JMIV manuscript No.
(will be inserted by the editor)

A Variational Framework for Structure from Motion in Omnidirectional Image Sequences

Luigi Bagnato · Pascal Frossard · Pierre Vandergheynst

Received: date / Accepted: date

Abstract We address the problem of depth and ego-motion estimation from omnidirectional images. We propose a correspondence-free structure-from-motion problem for sequences of images mapped on the 2-sphere. A novel graph-based variational framework is first proposed for depth estimation between pairs of images. The estimation is cast as a TV-L1 optimization problem that is solved by a fast graph-based algorithm. The ego-motion is then estimated directly from the depth information without explicit computation of the optical flow. Both problems are finally addressed together in an iterative algorithm that alternates between depth and ego-motion estimation for fast computation of 3D information from motion in image sequences. Experimental results demonstrate the effective performance of the proposed algorithm for 3D reconstruction from synthetic and natural omnidirectional images.

Keywords Structure From Motion · Ego-Motion · Depth Estimation · Omnidirectional · Variational

This work has been partially supported by the Swiss National Science Foundation under grant 200021-125651.

L. Bagnato
Signal Processing Laboratory (LTS2 and LTS4), Institute of Electrical Engineering, Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, 1015 Switzerland
E-mail: luigi.bagnato@epfl.ch

P. Frossard
Signal Processing Laboratory (LTS4), Institute of Electrical Engineering, Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, 1015 Switzerland E-mail: pascal.frossard@epfl.ch

P. Vandergheynst
Signal Processing Laboratory (LTS2), Institute of Electrical Engineering, Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, 1015 Switzerland E-mail: pierre.vandergheynst@epfl.ch

1 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* (SFM) [9] in the literature. Let us imagine a monocular observer that moves in a rigid unknown world; the SFM problem consists in estimating the 3D rigid self-motion parameters, i.e., rotation and direction of translation, and the structure of the scene, 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 with applications such as autonomous navigation, mixed reality, or 3D video.

In this paper we introduce a novel structure from motion framework for omnidirectional image sequences. We first consider that the images can be mapped on the 2-sphere, which permits to unify various models of single effective viewpoint cameras. Then we propose a correspondence-free SFM algorithm that uses only differential motion between two consecutive frames of an image sequence through brightness derivatives. Since the estimation of a dense depth map is typically an ill-posed problem, we build on [3] and we propose a novel variational framework that solves the SFM problem on the 2-sphere when the camera motion is unknown. Variational techniques are among the most successful approaches to solve under-determined inverse problems and efficient implementations have been proposed recently so that their use becomes appealing [26]. We show in this paper that it is possible to extend very ef-

ficient variational approaches to SFM problems, 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 pixels. We then cast the depth estimation problem as a TV-L1 optimization problem, and we solve the resulting variational problem with fast graph-based optimization techniques similar to [20,10,27]. 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 image pairs.

Then we address the problem of ego-motion estimation from the depth information. The camera motion is not perfectly known in practice, but it can be estimated from the depth map. We propose to compute the parameters of the 3D camera motion with the help of a low-complexity least square estimation algorithm that determines the most likely motion between omnidirectional images using the depth information. Our formulation permits to avoid the explicit computation of the optical flow field and the use of feature matching algorithms. Finally, we combine both estimation procedures to solve the SFM problem in the generic situation where the camera motion is not known a priori. The proposed iterative algorithm alternatively estimates depth and camera ego-motion in a multi-resolution framework, providing an efficient solution to the SFM problem in omnidirectional image sequences. Experimental results with synthetic spherical images and natural images from a catadioptric sensor confirm the validity of our approach for 3D reconstruction.

The rest of the paper is structured as follows. We first provide a brief overview of the related work in Section 2. Then, we describe in Section 3 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 4, and the ego-motion estimation is discussed in Section 5. Section 6 presents the joint depth and ego-motion estimation algorithm, while Section 7 presents experiments of 3D reconstruction from synthetic and natural omnidirectional image sequences.

2 Related work

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. Correspondence-free algorithms get rid of feature computation and matching steps that might prove to be complex and sensitive to transformations between images. Most of the literature in correspondence-free depth estimation is dedicated to stereo depth estimation [22]. 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 as it generally happens in image sequences. This permits to compute the differential motion between images and to build low-complexity depth estimation through image brightness derivatives. Then, most of the research about correspondence-free depth estimation has concentrated on perspective images; the depth estimation has also been studied in the case of omnidirectional images in [18], which stays as one of the rare works that carefully considers the specific geometry of the images in the depth estimation. We handle this geometry by graph-based processing on a spherical manifold and we introduce a novel variational framework in our algorithm, which is expected to provide a high robustness to quantization errors, noise or illumination gradients.

On the other hand, 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 [6,13,16,24]. 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 [14,5]. 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 [15,23] estimate the motion parameters that result into the smallest amount of negative values in the depth map. Some 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 [11], where the optical flow field is estimated in the catadioptric image plane and then back-projected onto a spherical surface. Not many, though, have been trying to take advantage from the wider field of view of the omnidirectional devices: in spherical images the focus of expansion and the focus of contraction are both present, which imply that translation motion cannot be confused with rotational one. In our work, we take advantage of the latter property and directly estimate the ego-motion with a very efficient scheme based on a least square optimization problem, which further permits to avoid the computation of the optical flow.

Ideas of alternating minimization steps have also been proposed in [12,1]. In these works, however, the authors use planar sensors and assume to have an initial rough estimate of the depth map. In addition, they use a simple locally constant depth model. In our experiments we show that this model is an oversimplification of the real world, which does not apply to scenes with a complex structure. In the novel framework proposed in this paper, we use a spherical camera model and we derive a linear set of motion equations that explicitly include camera rotation. The complete ego-motion parameters can then be efficiently estimated jointly with depth.

3 Framework Description

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.

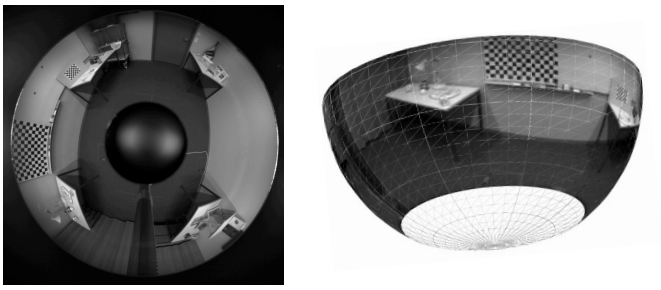

Fig. 1 Left: the original catadioptric image. Right: projection on the sphere

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 [8] 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 [4]. 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:

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

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.

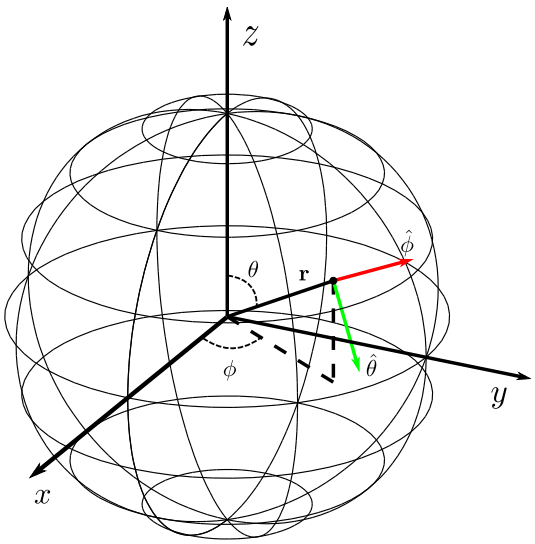

Fig. 2 The representation and coordinate on the 2-sphere S^2.

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. $\mathbf{u} : S^2 \subset \mathbb{R}^3 \to TS$.

3.1 Global motion and optical flow

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

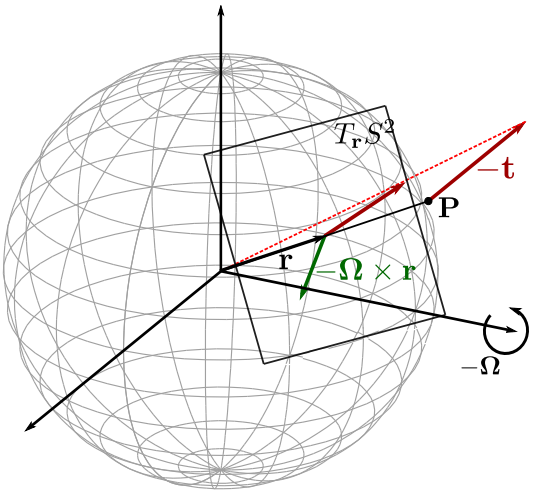

Fig. 3 The sphere and the motion parameters

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, as illustrated in Fig. 3, the scene point moves with respect to the camera by the quantity :

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

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

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

where 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 SFM problem: the depth map $Z(\mathbf{r})$ describing the structure of the scene and the 3D motion parameters $\mathbf{t}$ and $\mathbf{\Omega}$. Due to the multiplication between $Z(\mathbf{r})$ and $\mathbf{t}$, both quantities can only be estimated 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:

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

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.

3.2 Discrete differential operators on the 2-Sphere

We have developed our previous equations in the 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 are used 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. This operation can be performed, for example, by mapping the omnidirectional image on the sphere and then using bilinear interpolation to extract the values at the given positions (θ_m, ϕ_n). In spherical coordinates, a simple discretization of the gradient obtained from finite differences reads:

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

The discrete divergence, by analogy with the continuous settings, is defined by $div = -\nabla^*$ where ∇^* is the

adjoint of ∇. It is then easy to verify that the divergence is given by:

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

Both Eq. (6) and Eq. (7) contain a $(\sin\theta)^{-1}$ term that induces very high values around the poles (i.e., for $\theta \simeq 0$ 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.

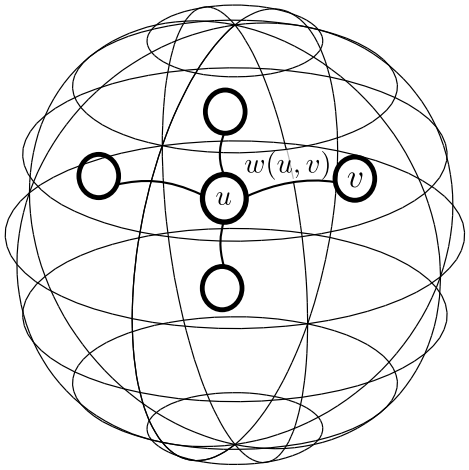

Fig. 4 Embedding of discrete sphere on a graph structure. The pixels u and v in the spherical image represent vertices of the graph, and the edge weight $w(u,v)$ typically captures the geodesic distance between the vertices

We represent our discretized (spherical) imaging surface 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 [27], we now define the gradient and divergence over Γ as :

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

and

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

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

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

The weight $w(u,v)$ is typically defined as a decreasing function of the geodesic distance between the vertices u and v. Since each node of the graph represents a point on the unitary sphere, the geodesic distance between two nodes is defined as the great-circle distance between the corresponding points on the sphere.

Even though both discretization methods are applicable to spherical images, the main advantages of the graph-based representation rely on the definition of differential operators directly in the 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.

4 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 possibly with sharp transitions.

We build the following error functional:

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

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:

$$\rho(I_0, I_1, Z) = I_1(\mathbf{y} + \mathbf{u}_0) + (\nabla I_1(\mathbf{y} + \mathbf{u}_0))^T(-Z(\mathbf{r})\mathbf{t} - \mathbf{\Omega} \times \mathbf{r} - \mathbf{u}_0) - I_0(\mathbf{y}), \quad (12)$$

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-L1 inverse problem. Several advantages come from this choice. First the TV-L1 model is very efficient in removing noise and robust against illumination changes : it inherits these properties from the Rudin-Osher-Fatemi (ROF) model [21] and the L^1 norm fidelity term ensures robustness to outliers and also non-erosion of edges [19]. 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 in practice we work with discrete images. 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 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 [2], 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) as shown in [2]. 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 [7]. 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:

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

where $\mathbf{p}$ is the dual variable, whose definition depends on the image domain. If the domain is the unit sphere S^2, then $\mathbf{p} \in TS$, where TS the tangent bundle already introduced in Section 3. If the image is defined on the graph $\Gamma = (V, E, w)$, then $\mathbf{p} \in E$. 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.

5 Least Square Ego-Motion Estimation

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

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 :

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

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

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

where $A(\mathbf{r}) = [(Z(\mathbf{r})\nabla I_1(\mathbf{r}))^T \ (\mathbf{r} \times \nabla I_1(\mathbf{r}))^T]$, and $C(\mathbf{r}) = I_0(\mathbf{r}) - I_1(\mathbf{r})$ are known matrices, while $\mathbf{b} = [\mathbf{t}; \mathbf{\Omega}]$ is the variable containing the unknown motion parameters.

We formulate the ego-motion estimation problem as follows:

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

The solution to this linear least square problem is simply:

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

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 $Z(\mathbf{r})\mathbf{t} + \mathbf{\Omega} \times \mathbf{r}$, i.e., the spherical projection of 3D motion. For a small field of view, $\mathbf{r}$ does not change much and the two terms $Z(\mathbf{r})\mathbf{t}$ 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 on a spherical retina thanks to the full field of view.

6 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 both estimation algorithms in a dyadic multi-resolution framework.

We embed the minimization process into a coarse-to-fine approach in order to avoid local minima during the optimization and to speed up the convergence of the algorithm. We employ a spherical gaussian pyramid decomposition as described in [25], with a scale factor of 2 between adjacent levels in order to perform the multi-resolution decomposition.

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

1. We use the depth map estimate at the previous level $\bar{Z}^{l+1}(\mathbf{r})$ to initialize the depth map $Z_0^l(\mathbf{r})$ at the current level l. Using the least square minimization from Eq.(25) we can refine the estimation of the motion parameters $\mathbf{t}^l, \mathbf{\Omega}^l$ at level l.
2. Using the estimated motion parameters $\mathbf{t}^l, \mathbf{\Omega}^l$ we can find an estimate of the depth map at current level $Z^l(\mathbf{r})$ by solving Eq. 15 using the variational framework described in Section 4.

Since we perform a coarse-to-fine approach we only need to initialize the algorithm at the coarsest level. Let

us assume that we use L levels. At the coarsest level L we make the hypothesis that a constant-depth model of the scene is sufficient to explain the apparent pixel motion between the low resolution images I_0^L and I_1^L, so we set $Z_0^L = K$, where K is a positive constant different from zero. At the coarsest level, the approximation that we introduce by flattening the depth map Z is well posed since all image edges are smoothed out at low resolution. We also find that the estimation of motion parameters is very accurate at this level independently on the choice of K. The reason is that Z and $\mathbf{t}$ are only known up to a scale factor, while $\mathbf{\Omega}$ is independent of Z. At each level l we can also obtain an estimate of the optical flow $\mathbf{u}_0^l$ as $\mathbf{u}_0^l = -Z_0^l(\mathbf{r})\mathbf{t}^{l+1} - \mathbf{\Omega}^{l+1} \times \mathbf{r}$, and use it to warp image I_1^l, i.e., to estimate $I_1^l(\mathbf{r} + \mathbf{u}_0^l)$. The joint depth and ego-motion estimation algorithm is summarized in Algorithm 1.

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

1. At the coarsest level L initialize: $Z_0^L = K$ with $K > 0$
2. For each level $l \in [L, L-1, \ldots, 2, 1]$:
   (a) Initialize Z with the solution at previous level

   $$Z_0^l = upsample(Z^{l+1}).$$

   (b) Estimate optical flow $\mathbf{u}_0^l$ as:

   $$\mathbf{u}_0^l = -Z_0^l(\mathbf{r})\mathbf{t}^{l+1} - \mathbf{\Omega}^{l+1} \times \mathbf{r}$$

   and use it to calculate $I_1^l(\mathbf{r} + \mathbf{u}_0^l)$.
   (c) Estimate $\mathbf{t}^l$ and $\mathbf{\Omega}^l$ using Eq.(25):

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

   (d) Estimate Z^l using the depth estimation algorithm described in Section 4 with the current estimates $\mathbf{t}^l$ and $\mathbf{\Omega}^l$.

We would like to conclude the section with some considerations regarding the complexity of the algorithm. For the depth map estimation part we observe that the complexity of the algorithm is strongly related to the number of connections in the graph. Usually a 4-connectivity scheme, where each node is connected at most to other 4 neighbors, is enough to represent accurately a sphere. This is specially true in the case of a regular pixelization (e.g., an equiangular grid), where the 4-connectivity scheme is naturally induced by the topology of the sphere (see also Fig. 4). In this case the complexity of the depth estimation algorithm stays equivalent to the case of planar images. Furthermore, since each operation in Eq. (19) and (21) can be performed pixel-wise, the algorithm can be efficiently implemented on graphics processing units in a similar way as described in [26]. The ego-motion estimation algorithm has low complexity since it runs at most in linear time O(n), where n is the total number of pixels. Its complexity is in fact dominated by the operation performed in Eq.(25). Furthermore the quantities $A(\mathbf{r})$ and $C(\mathbf{r})$ can be computed once at the beginning of each multi-resolution level. In practice the ego-motion estimation quickly converges at low resolution levels, so the algorithm can almost be considered as constant time O(1).

7 Experimental results

We analyze in this section the performance of the proposed algorithms for two sets of omnidirectional images, namely a synthetic and a natural sequence. For both

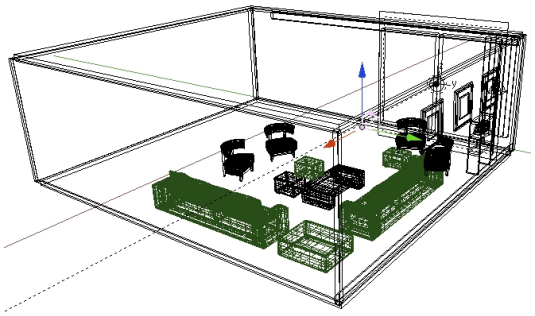

Fig. 5 The 3D model of the scene

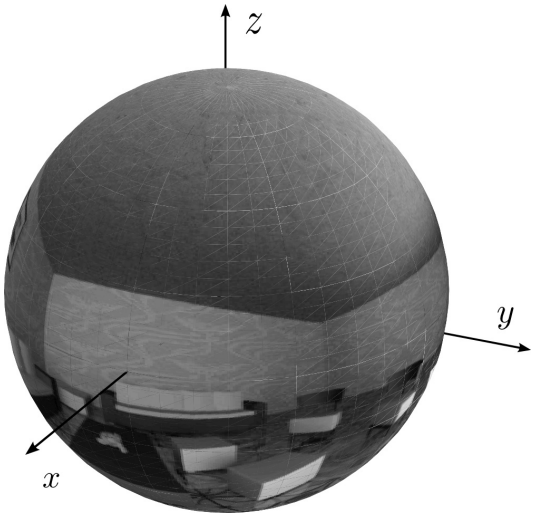

Fig. 6 Synthetic omnidirectional images in spherical representation

sets the images are defined on an equiangular grid, so they are easily representable on a plane, as shown for

example in Fig. 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 corresponds to the north pole and the bottom to the south pole.

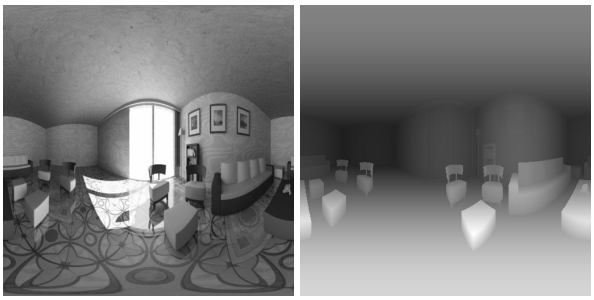

Fig. 7 The synthetic spherical image in a Mercator map (left) and the corresponding depth map ground truth (right)

7.1 Synthetic omnidirectional images

The first set of images is a spherical rendering of the 3D model of a living room shown in Fig. 5. We used the 3D content creation suite Blender to model the room in Fig. 5, while we used the raytracing engine Yafray to render the omnidirectional images and obtain the depth map ground truth. The images used in our experiments can be found online[1]. Since we only use 2 frames in our optimization scheme, in our experiments with the synthetic images the first frame I_0 is always the same and it is shown in Fig. 6 together with the associated depth map, that we use as ground truth for the numerical evaluation of the performance of our algorithm. We generate the other frames by translating and rotating the spherical camera. The camera translation has always the same module of 0.1 units, while the dimension of the room is 24 units by 23 units.

We first study the influence of the discretization scheme in the variational depth estimation algorithm. As discussed in Section 4 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 proposed algorithm, that we call *TVL1-GrH*, clearly leads to improved performance, especially around the poles where it is much more robust than *TVL1-naive*.

[1] http://lts2www.epfl.ch/∼bagnato/datasets/

In Fig. 8 we can observe a black area in the middle of both estimated depth maps, which is not present in the ground truth image. This structure is simply due to an occlusion generated by the reflection of the window, where the brightness consistency does not hold. 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*) similar to the one described in [17] and [12]. This approach assumes that the depth is constant for a given image patch and tries to find a least square depth estimate using the brightness consistency equation. We can observe that the TV-L1 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.

These results are confirmed in Table 1 in terms of mean square error of the depth map reconstruction for several synthetic sequences. It can be seen that the local-costant-depth algorithm *LK* 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. As it has been observed earlier, the discretization from Eq. (6) and (7) (*TVL1-naive*) clearly leads to the worst results, while the graph-based operators perform best.

Finally, we analyze in Table 2 the performance of the ego-motion estimation algorithm proposed in Section 5. 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 even if the estimation algorithm is quite simple. The relative error is usually smaller than one percent.

7.2 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. 10, 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 mea-

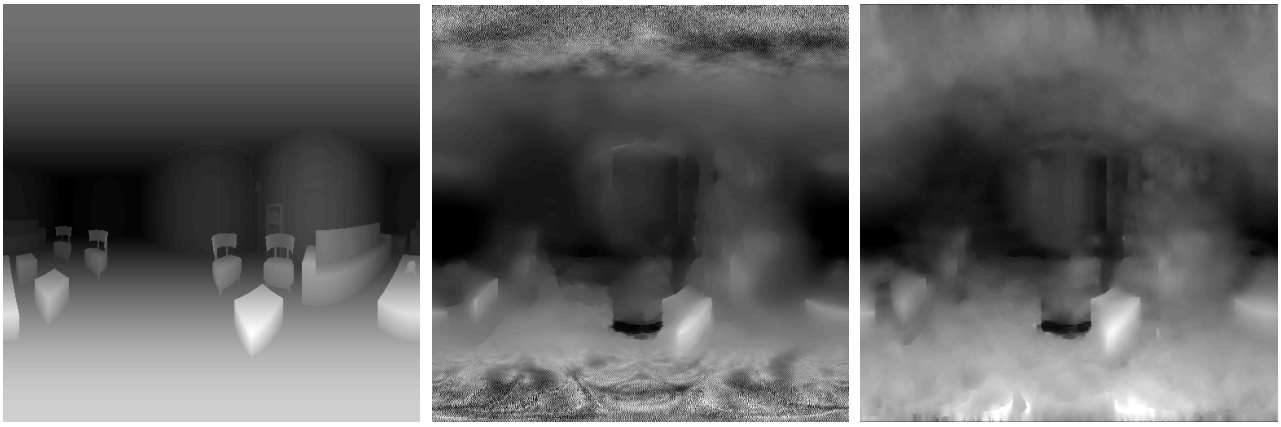

Fig. 8 Depth map estimation with different discrete differential operators. *Left*: ground truth. *Center*: TVL1-naive. *Right*: TVL1-GrH

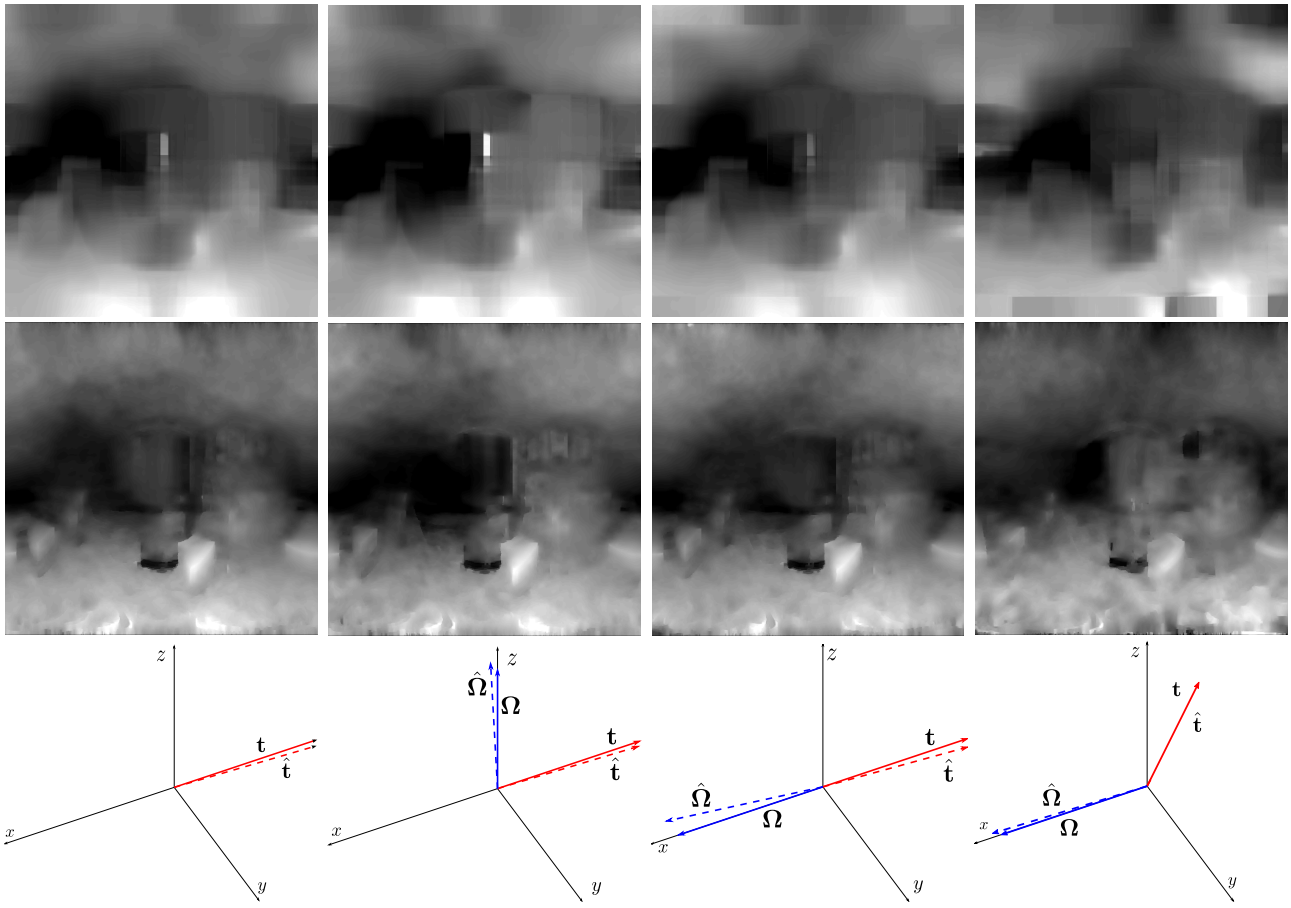

Fig. 9 LK (top) vs TVL1-naive (middle) for four different camera motions. On the bottom we show also $\mathbf{t}$ in red and $\boldsymbol{\Omega}$ in blue; the estimated motion vectors are represented with a dashed line

sures for visual evaluation of the depth map estimation algorithm.

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. 11 that the estimated depth map is quite accurate when compared to the LK algorithm, since the proposed algorithm is able to detect and delineate clearly the objects in the scene. It 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 image sequence as a reference image, and we predict the next image using the

Table 1 Mean Square Error (MSE) between the estimated depth map and the ground truth

	Seq1	Seq2	Seq3	Seq4	Seq5
LK	0.00117	0.00268	0.00158	0.00611	0.00216
TVL1-naive	0.00447	0.10319	0.10234	0.10824	0.10369
TVL1-GrH	**0.00103**	**0.00169**	**0.00167**	**0.00395**	**0.0017**

Table 2 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]

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. 12 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.

8 Conclusions

We have presented in this paper a novel variational framework for solving the structure from motion problem in omnidirectional image sequences. 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 low complexity of the proposed algorithm positions it as a promising solution for fast depth estimation and scene reconstruction from omnidirectional image sequences.

References

1. Agrawal, A., Chellappa, R.: Robust ego-motion estimation and 3d model refinement using depth based parallax model. In: Proceedings of IEEE International Conference on Image Processing, vol. 4, pp. 2483 – 2486 Vol. 4 (2004). DOI 10.1109/ICIP.2004.1421606
2. Aujol, J., Gilboa, G., Chan, T., Osher, S.: Structure-texture image decomposition - modeling, algorithms, and parameter selection. Int. J. Comput. Vis. **67**(1), 111–136 (2006). DOI 10.1007/s11263-006-4331-z
3. Bagnato, L., Frossard, P., Vandergheynst, P.: Optical flow and depth from motion for omnidirectional images using a tv-l1 variational framework on graphs. In: Proceedings of IEEE International Conference on Image Processing, pp. 1469 – 1472 (2009). DOI 10.1109/ICIP.2009.5414552
4. Baker, S., Nayar, S.K.: A theory of single-viewpoint catadioptric image formation. Int. J. Comput. Vis. **35**, 175–196 (1999). DOI 10.1023/A:1008128724364
5. Beauchemin, S., Barron, J.: The computation of optical flow. ACM Computing Surveys (CSUR) **27**(3), 433–466 (1995)
6. Bruss, A., Horn, B.: Passive navigation. Comput Vision Graph **21**(1), 3–20 (1983)
7. Chambolle, A.: An algorithm for total variation minimization and applications. J. Math. Imaging Vis. **20**(1-2), 89–97 (2004)
8. Daniilidis, K., Makadia, A., Bulow, T.: Image processing in catadioptric planes: spatiotemporal derivatives and optical flow computation. Proceedings of the Third Workshop on Omnidirectional Vision pp. 3– 10 (2002)
9. Faugeras, O., Luong, Q.T., Papadopoulo, T.: The Geometry of Multiple Images. MIT Press (2001)
10. Gilboa, G., Osher, S.: Nonlocal operators with applications to image processing. Multiscale Modeling and Simulation **7**(3), 1005–1028 (2008). DOI 10.1137/070698592
11. Gluckman, J., Nayar, S.: Ego-motion and omnidirectional cameras. In: Proceedings of Sixth International Conference on Computer Vision, pp. 999–1005 (1998)
12. Hanna, K.: Direct multi-resolution estimation of ego-motion and structure from motion. Proceedings of the IEEE Workshop on Visual Motion pp. 156–162 (1991)
13. Heeger, D., Jepson, A.: Subspace methods for recovering rigid motion .1. algorithm and implementation. Int. J. Comput. Vis. **7**(2), 95–117 (1992)
14. Horn, B., Schunck, B.: Determining optical flow. Artificial Intelligence **17**(1-3), 185–203 (1981)
15. Horn, B., Weldon, E.: Direct methods for recovering motion. Int. J. Comput. Vis. **2**(1), 51–76 (1988)
16. Jepson, A., Heeger, D.: A fast subspace algorithm for recovering rigid motion. In: Proceedings of the IEEE Workshop on Visual Motion, pp. 124 – 131 (1991). DOI 10.1109/WVM.1991.212779
17. Lucas, B., Kanade, T.: An iterative image registration technique with an application to stereo vision. International Joint Conference on Artificial Intelligence **81**, 674–679 (1981)

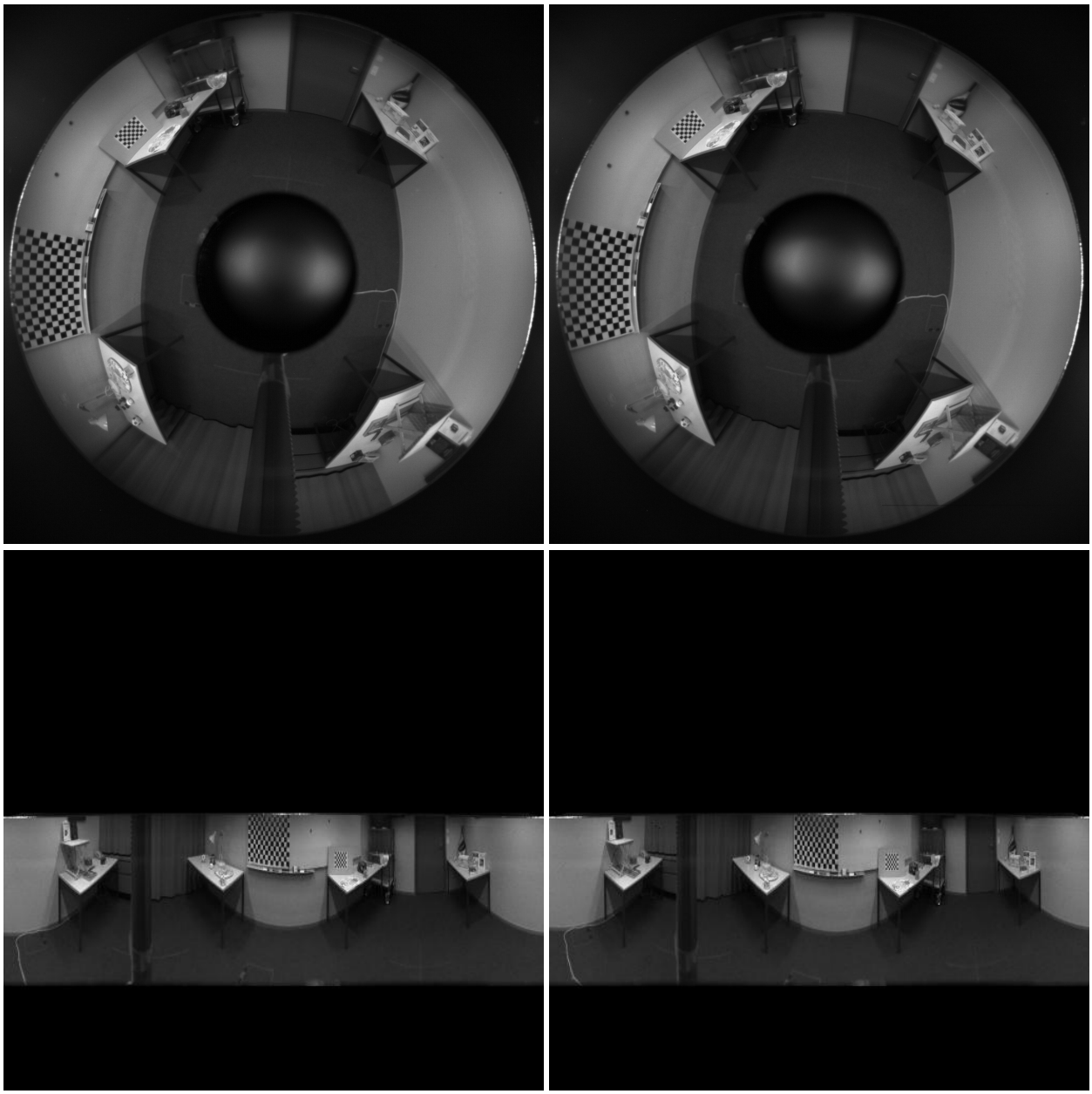

Fig. 10 Natural omnidirectional images from a room. *Top*: Catadioptric image sequence. *Bottom*: Projection of the catadioptric images on a spherical surface

18. Makadia, A., Geyer, C., Daniilidis, K.: Correspondence-free structure from motion. Int. J. Comput. Vis. **75**(3) (2007)
19. Nikolova, M.: A variational approach to remove outliers and impulse noise. Journal of Mathematical Imaging and Vision **20**, 99–120 (2004)
20. Peyré, G., Bougleux, S., Cohen, L.: Non-local regularization of inverse problems. In: D. Forsyth, P. Torr, A. Zisserman (eds.) Computer Vision – ECCV 2008, *Lecture Notes in Computer Science*, vol. 5304, pp. 57–68. Springer Berlin / Heidelberg (2008)
21. Rudin, L., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal. Physica D **60**, 259–268 (1992)
22. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. Int. J. Comput. Vis. **47**, 7–42 (2002)
23. Sinclair, D., Blake, A., Murray, D.: Robust estimation of egomotion from normal flow. Int. J. Comput. Vis. **13**(1), 57–69 (1994)
24. Tian, T., Tomasi, C., Heeger, D.: Comparison of approaches to egomotion computation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 315 –320 (1996)
25. Tosic, I., Bogdanova, I., Frossard, P., Vandergheynst, P.: Multiresolution motion estimation for omnidirectional images. In: Proceedings of EUSIPCO (2005)
26. Zach, C., Pock, T., Bischof, H.: A duality based approach for realtime tv-l1 optical flow. In: F. Hamprecht, C. Schnörr, B. Jähne (eds.) Pattern Recognition, *Lecture Notes in Computer Science*, vol. 4713, pp. 214–223. Springer Berlin / Heidelberg (2007)

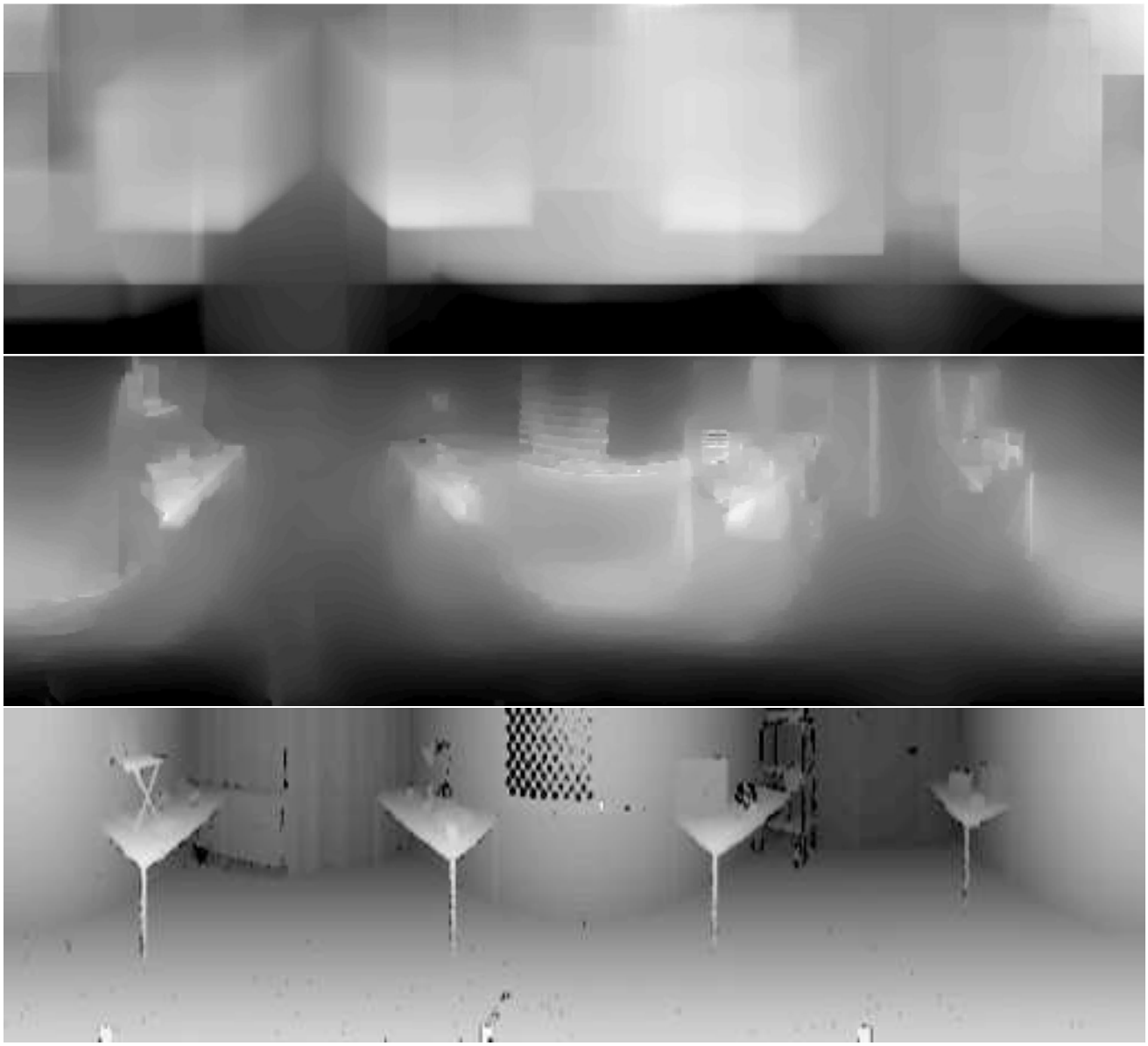

Fig. 11 Visual comparison of the estimated depth map on natural images. (*Top*): LK. (*Middle*): the proposed TVL1-GrH. *Bottom*: depth map from a laser scanner

27. Zhou, D., Scholkopf, B.: A regularization framework for learning from graph data. In: ICML Workshop on Statistical Relational Learning, pp. 132–137 (2004)

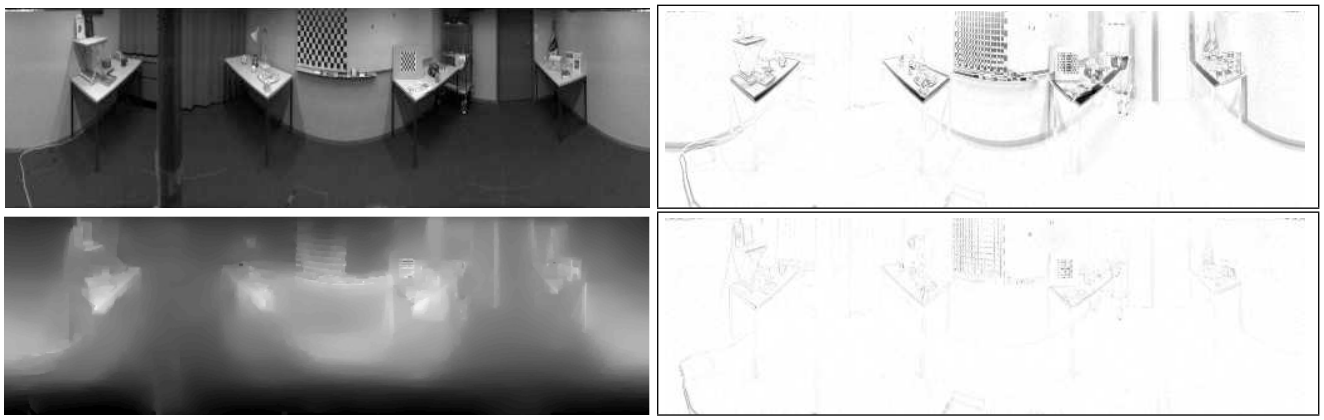

Fig. 12 Analysis of the estimated depth map. *Top - left*: First image of the catadrioptic sequence. *Top - right*: Image difference $I_0 - I_1$. *Bottom - left*: Estimated depth map. *Bottom - right*: Image difference after motion compensation