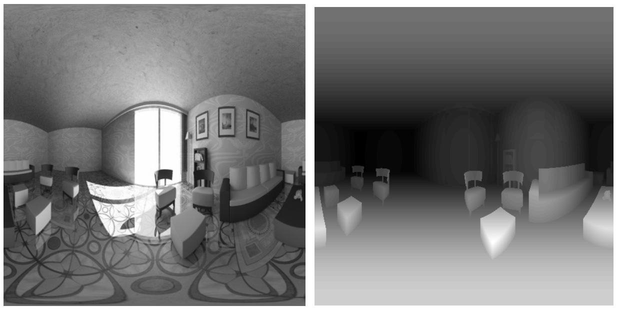

Fig. 7. The synthetic spherical image in a Mercator map (left) and the corresponding depth map ground truth (right).

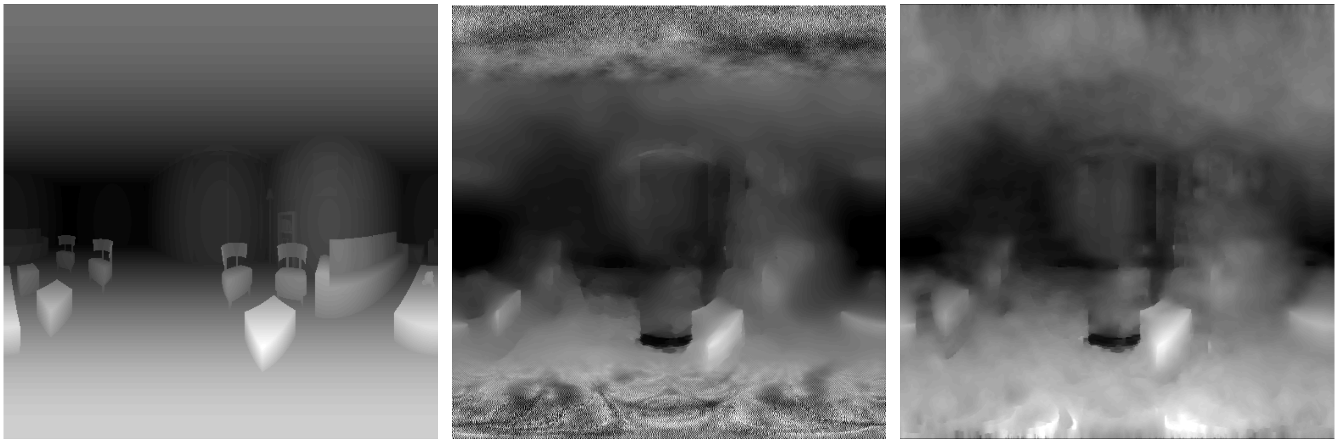

Fig. 8. Depth map estimation with different discrete differential operators. *Left*: ground truth. *Center*: TVL1-naive. *Right*: TVL1-GrH.

Sec. III the TV denoising part of the depth estimation algorithm is extremely sensitive to the choice of the discrete differential operators. We show in Fig. 8 that the use of the differential operators from Eq. (6) and (7) lead to noisy results around the poles. We call the resulting algorithm as *TVL1-naive*. We compare the results of this implementation to those obtained by choosing the graph-based definition of the differential operators from Eq. (8) and (9). The second algorithm, called *TVL1-GrH*, clearly lead to improved performance, especially around the poles where it is much more robust than *TVL1-naive*.

Then, we compare in Fig. 9 the results of the variational depth map estimation algorithm for four different camera motions, namely a pure translation or different combinations of rotation and translation. We compare our results to a local-constant-depth model algorithm (i.e., *LK*) described in [23]. This approach assumes that the depth is constant in regions of the image and tries to find a least square solution to the depth estimation problem. We can observe that the $TV - L^1$ model is much more efficient in preserving edges, so that it becomes possible to distinguish the objects in the 3D scene. The *LK* algorithm has a tendency to smooth the depth information so that objects are hardly visible. This results are confirmed in Table I in terms of mean square error of the depth map reconstruction for several sequences synthetic sequences. It can be seen that the local-costant-depth algorithm is outperformed by the variational depth estimation algorithm with graph-based operators (*TVL1-GrH*), It is also interesting to observe the influence of the choice of the discrete differential operators. The discretization from Eq. (6) and (7) (*TVL1-naive*) clearly leads to the worst results, while the graph-based operators perfom best.

TABLE I
MEAN SQUARE ERROR (MSE) BETWEEN THE ESTIMATED DEPTH MAP AND THE GROUND TRUTH.

	Seq1	Seq2	Seq3	Seq4	Seq5
LK	306.2129	702.6728	413.2414	1601	567.0927
TVL1-naive	1173	27051	26828	28375	27182
TVL1-GrH	269.5355	442.3501	437.8665	1036	445.1818

Finally, we analyze in Table. II the performance of the ego-motion estimation algorithm proposed in Sec. IV. We use the same synthetic sequences as before, and the depth estimation results are used in the least mean square optimization problem for motion parameter estimation. We compare the ego-motion estimation to the true motion parameters, given in terms of translation ($\mathbf{t}$) and rotation ($\mathbf{\Omega}$) parameters. We can see that the ego-motion estimation is quite efficient for all the sequences

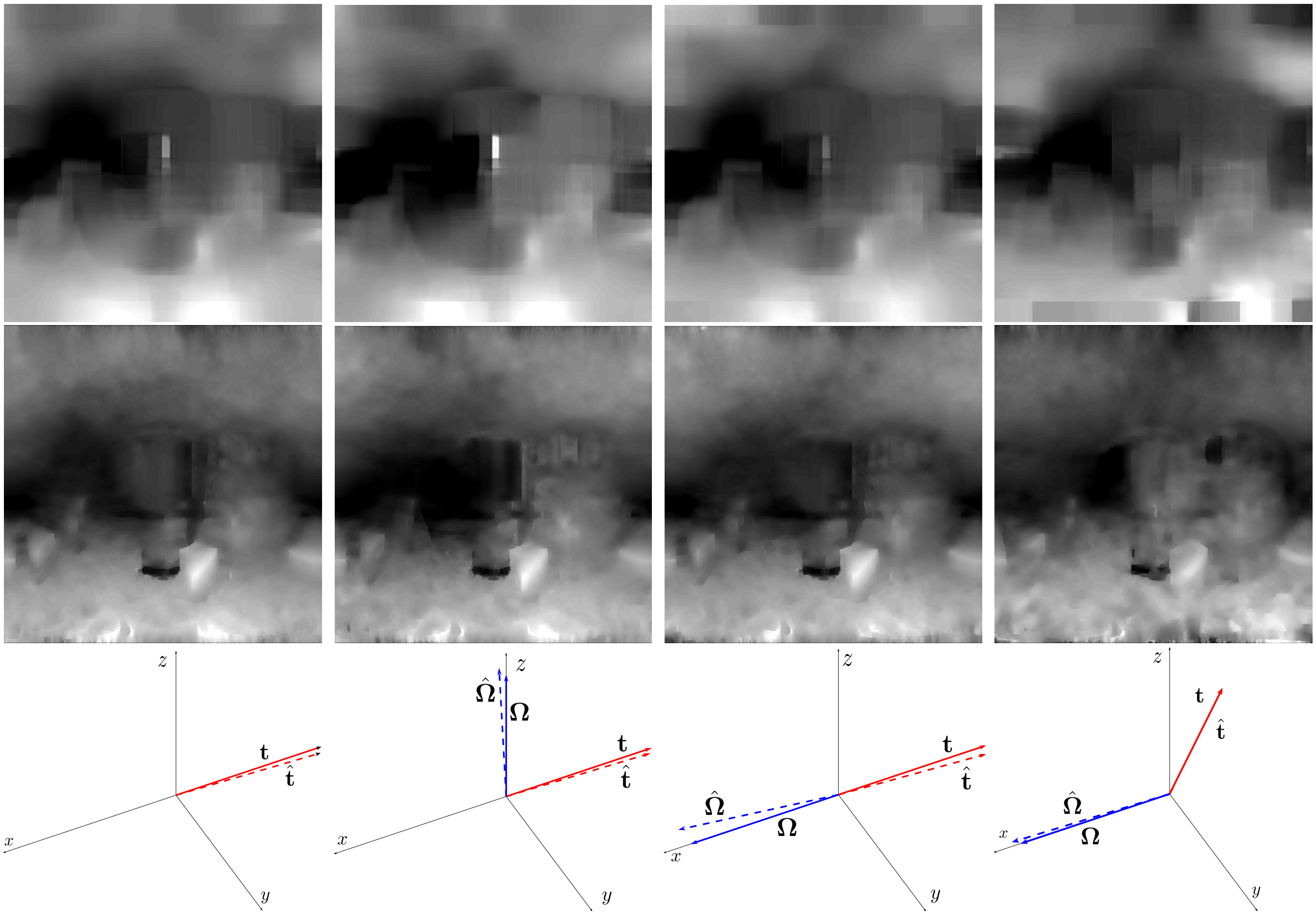

Fig. 9. LK (top) vs TVL1-naive (middle) for four different camera motions. On the bottom we show also $\mathbf{t}$ in red and $\mathbf{\Omega}$ in blue; the estimated motion vectors are represented with a dashed line

Fig. 10. Depth estimation performance for LK (*middle*) and TV-L1-GrH (*right*) algorithm, compared to the ground truth depth map (*Left*).

even if the estimation algorithm is quite simple. The relative error is usually smaller than one percent.

B. Natural omnidirectional images

We study the performance of our algorithm on natural omnidirectional images. These images have been captured by a catadrioptric system positioned in the middle of a room. We then move the camera on the ground plane and rotate it along the vertical axis. The resulting images are shown in Fig. 11, where we also illustrate the result of the projection of the captured images on the sphere. We have also measured the depth map in this environment with help of a laser scanner, and we use these measures for visual evaluation of the depth map estimation algorithm.

TABLE II
RESULTS FOR THE LEAST SQUARE MOTION PARAMETERS ESTIMATION

	Seq1	Seq2	Seq3	Seq4	Seq5
true-$\mathbf{t}$	[-0.1;0;0]	[-0.1;0;0]	[-0.1;0;0]	[0;-0.1;0]	[-0.07;-0.07;0]
$\mathbf{t}$	[-0.099;0.001;-0.004]	[-0.099;0;-0.004]	[-0.099;0.002;-0.005]	[0.;-0.099;-0.006]	[-0.069;-0.07;-0.009]
true-$\mathbf{\Omega}$	[0;0;0]	[0;0;0.0175]	[0.0175;0;0]	[0;0;0.0175]	[0.0175;0;0]
$\mathbf{\Omega}$	[0;-0.001;0]	[-0.0002;-0.0021;0.0167]	[0.0177;-0.0025;0.0001]	[0;0;0.0182]	[0.0181;0;0]

We first analyze the performance of our depth estimation algorithm for natural spherical images, and we compare the estimated depth map to the depth information measured by the laser scanner. We show in Fig. 12 that the estimated depth map is quite accurate, and the estimation algorithm is able to detect all the objects in the scene, and to position them accordingly. Even small details like the legs of the tables are detected, which confirms the efficiency of the variational framework proposed in this paper.

Finally, we show that our depth estimation provides accurate information about the scene content by using this information for image reconstruction. We use one of the images of the natural sequence as a reference image, and we predict the next image using the depth information. We compute the difference between the second image and respectively the reference image, and the approximation of the second image by motion compensation. We can observe in Fig. 13 that the estimated depth map leads to efficient image reconstruction, as the motion compensated image provides a much better approximation of the second image than the reference image. The depth information permits to reduce drastically the energy of the prediction error, especially around the main edges in the sequence. It outlines the potential of our depth estimation algorithm for efficient image or 3D reconstruction.

VII. CONCLUSIONS

We have presented in this paper a novel variational framework for solving the structure from motion problem in omnidirectional images. We have developed a graph-based construction of discrete differential operators for the processing of images on the 2-sphere. These operators permit to develop an efficient algorithm for depth estimation that is robust to the geometrical characteristics of the spherical manifold. We have then proposed a simple algorithm that directly estimates the camera motion from the depth information. Finally, we provide an iterative algorithm for joint depth and ego-motion estimation. The proposed framework provides accurate geometry information for both synthetic and natural omnidirectional images. The efficiency and small complexity of the proposed algorithm positions it as a promising solution for fast depth estimation and scene reconstruction from omnidirectional image sequences.

REFERENCES

[1] O. Faugeras, Q.-T. Luong, and T. Papadopoulo, *The Geometry of Multiple Images*. MIT Press, 2001.

[2] T. Pock, "Fast total variation for computer vision," p. 172, Feb 2008. [Online]. Available: https://online.tu-graz.ac.at/

[3] G. Peyre, S. Bougleux, and L. Cohen, "Non-local regularization of inverse problems," *Computer Vision-Eccv 2008: 10th European Conference on ...*, Jan 2008. [Online]. Available: http://www.ceremade.dauphine.fr/ peyre/publications/08-ECCV-NonLocalInversePbm.pdf

[4] G. Gilboa and S. Osher, "Nonlocal operators with applications to image processing," *UCLA CAM Report*, pp. 07–23, 2007.

[5] D. Zhou and B. Scholkopf, "A regularization framework for learning from graph data," *ICML Workshop on Statistical Relational Learning and Its ...*, Jan 2004. [Online]. Available: http://www.cs.umd.edu/projects/srl2004/srl2004complete.pdf

[6] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. Comput. Vis.*, vol. 47, pp. 7–42, 2002.

[7] A. Makadia, C. Geyer, and K. Daniilidis, "Correspondence-free structure from motion," *Int. J. Comput. Vis.*, vol. 75, no. 3, December 2007.

[8] A. Bruss and B. Horn, "Passive navigation," *Comput Vision Graph*, vol. 21, no. 1, pp. 3–20, Jan 1983. [Online]. Available: http://apps.isiknowledge.com/

[9] D. Heeger and A. Jepson, "Subspace methods for recovering rigid motion .1. algorithm and implementation," *Int J Comput Vis*, vol. 7, no. 2, pp. 95–117, Jan 1992.

[10] A. Jepson and D. Heeger, "A fast subspace algorithm for recovering rigid motion," *Visual Motion, 1991., Proceedings of the IEEE Workshop on*, pp. 124 – 131, Sep 1991. [Online]. Available: $http://ieeexplore.ieee.org/search/srchabstract.jsp?arnumber=212779\&isnumber=5559\&punumber=343\&k2dockey=212779@ieeecnfs$

[11] T. Tian, C. Tomasi, and D. Heeger, "Comparison of approaches to egomotion computation," *Computer Vision and Pattern Recognition, 1996. Proceedings CVPR '96, 1996 IEEE Computer Society Conference on*, pp. 315–320, 1996.

[12] B. Horn and B. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, no. 1-3, pp. 185–203, 1981.

[13] S. Beauchemin and J. Barron, "The computation of optical flow," *ACM Computing Surveys (CSUR)*, vol. 27, no. 3, pp. 433–466, 1995.

[14] B. Horn and E. WELDON, "Direct methods for recovering motion," *Int J Comput Vis*, vol. 2, no. 1, pp. 51–76, Jan 1988. [Online]. Available: http://apps.isiknowledge.com/

[15] D. Sinclair, A. Blake, and D. Murray, "Robust estimation of egomotion from normal flow," *Int J Comput Vis*, vol. 13, no. 1, pp. 57–69, Jan 1994.

[16] J. Gluckman and S. Nayar, "Ego-motion and omnidirectional cameras," *Computer Vision, 1998. Sixth International Conference on*, pp. 999–1005, 1998.

[17] K. Daniilidis, A. Makadia, and T. Bulow, "Image processing in catadioptric planes: spatiotemporal derivatives and optical flow computation," *Omnidirectional Vision, 2002. Proceedings. Third Workshop on*, pp. 3– 10, 2002.

[18] S. Baker and S. Nayar, "A theory of single-viewpoint catadioptric image formation," *Int J Comput Vis*, vol. 35, no. 2, pp. 175–196, Jan 1999. [Online]. Available: http://citeseer.ist.psu.edu/246275

[19] L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal," *Physica D*, vol. 60, pp. 259–268, 1992.

[20] M. Nikolova, "A variational approach to remove outliers and impulse noise," *Journal of Mathematical Imaging and Vision*, vol. 20, pp. 99–120, 2004.

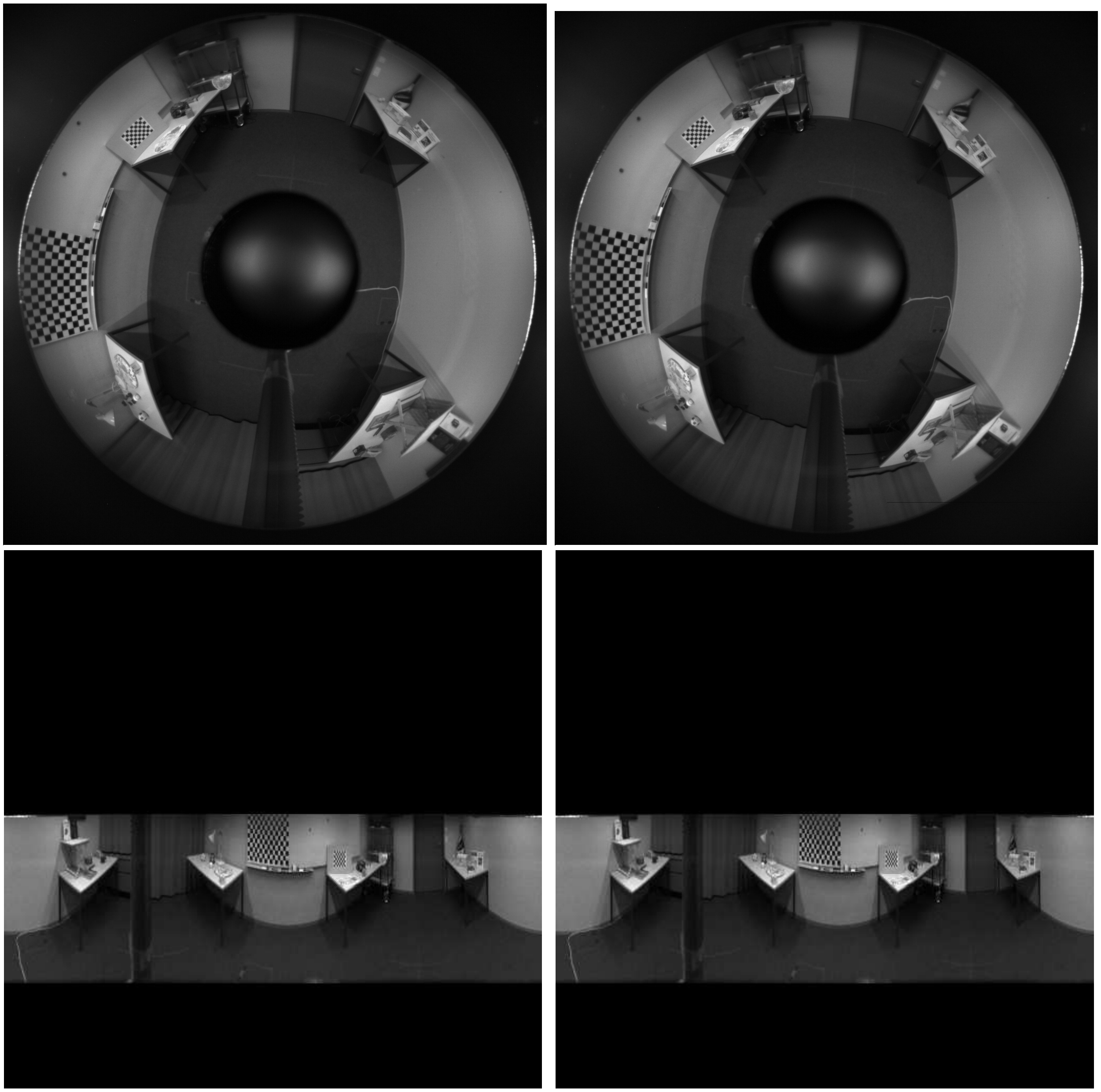

Fig. 11. Natural omnidirectional images from a room. *Top*: Catadioptric image sequence. *Bottom*: Projection of the catadioptric images on a spherical surface.

[21] J. Aujol, G. Gilboa, T. Chan, and S. Osher, “Structure-texture image decomposition - modeling, algorithms, and parameter selection,” *Int J Comput Vis*, vol. 67, no. 1, pp. 111–136, Jan 2006. [Online]. Available: http://www.springerlink.com/content/82873164r897381r/

[22] A. Chambolle, “An algorithm for total variation minimization and applications,” *J Math Imaging Vis*, vol. 20, no. 1-2, pp. 89–97, Jan 2004.

[23] K. Hanna, “Direct multi-resolution estimation of ego-motion and structure from motion,” *Visual Motion, 1991., Proceedings of the IEEE Workshop on*, pp. 156–162, 1991.

[24] A. K. Agrawal, “Robust ego-motion estimation and 3d model refinement using depth,” *Image Processing, 2004. ICIP ’04. 2004 International Conference on*, May 2004. [Online]. Available: http://citeseer.ist.psu.edu/700138

[25] B. Lucas and T. Kanade, “An iterative image registration technique with an application to stereo vision,” *International Joint Conference on Artificial Intelligence*, vol. 81, pp. 674–679, 1981.

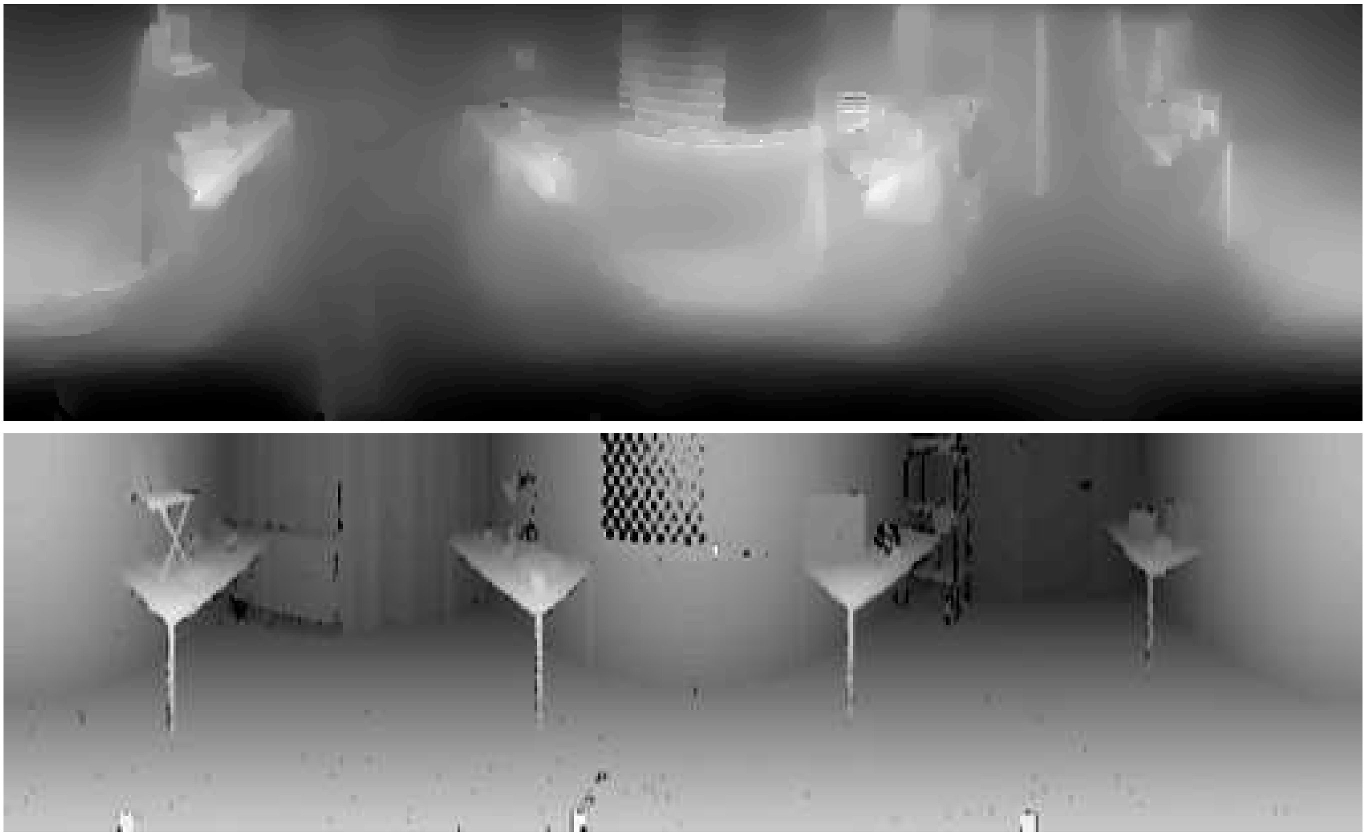

Fig. 12. Performance of the depth estimation algorithm on natural images. Visual comparison of the estimated depth map (*Top*) and the depth map from laser scanner (*Bottom*).

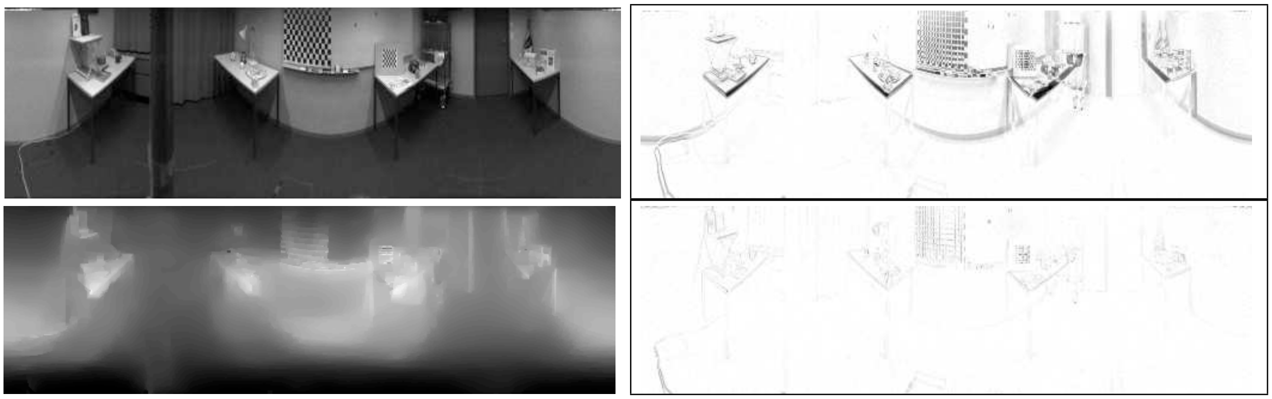

Fig. 13. Analysis of the estimated depth map. *Top - left*: First image of the catadrioptic sequence. *Top - right*: Image difference $I_0 - I_1$. *Bottom - left*: Estimated depth map. *Bottom - right*: Image difference after motion compensation.