Dense 3D Structure and Motion Estimation as an aid for Robot Navigation

Geert De Cubber
Royal Military Academy
Department of Mechanical Engineering (MSTA)
Av. de la Renaissance 30, 1000 Brussels
Geert.De.Cubber@rma.ac.be

Abstract

Three-dimensional scene reconstruction is an important tool in many applications varying from computer graphics to mobile robot navigation. In this paper, we focus on the robotics application, where the goal is to estimate the 3D rigid motion of a mobile robot and to reconstruct a dense three-dimensional scene representation. The reconstruction problem can be subdivided into a number of subproblems. First, the egomotion has to be estimated. For this, the camera (or robot) motion parameters are iteratively estimated by reconstruction of the epipolar geometry. Secondly, a dense depth map is calculated by fusing sparse depth information from point features and dense motion information from the optical flow in a variational framework. This depth map corresponds to a point cloud in 3D space, which can then be converted into a model to extract information for the robot navigation algorithm. Here, we present an integrated approach for the structure and egomotion estimation problem.

Introduction

Recovering 3D-information has been in the focus of attention of the computer vision community for a few decades now, yet no all-satisfying method has been found so far. Most attention in this area has been on stereo-vision based methods, which use the displacement of objects in two (or more) images. The problem with these vision algorithms is that they require the matching of feature points, which is not easy for untextured surfaces. Where stereo vision must be seen as a spatial integration of multiple viewpoints to recover depth, it is also possible to perform a temporal integration. The problem arising in this situation is known as the "Structure from Motion" (SfM) problem and deals with extracting 3-dimensional information about the environment from the motion of its projection onto a two-dimensional surface [4].

In general, there are two approaches to SfM. The first, feature based method is closely related to stereo vision. It uses corresponding features in multiple images of the same scene, taken from different viewpoints. The basis for feature-based approaches lies in the early work of Longuet-Higgins [8], describing how to use the epipolar geometry for the estimation of relative motion. In this article, the 8-points algorithm was introduced. It features a way of estimating the relative camera motion, using the essential matrix, which constrains feature points in two images. The first problem with these feature based techniques is of course the retrieval of correspondences, a problem which cannot be reliably solved in image areas with low texture. From these correspondences, estimates for the motion vectors can be calculated, which are then used to recover the depth. An advantage of feature based techniques is that it is relatively easy to integrate results over time, using bundle adjustment [12] or Kalman filtering [5]. Bundle adjustment is a maximum likelihood estimator that consist in minimizing the re-projection error. It requires a first estimate of the structure and then adjusts the bundle of rays between each camera and the set of 3D points.

The second approach for SfM uses the optical flow field as an input instead of feature correspondences. The applicability of the optical flow field for SfM calculation originates from the epipolar constraint equation which relates the optical flow ***u(u,v)*** to the relative camera motion (translation ***t*** and rotation $\boldsymbol{\omega}$) and 3D structure, represented by the depth parameter *d*, in a non-linear fashion, as indicated by equation (1).

$$\mathbf{u} = \mathbf{Q}_{\boldsymbol{\omega}}\boldsymbol{\omega} + d\mathbf{Q}_{\mathbf{t}}\mathbf{t} \tag{1}$$

In [6], Hanna proposed a method to solve the motion and structure reconstruction problem by parameterizing the optical flow and inserting it in the image brightness constancy equation. More popular methods try to eliminate the depth information first from the epipolar constraint and regard the problem as an egomotion estimation problem. Bruss & Horn already showed this technique in the early eighties using substitution of the depth equation [3], while Jepson & Heeger later used algebraic manipulation to come to a similar formulation [7]. The current state-of-the art in SfM systems considers the construction of sparse feature-based scene representations, e.g. from points and lines. The main drawback of such systems is the lack of surface information, which restricts their usefulness, as the number of features is limited. In the past, optical flow - based SfM methods such as the famous Bruss & Horn [3] and Jepson & Heeger [7] algorithms were also mainly aimed at motion and structure recovery using very low resolution optical flows. With the increase in available processing power, however, the SfM community is now trying to address the dense reconstruction problem. The optical flow based SfM approaches are more suited to address dense reconstruction problem, as they can go out from the optical flow over the whole image field. This leads to an approach as sketched by Figure 1 showing two main input paths to the dense reconstruction: sparse epipolar reconstruction and dense optical flow estimation.

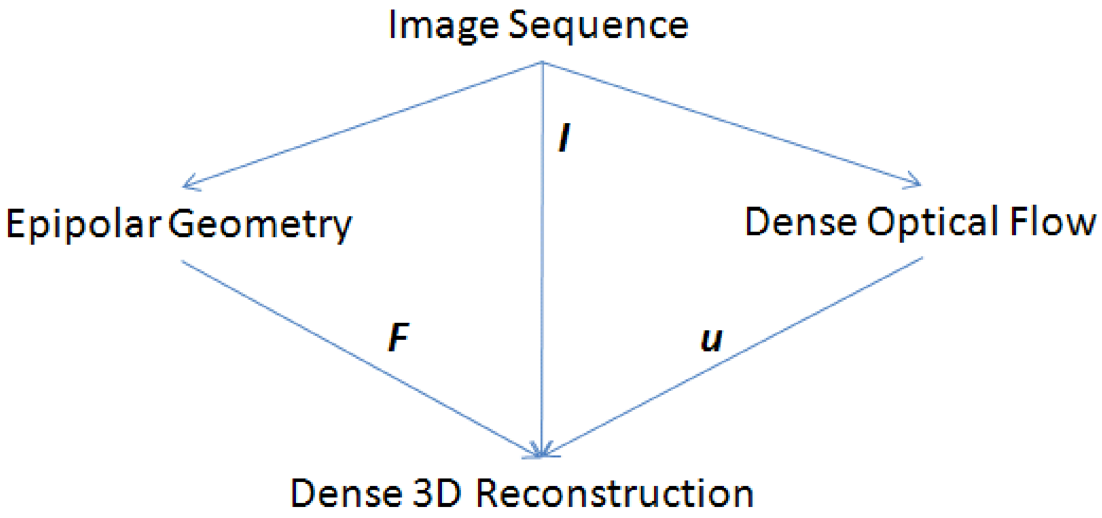

Figure 1: The general approach of the proposed dense 3D reconstruction algorithm: merging sparse information (epipolar geometry) with dense information (optical flow)

In order to bring together the advantages of both sparse and dense SfM theorems, we will here try to fuse both methods into an integrated structure recovery algorithm. In the context of this research work, the situation with a static scene and a dynamic observer is envisaged. The constraint of the static scene can be lifted by incorporating an independent motions segmentation preprocessing algorithm, segmenting the recorded images according to different motion patterns. The SfM analysis is then performed once for each of the motion patterns.

Sparse structure and motion estimation

Feature Detection and Matching

Every sparse structure from motion approach starts off with feature detection and matching to acquire the necessary input data for subsequent processing steps. The aim is to estimate the image locations of a point belonging to a certain 3D structure in different camera views. The landmark or feature point detection was usually performed with the Harris corner detector, but recently, the Scale Invariant Feature Transform (SIFT) is used more and more. A complete use of the SIFT approach has been presented in [10] for reliable point matching between different views of an object or scene. SIFT features are located at scale-space maxima and minima of a difference of Gaussian function. Since the vector of gradients consists of differences of intensity values, it is invariant to affine changes in intensity. Due to these advantageous properties, the SIFT feature detector was selected for our application.

The proposed feature detection and matching method aims at combining the advantages of the SIFT descriptor (robust detector and descriptor) with the advantages of feature tracking, where features can be tracked over longer time spans. On the subject of feature tracking, our choice went to an implementation of the Kanade-Lucas-Tomasi (KLT) Feature Tracking algorithm as proposed by Stan Birchfield in [2]. The Kanade-Lucas-Tomasi (KLT) Feature Tracking algorithm uses the window-based technique proposed in [9], because it is simple, fast, and gives accurate results if windows are selected appropriately. A hybrid feature matching algorithm was set up, beginning with a SIFT feature detection and description process. The detected features are then tracked by a KLT-based tracker.

Structure and Motion Estimation

The first step of the proposed structure and motion estimation procedure consists of an estimation of the optimal framerate using the Geometric Robust Information Criterion (GRIC), introduced by Torr in [11]. This step involves the computation of a scoring function evaluating the goodness of fit of the model. The optimal framerate for each frame is set to be the lowest time step for which the GRIC-scoring function for the fundamental matrix model provides a higher value than the GRIC-scoring function for the homography model. To come to one consistent framerate for the whole image sequence, a globally optimal framerate is calculated by taking an average of the individual optimal time steps for each frame.

In a second stage of reconstruction, three-view geometry reconstruction is performed by estimating the trifocal tensors across image triplets. The trifocal tensor $\mathcal{T} = \left[\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3\right]$ is a 3 x 3 x 3 array of numbers that relate the coordinates of corresponding points or lines in three views. The trifocal tensor estimation algorithm takes as input 6 random correspondences across 3 views, which are used by a non-linear estimation method to compute a first estimate of the trifocal tensor. In a following stage, the support for the proposed trifocal tensor is measured by counting the number of matches which follow the projection model proposed by the given trifocal tensor. This process is repeated numerous times and the trifocal tensor estimate with the largest number of inliers is chosen. In a final step, the trifocal tensor is re-estimated with a linear method by only taking into account the inlier correspondences.

After the trifocal tensors are estimated, the fundamental matrices and camera matrices can be calculated by decomposing the trifocal tensor. The fundamental matrix **F** encapsulates the intrinsic geometry between two views. It is a 3 x 3 matrix of rank 2. It is straightforward to compute the fundamental matrices $\mathbf{F}_{21}$ and $\mathbf{F}_{31}$ between the first and the other views from the trifocal tensor, once the epipoles **e**’ and **e**’’ are known, following equations (2) and (3).

$$\mathbf{F}_{21} = \left[\mathbf{e}'\right]_{\times} \left[\mathcal{T}_1, \mathcal{T}_2, \mathbf{T}_3\right]\mathbf{e}'' \tag{2}$$

$$\mathbf{F}_{31} = \left[\mathbf{e}''\right]_{\times} \left[\mathcal{T}_1^T, \mathcal{T}_2^T, \mathbf{T}_3^T\right]\mathbf{e}' \tag{3}$$

The camera matrix **P** expresses the action of a single projective camera on a point in space in terms of a linear mapping of a homogeneous 3D scene point $\mathbf{X} = (X, Y, Z, T)^T$ to a homogeneous 2D image point $\mathbf{x} = (x, y, w)^T$. In homogeneous coordinates this mapping is written as $\mathbf{x} = \mathbf{PX}$. In the general case, the camera matrix $\mathbf{P} = \mathbf{K}\left[\mathbf{R}^{3\times3} \middle| \mathbf{t}^{3\times1}\right]$ is a 3 x 4 matrix of rank 3, made up of a camera calibration matrix **K** and a rotation matrix **R** and translation vector **t**, relating the camera position and orientation to the world coordinate system. The camera matrices can be calculated from the trifocal tensor by applying equations (4):

$$\begin{aligned} &\mathbf{P} = \left[\mathbf{I} \middle| \mathbf{0}\right] \\ &\mathbf{P}' = \left[\left[\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3\right]\mathbf{e}'' \middle| \mathbf{e}'\right] \\ &\mathbf{P}'' = \left[\left(\mathbf{e}''\mathbf{e}''^T - \mathbf{I}\right)\left[\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3\right]\mathbf{e}' \middle| \mathbf{e}''\right] \end{aligned} \tag{4}$$

The information enclosed in the estimated fundamental matrices is now employed to estimate the camera motion parameters. For this, the essential matrix $\mathbf{E} = \mathbf{K}'^T\mathbf{FK}$, which is the specialization of the fundamental matrix to the case of normalized image coordinates, is calculated from the fundamental matrix. As the essential matrix can also be written as $\mathbf{E} = \mathbf{R}[\mathbf{t}]_\times$, it is clear that the rotation matrix and translation vector can then be estimated by applying singular value decomposition. A problem with this egomotion estimation process is that in general four different solutions are obtained. The correct estimate for rotation matrix and translation vector can be found back by imposing the constraint that reconstructed point locations must lie in front of both cameras. In a next step, the 3D structure data is reconstructed by triangulating all matched pixels to their 3D location. Nonlinear optimization is applied to estimate the point 3D location when multiple matches over time are available.

Up until this point in the SfM estimation procedure, the 3D motion and 3D reconstructions between image triplets were unrelated. Multi view merging addresses this issue. Merging is achieved by estimating the relative scale between the estimated structures and estimating the space homography between the different cameras using a linear least squares method. This space homography is then applied to bring all reconstructions to the same projective basis.

As a last step of the structure and motion estimation process, bundle adjustment is used to produce globally optimal 3D structure and camera motion estimates. This is achieved by minimizing the reprojection error, as expressed by equation (5), by a Levenberg-Marquadt non-linear least squares algorithm, taking advantage of the sparse structure of the system matrices.

$$\min_{\mathbf{P}^j\mathbf{X}_i} \sum_{i \in \text{points}} \sum_{\text{j} \in \text{frames}} d_E\left(\mathbf{x}_i^j, \mathbf{P}^j\mathbf{X}_i\right)^2 \tag{5}$$

Dense structure and motion estimation

The Optical Flow

Optical flow is the distribution of apparent velocities of movement of brightness patterns in an image. Optical flow can arise from relative motion of objects and the viewer. Consequently, optical flow can give important information about the spatial arrangement of the objects viewed and the rate of change of this arrangement. The optical flow estimation technique applied here relates the image correspondence problem from optical flow to other modules such as segmentation, shape and depth estimation, occlusion detection and signal processing.

Dense reconstruction

To reconstruct a dense depth field, it is necessary to maximize the information which can be retrieved out of the given data. To tackle the various data inputs and constraints imposed on the depth reconstruction, energy based methods are very well suited. Here, we follow the approach proposed by Alvarez in [1]. Alvarez proposes an energy based approach to estimate a dense disparity map between two images. Each of the two input paths to the dense reconstruction process, as sketched by Figure 1, needs to be present in the constraint equations. However, only using this information would lead to problems at spatial (image) and temporal (movement) discontinuities. Therefore, an anisotropic smoothing term was added to preserve the depth discontinuities at image discontinuities. Here, we'll elaborate more on the different constraint equations which can be used for this purpose.

The image brightness constraint is based upon the Lambertian assumption that corresponding pixels have equal grey values. To express this, Alvarez first derived a simplified expression for the disparity which is based upon the knowledge of the epipolar geometry, calculated before by the sparse structure and motion estimation algorithms. This formulation can be expressed as:

$$\phi_1 = \left(I_1(x,y) - I_2\left(x + u\left(\lambda(x,y)\right), y + v\left(\lambda(x,y)\right)\right)\right)^2, \tag{6}$$

where I_1 and I_2 represent two image frames and λ is a depth parameter to be estimated. The constraint above does not contain any diffusion terms in feature space. To increase the numerical stability, we add a regularization term. This term has to ensure that discontinuities and smooth areas are well preserved by the reconstruction process. We chose to use the Nagel and Enkelmann regularization model, as this method has already been proven successful in a range of independent experiments. The regularization term has the following form:

$$\phi_2 = (\nabla\lambda)^T \mathbf{D}(\nabla I)(\nabla\lambda) \tag{7}$$

Where **D** is a regularized projection matrix, leaving the energy functional to be minimized as:

$$E = \int_{\Omega} \phi_1 + \mu \phi_2 d\Omega \qquad , \tag{8}$$

where the integration domain is the image field and μ is a regularization parameter. This formulation can be introduced into the Euler-Langrange equation. Eventually, we retrieve:

$$\left(I_1\left(x,y\right) - I_2\left(x+u\left(\lambda\left(x,y\right)\right), y+v\left(\lambda\left(x,y\right)\right)\right)\right) \frac{\partial I_2\left(x+u\left(\lambda\left(x,y\right)\right), y+v\left(\lambda\left(x,y\right)\right)\right)}{\partial \lambda} - \mu div\left(D\left(\nabla I\right)\nabla \lambda\right) = 0 \tag{9}$$

The Euler-Lagrange equation can be solved, provided that an initial condition is given, by calculating the asymptotic state. The initial condition is a backprojected depth map, which we calculate by inserting into equation (1) the dense optical flow estimate and the estimated motion vectors calculated earlier.

Results

A comprehensive test & analysis environment

The main problem in evaluating the performance of any 3D reconstruction algorithm, is the absence of quality ground truth data. Available data on the internet most often only consists of series of images with some camera data. In order to overcome this problem, we went out from an artificial 3D scene and added a well defined camera which we set up to follow a predefined trajectory. By doing so, we were able to control all variables - depth information, camera calibration data and camera motion - needing to be estimated by the structure estimation algorithms. We then made photorealistic renderings of the scene as seen by the camera at different timesteps. These renderings serve as base data for the image processing algorithms. Also the depth information was exported at this stage by constructing depth maps at each time frame. This was achieved by rendering all data as a single Rich Pixel Format (RPF) file. RPF is a multi-layered high precision data format developed for integrated 3D data storage. Figure 2 shows the 3D model along with the camera trajectory and some frames together with their depth map rendering, taken from a 40-frame sequence.

Based on this data, it is now possible to compute for any pixel of an image the ground truth corresponding pixel in any of the cameras. This allows us to extract useful data to analyze the calculation chain of the algorithm. For feature matching between two camera views, ground truth data can be provided by projecting the point in the first camera to its 3D location and by backprojecting this 3D point onto the image plane of the second camera. For epipolar geometry reconstruction, the ground truth camera matrices can be calculated based upon the ground truth motion data, which is also useful to analyze the accuracy of the egomotion estimation.

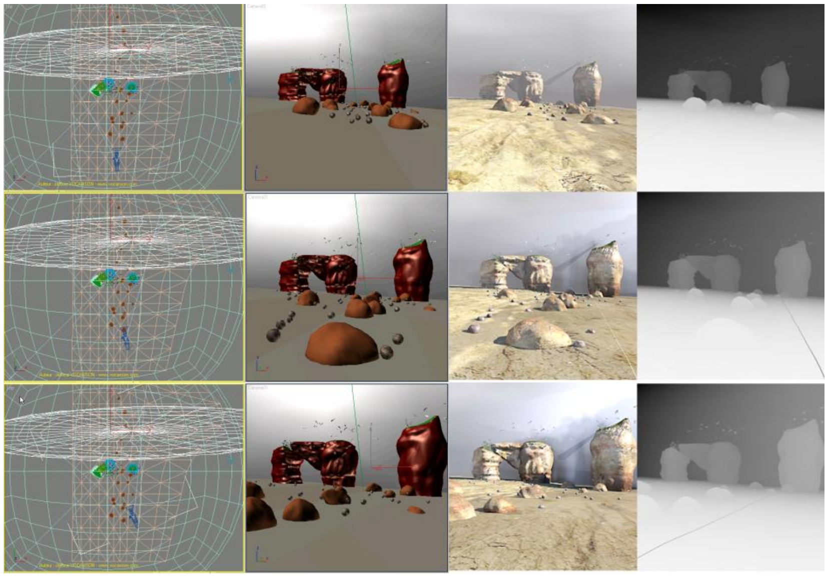

Figure 2: From left to right: the 3D wireframe and model and the camera trajectory used for the test environment (the original 3D model was developed by Jerome Vaucanson in Autodesk® 3D Studio Max®), photorealistic rendering and associated depth map rendering for some frames of the Seaside sequence

This leads to a comprehensive testing and analysis workflow for the evaluation of SfM algorithms. Current approaches are mostly limited to the reconstruction one specific scene. It is firstly hard to quantify the accuracy of a 3D reconstruction on paper and secondly, this leads the way for tuning algorithms towards certain scenes. In our workflow, it is straightforward to change the 3D scene and to produce photorealistic images with ground truth data. It is our hope that with sound benchmarking techniques, the performance of different structure estimation techniques can be accredited in a more reliable fashion.

Feature detection & matching

Figure 3 shows the tracked correspondences, compared to the ground truth feature movement.

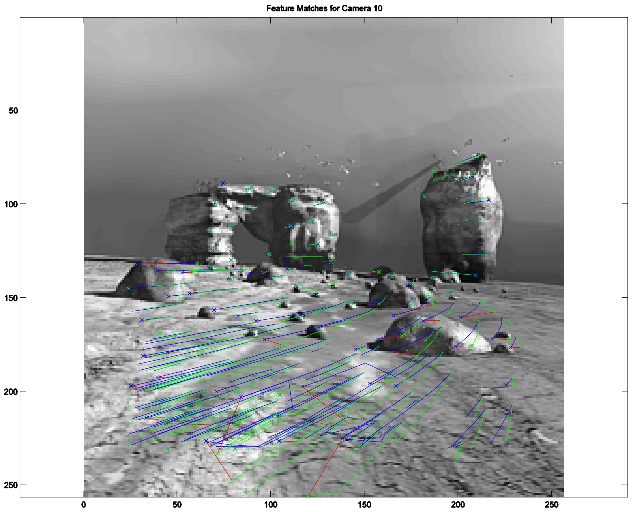

Figure 3: Matches for the different feature points (blue), compared to the ground truth feature motion (green).

As can be noticed, the KLT-tracking algorithm performs very well. Only in the lower right quadrant of the image, the measurement somewhat differs from the ground truth movement due to the relatively large vertical motion field present in this quadrant, while in the other parts of the image, the motion field is mostly horizontal.

Sparse reconstruction

As results for the sparse reconstruction algorithms, we only show the estimated motion vectors on Figure 4, as these results show the applicability of the proposed multi-view reconstruction technique. Figure 4 shows respectively the translation and rotation vector for some camera views and compare them with the ground truth value. To obtain these results, we imposed an extra check in the algorithm such that all vectors acquire the dominant sign computed for the sequence. Doing so makes the translation and rotation vector converge to within an acceptable error margin of the ground truth value, except for one single camera where the estimate for the rotation vector is seriously wrong.

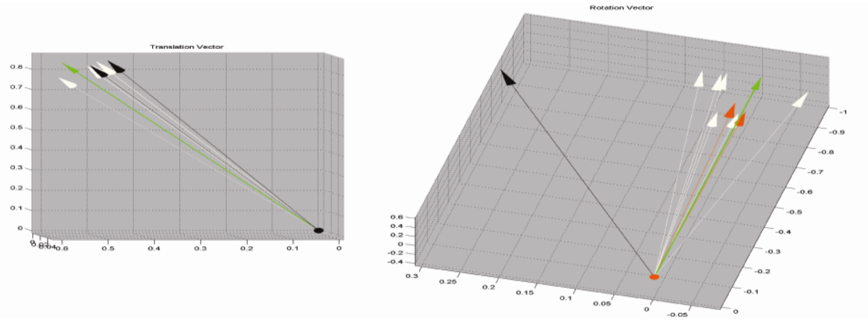

Figure 4: Estimates of the 3D motion vectors for different camera views compared to the ground truth motion in green

Dense reconstruction

In order to preserve stability, we chose to use a semi-implicit numerical scheme to calculate the depth field iteratively. Figure 5 compares the obtained result from dense reconstruction to the ground truth depth map. It is clear that artifacts are still visible, but the relative depths can be discerned very well. Calculation time for this estimation is about 5 minutes on a 3.0GHz CPU.

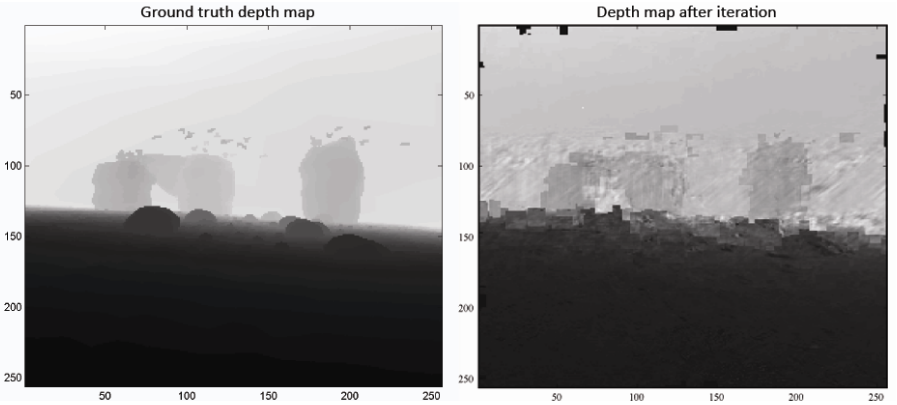

Figure 5: The ground truth depth map and the depth map retrieved after dense reconstruction

Conclusions and future work

In this paper, we proposed an approach towards dense depth reconstruction. The approach aims to combine the strength of the more robust feature-based structure from motion approaches with the spatial coherence of dense reconstruction algorithms. To achieve this, a variational framework was set up, minimizing the epipolar reprojection error and the image brightness constraint, while preserving discontinuities in the depth field by introducing an anisotropic diffusion term. The dense optical flow information is backprojected and serves as initial guess for the iterative solver. The resulting depth maps can serve a very useful input for a robot navigation planner as they provide rich information about the environment. Using this data for robotic navigation tasks in outdoor environments will be one of the first issues to address with respect to future work.

References

[1] L. Alvarez, R. Deriche, J. Sanchez, and J.Weickert. Dense disparity map estimation respecting image derivatives: a PDE and scale-space based approach. *Journal of Visual Communication and Image Representation*, 13(1/2):3–21, 2002.

[2] Stan Birchfield. Klt: An implementation of the kanade-lucas-tomasi feature tracker. http://www.ces.clemson.edu/stb/klt/, January 1997.

[3] Anna R Bruss and Berthold K. P Horn. Passive navigation. *Computer Vision, Graphics and Image Processing*, 21:3–20, January 1983.

[4] A. Chiuso, P. Favaro, H. Jin and S. Soatto. Structure from motion causally integrated over time. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(4):523–535, 2002.

[5] Hailin Jin, Paolo Favaro, and Stefano Soatto. A semi-direct approach to structure from motion. *The Visual Computer*, 19(6):377–394, October 2003.

[6] K. J. Hanna. Direct multi-resolution estimation of ego-motion and structure from motion. In *Workshop on VisualMotion*, pages 156–162, October 1991.

[7] D.J. Heeger and A.D. Jepson. Subspace methods for recovering rigid motion: Algorithm and implementation. *International Journal of Computer Vision*, 7(2):95–117, 1992.

[8] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293(5828):133135, September 1981.

[9] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Int. Conf. on Artificial Intelligence*, pp. 674–679, Vancouver, 1981.

[10] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):pp. 91–110, 2004.

[11] P. H. S. Torr. Bayesian model estimation and selection for epipolar geometry and generic manifold fitting. *Int. J.Comput. Vision*, 50(1):35–61, 2002.

[12] B. Triggs, P.F. Mclauchlan, R.I. Hartley and A.W. Fitzgibbon. Bundle adjustment – a modern synthesis. *Lecture Notes in Computer Science*, 1883:298 – 372, 2000.