ISSN 2007-9737

Depth Map Building and Enhancement using a Monocular Camera, Shape Priors and Variational Methods

Andrés Díaz[1], Lina Paz[2], Pedro Piniés[2], Eduardo Caicedo[1]

[1] Universidad del Valle,
Colombia

[2] Intel Corporation,
USA

{andres.a.diaz, eduardo.caicedo}@correounivalle.edu.co,
{paz.linapaz, peternac@gmail.com}

Abstract. We present a monocular system that uses shape priors for improving the quality of estimated depth maps, specially in the region of an object of interest, when the environment presents complex conditions like changes in light, with low-textured, very reflective and translucent objects. A depth map is built by solving a non-convex optimization problem using the primal-dual algorithm and a coupling term. The energy functional consists of a photometric term for a set of images with common elements in the scene and a regularization term that allows smooth solutions. The camera is moved by hand and tracked using ORB-SLAM2. The resulting depth map is enhanced by integrating, with a novel variational formulation, depth data coming from the 3D model that best fits to observed data, optimized w.r.t. shape, pose and scale (shape prior). We also present an alternative algorithm that simultaneously builds a depth map and integrates a previously estimated shape prior. We quantify the improvements in accuracy and in noise reduction of the final depth map.

Keywords. Dense mapping, shape priors, variational methods, primal-dual algorithm, depth integration, depth denoising.

1 Introduction

For building a depth map with a monocular camera its location for a set of frames must be known, photo-consistency must be satisfied and the images must have texture. However, real environments present changes in light conditions and low-texture, very reflective or translucent objects. These facts break the Lambertian condition (photo-consistency) and affects the photometric error estimation reducing the accuracy of the estimated depth maps. The regularizer term in a variational framework tackles this problem to some extent, but under difficult conditions, the estimations still have low accuracy.

One alternative is to include information of a known object in the scene (shape prior). In this sense, we propose to optimize a model w.r.t. shape, pose and scale and then include depth data of the shape prior seen from the estimated camera pose. This data integration is done using also variational methods and the primal dual algorithm, achieving a denoised and enhanced depth map, especially in the region of the selected object.

The main contributions of this work are as follows. (1) The coupling of four modules: a module for tracking the camera based on keyframes, bundle adjustment and ORB features called ORB-SLAM2; a module for dense mapping based on a photometric error, a regularizer and a decoupling term; a module for estimating the optimal 3D model that best fits to observed data based on Gaussian Process Latent Variable Models GPLVM ; a module for denoising, inpainting and depth merging based on variational methods.

(2) A novel variational formulation that integrates in one algorithm both the module for dense mapping with monocular camera and the module for depth merging using a shape prior. (3) The experiments carried out in order to quantify the improvements in depth accuracy.

This paper is structured as follows. In section II we describe related work. In section III we present the proposed methodology: initial depth map estimation, depth refinement using variational methods, shape prior estimation, integration of depth data of the optimal model (shape prior) for enhancing the estimated depth map in sequential (depth map building followed by shape prior integration) and simultaneous way (depth map building and shape prior integration at the same time). Finally, we present the results and conclusions in section IV and V, respectively.

2 Related Work

Two techniques for minimizing the energy functional in the process of building a depth map with a monocular camera stand out. The technique of sequential convex optimization linearises the photometric error, as is explained in [20], so the camera motion must be small.

A coarse-to-fine scheme with a pyramid of several levels is built to copy with fast camera motion. This technique was successfully implemented in the work of dense mapping of [18]. The other technique, the non-convex variational one, based on optical flow for long displacements [17], uses an auxiliary variable that decouples the cost function into two terms. The regularizer term is solved with the *primal-dual* algorithm [1], [21], and the photometric term is solved by exhaustive search over a finite range of discrete values of the inverse depth. This technique was implemented in the work of dense localization and dense mapping of [12].

In order to improve the accuracy of depth maps created with a monocular camera, a shape prior that considers the scene with box-like structures, with extensive low-texture surfaces like walls, ceilings and floors, can be used. This scene shape prior allows to improve the whole scene. For example, the system [13] estimates depth maps using a monocular camera in workspaces with large plain structures like floors, walls or ceilings. The curvature of a second order approximation of the data term at the minimum cost defines the reliability of the initial depth, getting good depth estimates at the borders of bland objects (high curvature). Good depth data is propagated to an interior pixel (inpainting) from the closest valid pixels along the main 8 star directions by using a non-local high-order regularization term, in a variational approach, that favours solutions with affine surfaces (prior). The energy is minimized in straight way with the primal-dual algorithm.

The system [3] shows outstanding performance in low-textured image regions and for low-parallax camera motion. It includes a term, besides the data term and regularization term, that depends on three scene priors: planarity of homogeneous color regions (using superpixels), the repeating geometry primitives of the scene (data-driven 3D primitives learned from RGBD data), and the Manhattan structure of indoor rooms (layout estimation and classification of box-like structures). The scene prior terms model the distance from every point to its estimated planar prior. The energy is minimized using a variational approach with a coupling term, the primal-dual algorithm and exhaustive search. In contrast to [13], it requires a preprocessing step.

Other kind of shape prior, the object-based one, is also used for 2D segmentation, 3D reconstruction and point cloud refinement. The monocular system [14] uses DCT for compressing the 3D level-set embedding functions and GPLVM for nonlinear dimensionality reduction to capture shape variance.

The energy function measures the discrepancy between the projected 3D model into the image plane and the probabilistic 2D occupancy map that defines the foreground of the observed object in the image (image-based energy). The minimization is done w.r.t. pose and shape of the 3D model. A 2D segmentation results automatically after convergence. The system [4] also uses DCT, GPLVM, and a monocular camera but unlike [14] it builds depth maps minimizing the photo-consistency error with variational methods

and PTAM [7] for camera tracking and fuses them into a volumetric grid through time.

The main goal is to improve the dense reconstruction by replacing the TSDF values of the optimal model in the volumetric grid in a straight way. Moreover, the energy function combines image and depth data for pose, shape and scale optimization (image and depth-based energy). The system [9] removes point cloud artifacts like noisy points, missing data and outliers using a learned shape prior. Besides using DCT and GPLVM as [4, 14], it uses part-based object detector [5] for detecting the object in the scene, VisualSFM [19] for performing structure from motion and getting a point cloud that represents the scene, SAC-segementation for segmenting the point cloud into the region of the object, and iterative optimization of an energy function that depends on the evaluation the point cloud into the embedding function (depth-based energy). The shape prior is finally used for enhancing the accuracy and the completeness of the estimated 3D representation.

Our system uses, like in [12], an auxiliary variable that decouples the data term and the regularization term. The solution is found with the primal-dual algorithm and exhaustive search. We employ object shape priors like [4, 9], but instead of modifying a point cloud like [9] or a volumetric structure directly like [4], we enhance the built depth maps by merging a synthetic depth map coming from the shape prior using a novel variational formulation that considers an additional term for the shape prior data, like is done in [3, 13], and exploiting the ideas of [8] where color aerial images are fused considering the redundant information of the scene.

Finally, we propose to couple both modules (depth map creation and shape prior integration) in just one module, considering a known shape prior previous to build the depth map.

3 Methodology

The main pipeline of the system is shown in fig. 1. An initial depth map is estimated by minimizing the photo-metric error gathered from a set of images with the camera pose estimated with ORB-SLAM2. This coarse depth map is refined using a variational framework with an energy functional made up of a data term, a regularizer term and an additional decoupling term. The primal-dual algorithm and exhaustive search are employed in an alternating fashion for solving this problem.

We use DCT for compressing the 3D models of the object of interest represented as 3D level sets embedding functions, GPLVM for dimensionality reduction and Levenberg-Marquart for minimizing the discrepancy between a model hypothesis and depth data of the segmented region of the object, having as argument its shape, pose and scale.

The optimal model is used for creating a synthetic depth map by reading the depth buffer of its explicit representation in OpenGL, seen from the estimated camera pose. The synthetic depth map is merged with the built depth map using a novel variational formulation. Finally, a variant in this formulation is presented for making simultaneously depth map building and shape prior integration.

3.1 Building an Initial Depth Map

A 3D point in the reference frame r is represented as $\boldsymbol{X_r} = (X_r, Y_r, Z_r)$. This 3D coordinate can be computed as:

$$\boldsymbol{X_r}(\mu) = {\zeta_r}^{-1}(\boldsymbol{\mu}) K^{-1} \hat{\boldsymbol{\mu}}, \tag{1}$$

where $\hat{\boldsymbol{\mu}}$ is the homogeneous version of the coordinate in the image plane, that is $\hat{\boldsymbol{\mu}} = (u_r, v_r, 1)^T$, K is the intrinsic camera matrix, and ζ_r is the inverse depth. We use inverse depth instead of depth because a uniform sampling of the inverse depth corresponds to a uniform sampling in epipolar lines in the image, allowing the system to make exhaustive search to solve the photometric error and the decoupling term in the refinement process. The 3D point $\boldsymbol{X_r}$ can be referenced to the camera frame c, as follows:

$$\boldsymbol{X}_c(\boldsymbol{\mu}) = T_{cr} \boldsymbol{X}_r(\boldsymbol{\mu}). \tag{2}$$

This 3D point is projected into the image plane $\boldsymbol{I}_c$ for getting the coordinate $\boldsymbol{\mu}_c = (u_c, v_c) = \acute{\boldsymbol{\mu}} = K\pi(\boldsymbol{X}_c)$ that corresponds to the estimated observation of the 3D point from

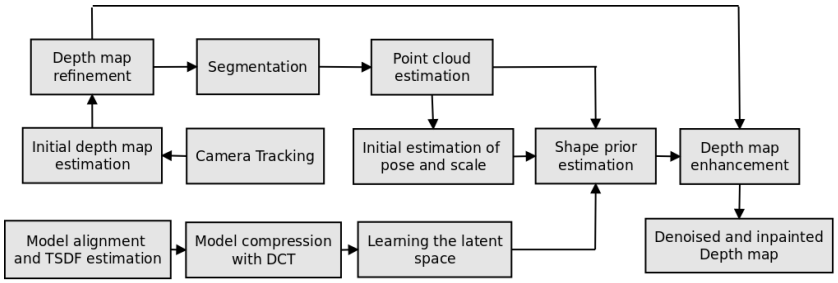


Fig. 1. Main pipeline of the proposed system. The resulting depth map integrates depth data from the optimal model (shape prior)

the new camera pose (see fig. 2), where π represents the function that computes normalized homogeneous coordinates:

$$\pi(\boldsymbol{X}_c) = (X_n, Y_n, 1)^T = \left(\frac{X_c}{Z_c}, \frac{Y_c}{Z_c}, 1\right)^T. \quad (3)$$

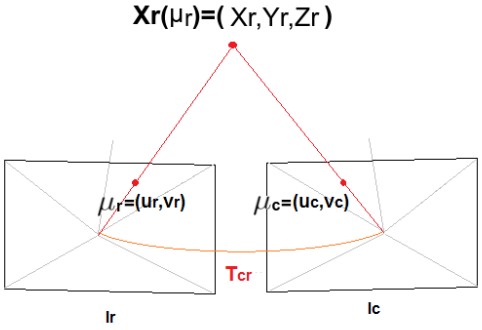


Fig. 2. Projection of a 3D point referenced to the coordinate system r into the image plane c, using the transformation T_{cr}

If the same process is carried out for all the pixels of the reference image, a synthetic image that represents the scene observed from the pose defined by the transformation matrix T_{cr}, can be estimated.

However, the inverse depth of the pixel $\zeta_r(\boldsymbol{\mu})$, required in equation (1), is unknown. To estimate the inverse depth of each pixel of the image $\boldsymbol{I}_r$, the photometric error is defined as the difference in intensity between a pixel of the reference image $\boldsymbol{I}_r(\boldsymbol{\mu})$ and a pixel of the current image in the projected coordinate $\boldsymbol{I}_c(\acute{\boldsymbol{\mu}})$, that is:

$$\rho_r(\boldsymbol{I}_c, \boldsymbol{\mu}, \zeta_r(\boldsymbol{\mu})) = \boldsymbol{I}_r(\boldsymbol{\mu}) - \boldsymbol{I}_c(\acute{\boldsymbol{\mu}}) = \boldsymbol{I}_r(\boldsymbol{\mu}) - \boldsymbol{I}_w(\boldsymbol{\mu}), \quad (4)$$

where the projected image $\boldsymbol{I}_w(\boldsymbol{\mu})$ is:

$$\boldsymbol{I}_w(\boldsymbol{\mu}) = \boldsymbol{I}_c(K\pi(T_{cr}{\zeta_r}^{-1}(\boldsymbol{\mu})K^{-1}\dot{\boldsymbol{\mu}})), \quad (5)$$

as T_{cr} is supposed to be known, the true value of $\zeta_r(\boldsymbol{\mu})$ minimizes the photometric error. Now, this process is extended to a set of consecutive images $\boldsymbol{I}_{c_i} : i \in [1, n]$ that share common elements of the scene. An average photometric error is defined as a function of the inverse depth $\zeta_r(\boldsymbol{\mu})$:

$$\boldsymbol{C}_r(\boldsymbol{\mu}, \zeta_r(\boldsymbol{\mu})) = \frac{1}{n}\sum_{i=1}^{n} \|\rho_r(\boldsymbol{I}_{c_i}, \boldsymbol{\mu}, \zeta_r(\boldsymbol{\mu}))\|_1, \quad (6)$$

where ρ was defined in the equation (4). Finally, the problem of finding the inverse depth of a pixel $\zeta_r(\boldsymbol{\mu})$ is equivalent to solve:

$$\min_{\zeta_r(\boldsymbol{\mu})} \boldsymbol{C}_r(\boldsymbol{\mu}, \zeta_r(\boldsymbol{\mu})). \quad (7)$$

In real environments, there are changes in light conditions, which break the assumption of photo-consistency and the estimations are affected drastically. Besides, images of real scenes present regions with low texture that generate depth estimations in the most dominated by noise [2]. All these problems are reduced to some extent when using a regularizer in a variational framework.

3.2 Refining the Initial Depth Map with Variational Methods

A variational approach [16] is adopted for smoothing the depth map, preserving discontinuities and

increasing the robustness of the algorithm against illumination changes, occlusions and noise. It was proposed first by Ruding, Osher and Fatemi ROF to consider the "Total Variation" as a regularizer $\int_\Omega |\nabla h(\boldsymbol{\mu})| d\boldsymbol{\mu}$, for functions $h(\boldsymbol{\mu})$ in the Sobolev space $W^{1,1}$. The big advantage is that it is convex in the variable h, so this problem has a unique solution. For the pure denoising case [15] it is:

$$\min_h \int_\Omega \underbrace{\|\nabla h(\boldsymbol{\mu})\|_1}_{\text{regularizer term}} d\boldsymbol{\mu} \quad + \frac{\lambda}{2} \int_\Omega \underbrace{\|h(\boldsymbol{\mu}) - g(\boldsymbol{\mu})\|_2^2}_{\text{data term}} d\boldsymbol{\mu}, \tag{8}$$

where h is the sought solution and g is the noisy input image. The parameter λ defines the tradeoff between regularization and data fitting. In our context, the data term measures the photo-consistency between images, the regularizer term smoothes surfaces preserving discontinuities and λ plays the same role as in eq. (8), resulting the energy functional:

$$E(\boldsymbol{\mu}, \zeta_r(\boldsymbol{\mu})) = \int \left(\underbrace{w(\boldsymbol{\mu}) \|\nabla \zeta_r(\boldsymbol{\mu})\|_\epsilon}_{\text{regularizer term}} + \underbrace{\lambda \boldsymbol{C}(\boldsymbol{\mu}, \zeta_r(\boldsymbol{\mu}))}_{\text{data term}} \right) d\boldsymbol{\mu}, \tag{9}$$

where $w(\boldsymbol{\mu})$ is a weighting function, $\boldsymbol{C}(\boldsymbol{\mu}, \zeta_r(\boldsymbol{\mu}))$ is the photometric error, and $\|\cdot\|_\epsilon$ is the Huber norm over the gradient of the inverse depth map, with:

$$\|\boldsymbol{x}\|_\epsilon = \begin{cases} \frac{\|\boldsymbol{x}\|_2^2}{2\epsilon} & \text{if } \|\boldsymbol{x}\|_2 \leq \epsilon, \\ \|\boldsymbol{x}\|_1 - \frac{\epsilon}{2} & \text{otherwise} \end{cases} \tag{10}$$

The L_2^2 norm promotes smooth solutions while the L_1 norm (total variation regularizer) allows discontinuities at depth edges. As depth discontinuities often coincide with edges in the reference image, the per pixel weight $w(\boldsymbol{\mu})$ is:

$$w(\boldsymbol{\mu}) = e^{-\alpha \|\nabla \boldsymbol{I}_r(\boldsymbol{\mu})\|_2^\beta}, \tag{11}$$

reducing the regularity strength where the edge magnitude is high, therefore decreasing the smoothing effect in boundaries. The problem of computing the inverse depth map becomes:

$$\min_{\zeta_r(\boldsymbol{\mu})} E(\boldsymbol{\mu}, \zeta_r(\boldsymbol{\mu})), \tag{12}$$

where E is the energy defined in eq. (9). It is a non-convex problem: the regularizer term is convex and the photometric error is not convex. Next, we describe how to solve it with a decupling term.

3.2.1 The Decoupling Approach

In order to solve (12), we use the iterative primal-dual algorithm described in [12] for depth map building. This algorithm requires both the regularizer and the data term to be convex. However the last term is not a convex function. One solution to this problem is to decouple both terms and solve the decoupled version instead of the original one. The advantage of the decoupling approach is that it allows us to independently solve for the regularizer term using convex optimization methods and for the data term using a simple exhaustive search. The decoupling approach is based on eliminating the constraint $\zeta_r(\boldsymbol{\mu}) = \eta(\boldsymbol{\mu})$ of the problem:

$$\begin{array}{c} \min_{\zeta_r, \eta} E_{\text{reg}}(\zeta_r(\boldsymbol{\mu})) + E_{\text{data}}(\eta(\boldsymbol{\mu})), \\ s.t. \qquad \zeta_r(\boldsymbol{\mu}) = \eta(\boldsymbol{\mu}), \end{array} \tag{13}$$

where $E_{\text{reg}}(\zeta_r(\boldsymbol{\mu})) = w(\boldsymbol{\mu}) \|\nabla \zeta_r(\boldsymbol{\mu})\|_\epsilon$, $E_{\text{data}}(\eta(\boldsymbol{\mu})) = \lambda \boldsymbol{C}(\boldsymbol{\mu}, \eta(\boldsymbol{\mu}))$ and $\eta(\boldsymbol{\mu})$ is an auxiliary variable, through the use of a penalty function. Using this approach, (13) is minimized by sequentially solving an unconstrained minimization problem of the form:

$$\min_{\zeta_r(\boldsymbol{\mu}), \eta(\boldsymbol{\mu})} E_{\text{reg}}(\zeta_r(\boldsymbol{\mu})) + \frac{1}{2\theta} \|\zeta_r(\boldsymbol{\mu}) - \eta(\boldsymbol{\mu})\|_2^2 + E_{\text{data}}(\eta(\boldsymbol{\mu})), \tag{14}$$

enforcing $\zeta_r(\boldsymbol{\mu}) = \eta(\boldsymbol{\mu})$ as $\theta \to 0$ and therefore $E(\zeta_r(\boldsymbol{\mu}), \eta(\boldsymbol{\mu})) \to E(\zeta_r(\boldsymbol{\mu}))$. This new energy functional allows us to split the minimization into two different problems that are alternately solved until convergence. The regularizer term and the decoupling term:

$$w(\boldsymbol{\mu}) \|\nabla \zeta_r(\boldsymbol{\mu})\|_\epsilon + \frac{1}{2\theta} \|\zeta_r(\boldsymbol{\mu}) - \eta(\boldsymbol{\mu})\|_2^2, \tag{15}$$

correspond to the TV-ROF denoising problem, defined in eq. (8). It is convex in $\zeta_r(\mu)$ and can be solved using a primal-dual algorithm. Doing an analogy with image denoising, $\eta(\boldsymbol{\mu})$ represents a noisy image whereas $\zeta_r(\boldsymbol{\mu})$ is the searched denoised result. The non convex in the auxiliary variable $\eta(\boldsymbol{\mu})$, $\frac{1}{2\theta} \|\zeta_r(\boldsymbol{\mu}) - \eta(\boldsymbol{\mu})\|_2^2 +$

$\lambda \boldsymbol{C}(\boldsymbol{\mu}, \eta(\boldsymbol{\mu}))$ is point-wise optimisable and the solution is exhaustively searched over a finite range of discretely sampled inverse depth values. The energy with the decoupling term of eq. (14) can be written as follows:

$$\min_{\boldsymbol{y},\boldsymbol{z}} \underbrace{\|AW\boldsymbol{y}\|_\epsilon}_{\text{regularizer term}} + \underbrace{\frac{1}{2\theta}\|\boldsymbol{y}-\boldsymbol{z}\|_2^2}_{\text{decoupling term}} + \underbrace{\lambda \boldsymbol{C}(\boldsymbol{z})}_{\text{data term}}, \quad (16)$$

where $A = \nabla$ is the gradient operator, W is the element-wise weighting matrix, $\boldsymbol{y}$ and $\boldsymbol{z}$ are row-wise vector versions of the sought solution $\zeta_r(\boldsymbol{\mu})$ and auxiliary variable $\eta(\boldsymbol{\mu})$ respectively. The regularizer and the decoupling term of eq. (16) for a fixed auxiliary variable $\boldsymbol{z}$ have the more general form:

$$\min_{\boldsymbol{y}} F(A\boldsymbol{y}) + G(\boldsymbol{y}). \quad (17)$$

In our case, the convex functions are $F(A\boldsymbol{y}) = \|AW\boldsymbol{y}\|_\epsilon$, $G(\boldsymbol{y}) = \frac{1}{2\theta}\|\boldsymbol{y}-\boldsymbol{z}\|_2^2$. We do a Legendre-Fenchel transformation:

$$F(A\boldsymbol{y}) = \max_{\boldsymbol{\varrho},\|\boldsymbol{\varrho}\|_2\leq 1} \langle AW\boldsymbol{y}, \boldsymbol{\varrho}\rangle - F^*(\boldsymbol{\varrho}), \quad (18)$$

where $\boldsymbol{\varrho}$ is the dual of $\boldsymbol{y}$ and $F^*(\boldsymbol{\varrho})$ is the conjugate of $F(A\boldsymbol{y})$:

$$F^*(\boldsymbol{\varrho}) = \delta(\boldsymbol{\varrho}) + \frac{\epsilon}{2}\|\boldsymbol{\varrho}\|_2^2, \quad (19)$$

$$\delta(\boldsymbol{\varrho}) = \begin{cases} 0 & \text{if } \|\boldsymbol{\varrho}\|_1 \leq 1. \\ \infty & \text{if otherwise} \end{cases} \quad (20)$$

Replacing $F(A\boldsymbol{y})$ and $G(\boldsymbol{y})$ in (17), we get the primal-dual formulation of this problem. It is a generic saddle-point problem:

$$\min_{\boldsymbol{y}} \max_{\boldsymbol{\varrho},\|\boldsymbol{\varrho}\|_2\leq 1} E = \langle AW\boldsymbol{y}, \boldsymbol{\varrho}\rangle + \frac{1}{2\theta}\|\boldsymbol{y}-\boldsymbol{z}\|_2^2 - \delta(\boldsymbol{\varrho}) - \frac{\epsilon}{2}\|\boldsymbol{\varrho}\|_2^2. \quad (21)$$

We do a step of projected gradient ascent (maximization problem) for the dual variable $\boldsymbol{\varrho}$ and one step of gradient descent (minimization problem) for the primal variable $\boldsymbol{y}$ (considering a fixed auxiliary variable $\boldsymbol{z}$), resulting the updates:

$$\begin{cases} \boldsymbol{\varrho}^{n+1} = \text{proj}_{\boldsymbol{\varrho}}((\boldsymbol{\varrho}^n + \sigma WA\boldsymbol{y}^n)/(1+\sigma\epsilon)), \\ \boldsymbol{y}^{n+1} = \frac{\boldsymbol{y}^n - \tau(WA^*\boldsymbol{\varrho}^{n+1} - \frac{1}{\theta^n}\boldsymbol{z}^n)}{1+\frac{\tau}{\theta^n}}, \end{cases} \quad (22)$$

where A^* is the adjoint operator of the gradient operator and corresponds to the negative divergence operator, σ and τ are the step size for the dual variable $\boldsymbol{\varrho}$ and primal variable $\boldsymbol{y}$ respectively, $\text{proj}_{\boldsymbol{\varrho}}(x) = x/max(1, \|x\|_2)$ projects the gradient ascent step back onto the constraint $\|\boldsymbol{\varrho}\|_1 \leq 1$. Finally, for a fixed (and updated) $\boldsymbol{y}$ we use a point-wise search to solve the remaining non-convex functional:

$$\arg\min_{\boldsymbol{z^{n+1}}} E^{aux} = \frac{1}{2\theta^n}(\boldsymbol{y}^{n+1} - \boldsymbol{z^{n+1}})^2 + \lambda \boldsymbol{C}(\boldsymbol{z^{n+1}}). \quad (23)$$

The primal-dual algorithm and the exhaustive search are alternated and θ^n is decreased in each step until convergence, it means, until $\theta^n \leq \theta_{th}$, where θ_{th} is a predefined threshold.

3.3 Tracking the Monocular Camera

For tracking the camera we use ORB-SLAM2 [11] with RGB-D inputs. It builds globally consistent sparse reconstructions for long-term trajectories with either monocular, stereo or RGB-D inputs, performing in real time on standard CPUs and including loop closure, map reuse and relocalization. This system has three main parallel threads: the tracking with motion only bundle adjustment (BA), the local mapping with local BA, and the loop closing with pose graph optimization and full BA. It does not fuse depth maps but uses ORB features for tracking, mapping and place recognition tasks. With BA and keyframes, it achieves more accuracy in localization than state-of-the-art methods based on ICP or photometric and depth error minimization. Place recognition, based on bag of words, is used for relocalization in case of tracking failure.

3.4 Shape Prior Estimation

In order to estimate the 3D model (shape prior) that best fits to depth data associated to an object of interest in the scene, we minimize the discrepancy between a model hypothesis and observed depth data back projected from the segmented region of the object. We evaluate the resulting point cloud in a 3D level-set embedding function that encodes the object model implicitly. This alignment consist in

reducing the distance of the points to the zero-level of the embedding function having as arguments the pose, scale and shape (latent variable), using Levenberg-Marquardt.

Initially, the 49 3D models are loaded in OpenGL using just geometric data. These models are aligned using ICP for getting models with the same position, orientation and scale. A volumetric structure with truncated signed distance function TSDF values is estimated and compressed, passing from 128^3 STDF values to 25^3 coefficients, using the discrete cosine transform DCT.

The Latent variable Model LVM is used for dimensionality reduction, to capture the shape variance as low dimensional latent shape spaces. The dimensionality reduction is applied to DCT coefficients such that the resulting latent variables have 2 dimensions instead of 25^3 dimensions of the original observed data (coefficients). We initialize the latent variables with the estimation got with Dual Probabilistic PCA. Then, we use the scaled conjugate gradient SCG algorithm for refining the initial estimation [10].

The mapping is modeled using a Gaussian process that defines areas where there are high certainty of getting a valid shape. Next, the latent variable that best fits to depth data is searched over a continuous space (no just the ones used for learning) and the coefficients associated to it are estimated.

The 3D level-set embedding function encoded in the coefficients is computed with the inverse discrete cosine transform IDCT. Besides shape optimization, the pose and scale of the 3D model are optimized in alternating way, using initially a coarse estimation of pose and scale, computed with depth data of the object and assuming that the car is over a flat surface. For model pose we use Lie algebra instead of Rodrigues notation for rotations.

3.5 Shape Prior Integration

Following a similar process for solving the minimization problem of the regularizer term and decoupling term of eq. (16), we minimize the energy:

$$\min_{\boldsymbol{D_f}} E(\boldsymbol{D_f}(\boldsymbol{\mu})) = \int (\underbrace{\|\nabla \boldsymbol{D_f}(\boldsymbol{\mu})\|_1}_{\text{regularizer term}} + \underbrace{\lambda \sum_{k=1}^{2} w_k(\boldsymbol{\mu}) \|\boldsymbol{D_f}(\boldsymbol{\mu}) - \boldsymbol{D_k}(\boldsymbol{\mu})\|_\epsilon}_{\text{data term}}) d\boldsymbol{\mu}, \tag{24}$$

where λ defines the balance between the regularizer term and the data term, $w_k(\boldsymbol{\mu}) \in \{0, 1\}$ defines the inpainting domain of the depth maps, with $w_k(\boldsymbol{\mu}) = 0$ for pure inpainting at location $\boldsymbol{\mu}$, $\|\cdot\|_\epsilon$ is the Huber norm defined in eq. (10), $\boldsymbol{D_1}(\boldsymbol{\mu}) = \boldsymbol{D_s}(\boldsymbol{\mu})$ is the refined depth coming from the monocular camera, $\boldsymbol{D_2}(\boldsymbol{\mu}) = \boldsymbol{D_m}(\boldsymbol{\mu})$ is the depth coming from the optimal 3D model (shape prior) and $\boldsymbol{D_f}(\boldsymbol{\mu})$ is the sought solution. We express eq. (24) in a more general form:

$$\min_{\boldsymbol{y}} F(A\boldsymbol{y}) + \sum_{k=1}^{2} G_k(\boldsymbol{y}), \tag{25}$$

where $A = \nabla$ the gradient operator, $F(A\boldsymbol{y}) = \|A\boldsymbol{y}\|_1$, $G_k(\boldsymbol{y}) = \lambda \boldsymbol{\varpi_k} \|\boldsymbol{y} - \boldsymbol{\varphi_k}\|_\epsilon$, $\boldsymbol{y}$, $\boldsymbol{\varphi_k}$ and $\boldsymbol{\varpi_k}$ are row-wise vector versions of the sought solution $\boldsymbol{D_f}(\boldsymbol{\mu})$, the depth sources $\boldsymbol{D_k}(\boldsymbol{\mu})$, and the matrix w_k that defines the inpainting domain, respectively. With Legendre-Fenchel transformations we get:

$$\min_{\boldsymbol{y}} \max_{\|\boldsymbol{r_k}\|_1 \leq \lambda \boldsymbol{\varpi_k}, \|\boldsymbol{\varrho}\|_2 \leq 1} \langle A\boldsymbol{y}, \boldsymbol{\varrho} \rangle - F^*(\boldsymbol{\varrho}) + \sum_{k=1}^{2} [\langle \boldsymbol{y} - \boldsymbol{\varphi_k}, \boldsymbol{r_k} \rangle - {G_k}^*(\boldsymbol{r_k})], \tag{26}$$

where $\boldsymbol{\varrho}$ and $\boldsymbol{r_k}$ are dual variables associated to the primal variables $\boldsymbol{y}$ and $\boldsymbol{\varphi_k}$ respectively, $F^*(\boldsymbol{\varrho})$ and ${G_k}^*(\boldsymbol{r_k})$ are the convex conjugates of $F(A\boldsymbol{y})$ and $G_k(\boldsymbol{y})$, respectively, and are defined as:

$$F^*(\boldsymbol{\varrho}) = \delta_{\boldsymbol{\varrho}}(\boldsymbol{\varrho}), \tag{27}$$

$${G_k}^*(\boldsymbol{r_k}) = \delta_{\boldsymbol{r_k}}(\boldsymbol{r_k}) + \frac{\epsilon}{2} \|\boldsymbol{r_k}\|_2^2, \tag{28}$$

$\delta_{\boldsymbol{\varrho}}$ and $\delta_{\boldsymbol{r_k}}$ are indicator functions of the convex sets, defined as:

$$\delta_{\boldsymbol{\varrho}}(\boldsymbol{\varrho}) = \begin{cases} 0 & \text{if } \|\boldsymbol{\varrho}\|_1 \leq 1, \\ \infty & \text{otherwise.} \end{cases} \tag{29}$$

$$\delta_{\boldsymbol{r_k}}(\boldsymbol{r_k}) = \begin{cases} 0 & \text{if } \|\boldsymbol{r_k}\|_1 \leq \lambda \boldsymbol{\varpi_k}. \\ \infty & \text{otherwise.} \end{cases} \tag{30}$$

We propose to use the scheme of [8] (where color images for aerial applications are merged, denoised and inpainted) for merging depth data of two sources, getting an enhanced depth map. The iterative optimization corresponds to perform in alternating way gradient ascent over the dual variables and gradient descent over the primal variable $\boldsymbol{y}$, projecting the results onto the constraints and updating the primal variable, as is summarized next:

$$\begin{cases} \boldsymbol{\varrho}^{n+1} = \text{proj}_{\boldsymbol{\varrho}}(\boldsymbol{\varrho}^n + \sigma A \bar{\boldsymbol{y}}^n), \\ \boldsymbol{r_k}^{n+1} = \text{proj}_{\boldsymbol{r_k}}\left(\frac{\boldsymbol{r_k}^n + \sigma(\bar{\boldsymbol{y}}^n - \boldsymbol{\varphi_k})}{1+\sigma\epsilon}\right), \quad k = 1, 2., \\ \boldsymbol{y}^{n+1} = \boldsymbol{y}^n - \tau(A^* \boldsymbol{\varrho}^{n+1} + \sum_{k=1}^{2} \boldsymbol{r_k}^{n+1}), \\ \bar{\boldsymbol{y}}^{n+1} = \boldsymbol{y}^{n+1} + \Phi(\boldsymbol{y}^{n+1} - \boldsymbol{y}^n), \end{cases} \tag{31}$$

where A^* is the adjoint operator of the gradient operator and corresponds to the negative divergence operator, $\Phi = 1$, $\text{proj}_{\boldsymbol{\varrho}}$ and $\text{proj}_{\boldsymbol{r_k}}$ are projections of the dual variables $\boldsymbol{\varrho}$ and $\boldsymbol{r_k}$, respectively, onto convex sets. They are defined for each element of the vectors as:

$$\text{proj}_{\boldsymbol{\varrho}}(\tilde{\boldsymbol{\varrho}}) = \frac{\tilde{\boldsymbol{\varrho}}}{\max(1, \|\tilde{\boldsymbol{\varrho}}\|_1)}, \tag{32}$$

$$\text{proj}_{\boldsymbol{r_k}}(\tilde{\boldsymbol{r}}_{\boldsymbol{k}}) = \begin{cases} \tilde{\boldsymbol{r}}_{\boldsymbol{k}} & \text{if } \|\tilde{\boldsymbol{r}}_{\boldsymbol{k}}\|_1 < \lambda \boldsymbol{\varpi_k}, \\ \lambda \boldsymbol{\varpi_k} & \text{if } \tilde{\boldsymbol{r}}_{\boldsymbol{k}} > \lambda \boldsymbol{\varpi_k}, \\ -\lambda \boldsymbol{\varpi_k} & \text{if } \tilde{\boldsymbol{r}}_{\boldsymbol{k}} < -\lambda \boldsymbol{\varpi_k}. \end{cases} \tag{33}$$

Following the algorithm 1 of [1] and the parameter setting of [8] we set the primal and dual time steps with $\tau = 0.05$, $\sigma = 1/(8\tau)$, the Huber norm parameter $\epsilon = ...$, and $\lambda = ...$ We set the initial primal variable as $\bar{\boldsymbol{y}}_0 = \boldsymbol{\varphi_s}$ since it is the most informative depth source. The dual variables $\boldsymbol{\varrho}$ and $\boldsymbol{r_k}$ are initialized with zeros.

3.6 Putting together Depth Map Estimation and Shape Prior Integration

The algorithms previously described for building a depth map with a monocular camera and for integrating the shape prior are based on variational techniques that are solved with the primal-dual algorithm so they share common modules. Moreover, the object of interest is static and rigid so its pose, scale and shape do not change with time. We exploit these facts for implementing one algorithm that takes a shape prior estimated previously (for example in a previous keyframe) and makes simultaneously depth map building and shape prior integration. Now, we integrate shape prior data into the energy functional (9) with an additional term, the shape prior term:

$$E(\boldsymbol{\mu}, \zeta_r(\boldsymbol{\mu})) = \int (\underbrace{w(\boldsymbol{\mu})\|\nabla\zeta_r(\boldsymbol{\mu})\|_\epsilon}_{\text{regularizer term}} + \underbrace{\lambda_m w_m(\boldsymbol{\mu})\|\zeta_r(\boldsymbol{\mu}) - \zeta_m(\boldsymbol{\mu})\|_{\epsilon_m}}_{\text{shape prior term}} + \underbrace{\lambda \boldsymbol{C}(\boldsymbol{\mu}, \zeta_r(\boldsymbol{\mu}))}_{\text{data term}}) d\boldsymbol{\mu}, \tag{34}$$

where λ_m is a balance factor for the shape prior term, $w_m \in \{0, 1\}$ defines the inpainting domain of the inverse depth map coming from the model $\zeta_m(\mu)$. The shape prior term forces the solution $\zeta_r(\boldsymbol{\mu})$ to be similar to the shape prior $\zeta_m(\boldsymbol{\mu})$. Using the decoupling approach and discretizing the energy we have:

$$\min_{\boldsymbol{y},\boldsymbol{z}} \underbrace{\|AW\boldsymbol{y}\|_\epsilon}_{\text{regularizer term}} + \underbrace{\lambda_m \boldsymbol{\varpi_m}\|(\boldsymbol{y} - \boldsymbol{\varphi_m})\|_{\epsilon_m}}_{\text{shape prior term}} + \underbrace{\frac{1}{2\theta}\|\boldsymbol{y} - \boldsymbol{z}\|_2^2}_{\text{decoupling term}} + \underbrace{\lambda \boldsymbol{C}(\boldsymbol{z})}_{\text{data term}}, \tag{35}$$

where $A = \nabla$ is the gradient operator, W is the element-wise weighting matrix, $\boldsymbol{y}$, $\boldsymbol{\varphi_m}$, $\boldsymbol{z}$ and $\boldsymbol{\varpi_m}$ are row-wise vector versions of the sought solution $\zeta_r(\boldsymbol{\mu})$, the inverse depth map of the model $\zeta_m(\boldsymbol{\mu})$, the auxiliary variable η and the inpainting domain of the inverse depth map of the model $w_m(\boldsymbol{\mu})$, respectively. For a fixed auxiliary variable $\boldsymbol{z}$, the regularization term and the shape prior term have the general form of eq. (17), where $F(A\boldsymbol{y}) = \|AW\boldsymbol{y}\|_\epsilon$, $G_k(\boldsymbol{y}) = \lambda\boldsymbol{\varpi_k}\|\boldsymbol{y} - \boldsymbol{\varphi_m}\|_\epsilon$. With Legendre-Fenchel transformations, we get:

$$\min_{\boldsymbol{y}} \max_{\|\boldsymbol{r_k}\|_1 \leq \lambda\boldsymbol{\varpi_k}, \|\boldsymbol{\varrho}\|_2 \leq 1} \langle AW\boldsymbol{y}, \boldsymbol{\varrho}\rangle - F^*(\boldsymbol{\varrho}) + \langle \boldsymbol{y} - \boldsymbol{\varphi_m}, \boldsymbol{r_m}\rangle - {G_k}^*(\boldsymbol{r_k}) + \frac{1}{2\theta}\|\boldsymbol{y} - \boldsymbol{z}\|_2^2, \tag{36}$$

where $\boldsymbol{\varrho}$ and $\boldsymbol{r_k}$ are dual variables associated to the primal variables $\boldsymbol{y}$ and $\boldsymbol{\varphi_m}$ respectively. $F^*(\boldsymbol{\varrho})$

ISSN 2007-9737

and $G_k{}^*(\boldsymbol{r_k})$ are the convex conjugates of $F(A\boldsymbol{y})$ and $G_k(\boldsymbol{y})$, respectively. They are defined as:

$$F^*(\boldsymbol{\varrho}) = \delta_\varrho(\boldsymbol{\varrho}) + \frac{\epsilon}{2}\|\boldsymbol{\varrho}\|_2^2, \tag{37}$$

$$G_k{}^*(\boldsymbol{r_k}) = \delta_{\boldsymbol{r_k}}(\boldsymbol{r_k}) + \frac{\epsilon_m}{2}\|\boldsymbol{r_k}\|_2^2, \tag{38}$$

$\delta_{\boldsymbol{\varrho}}$ and $\delta_{\boldsymbol{r_k}}$ are the same as the ones defined in eq. (29) and (30). The parameter setting is similar to the one used for denoising, inpainting and integration described previously in section 3.5. Considering the auxiliary variable $\boldsymbol{z}$ fixed, the gradient ascent and gradient descent steps of the primal dual algorithm are:

$$\begin{cases} \boldsymbol{\varrho}^{n+1} = \text{proj}_{\boldsymbol{\varrho}}(\frac{\boldsymbol{\varrho}^n + \sigma W A \bar{\boldsymbol{y}}^n}{1+\epsilon\sigma}), \\ \boldsymbol{r_m}^{n+1} = \text{proj}_{\boldsymbol{r_m}}\left(\frac{\boldsymbol{r_m}^n + \sigma(\bar{\boldsymbol{y}}^n - \boldsymbol{\varphi_m})}{1+\sigma\epsilon_m}\right), \\ \boldsymbol{y}^{n+1} = \frac{\boldsymbol{y}^n - \tau(WA^*\boldsymbol{\varrho}^{n+1} + \boldsymbol{r_m}^{n+1} - \frac{1}{\theta^n}z^n)}{1+\frac{\tau}{\theta^n}}, \\ \bar{\boldsymbol{y}}^{n+1} = \boldsymbol{y}^{n+1} + \Phi(\boldsymbol{y}^{n+1} - \boldsymbol{y}^n), \end{cases} \tag{39}$$

where A^* is the negative divergence operator, σ and τ are the step size for the dual variables $\boldsymbol{\varrho}$, $\boldsymbol{r_m}$, and for the primal variable $\boldsymbol{y}$ respectively, $\Phi = 1$, $\text{proj}_{\boldsymbol{\varrho}}$ and $\text{proj}_{\boldsymbol{r_m}}$ are projections of the dual variables $\boldsymbol{\varrho}$ and $\boldsymbol{r_m}$, respectively, onto convex sets. They were defined in eq. (32) and (33). Finally, for a fixed (and updated) $\boldsymbol{y}$ we use a point-wise search to solve the remaining non-convex functional defined in eq. (23). The primal-dual algorithm and the exhaustive search are alternated and θ^n is decreased as was done for building the depth map in section 3.2.

4 Results

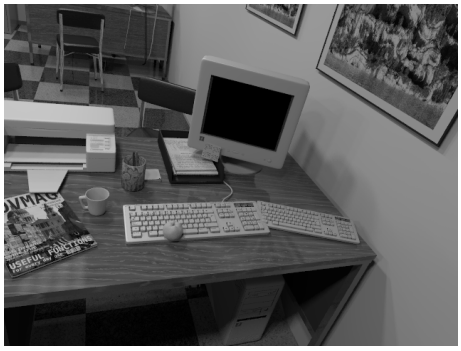
We carry out three experiments: one with synthetic data for computing the accuracy of the estimated depht map and the other ones with real data for enhancing the created depth map with shape priors and variational methods. For the first experiment we use 40 images ($n = 40$) and one reference image from the dataset of Ankur Handa [6]. The depth is in the range $[0.5 \quad 5]$m that corresponds to an inverse depth range of $[2 \quad 0.2]\text{m}^{-1}$. The number of samples (linear sampling in inverse depth) is $n_s = 100$, the balance λ is 1, the threshold for the Huber norm ϵ is 0.01, α and β of eq. (11) are 0.4 and 2.4 respectively. Figures 3(a) and 3(b) show the image of reference in gray and the ground truth in depth. Figures 3(c) and 3(d) show the initial depth map obtained by minimizing eq. (7) and the refined depth map after 200 iterations, respectively. The mean error in depth diminishes from 0.1685m (standard deviation of 0.4483) to 0.0953m (standard deviation of 0.2397) when the regularization term is used in the optimization problem. Figure 3(e) shows the absolute error for the solution (using both the data term and the regularization term) while fig. 3(f) compares the depth for row 240 and its ground truth. Note that the solutions is smooth but preserves discontinuities.

For the second and third experiments the kinect 1.0 is employed: the RGB images for building a depth map and the depth map coming from the sensor as reference for estimating the accuracy of the solution. We use 40 images ($n = 40$) and one reference image. The range of depth is $[0.3 \quad 1.5]$m that corresponds to an inverse depth range of $D = [3.33 \quad 0.66]\text{m}^{-1}$. The number of samples is $n_s = 100$, the balance λ is 0.7, the threshold for the Huber norm ϵ is 0.01, α and β of eq. (11) are 0.4 and 2.4 respectively.

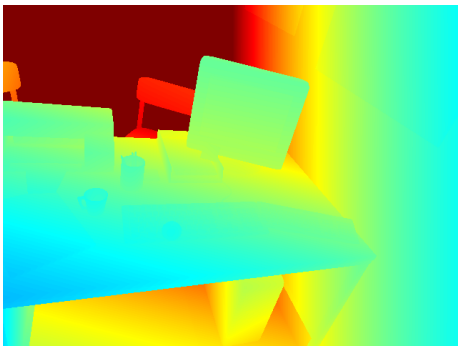
Figures 4(a) and 4(b) show the image of reference in gray and the depth coming from the sensor. Figures 4(c) and 4(d) show the initial depth map obtained by minimizing eq. (7) and the refined depth map after 200 iterations, respectively. The mean error in depth diminishes from 0.0841m (standard deviation of 0.1161) to 0.0474m (standard deviation of 0.0964) when the regularization term is used in the optimization problem. Figure 4(e) shows the absolute error for the solution (using both the data term and the regularization term) while fig. 4(f) compares the depth for row 210 and its ground truth.

Next, we manually segment the car (see fig 5(a)) and compute a point cloud (red points in fig. 5(b)) with the depth data of the built depth map in the segmented area. This point cloud is aligned with a 3D model by minimizing an energy function w.r.t. pose, scale and shape. For an initial estimation of position we use the centroid of the point cloud, adding 4% of the x component to itself since the data belongs just to a side of the whole car.

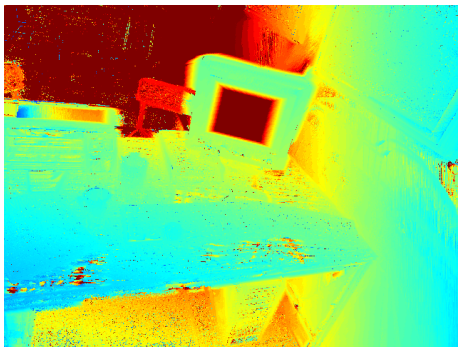
ISSN 2007-9737



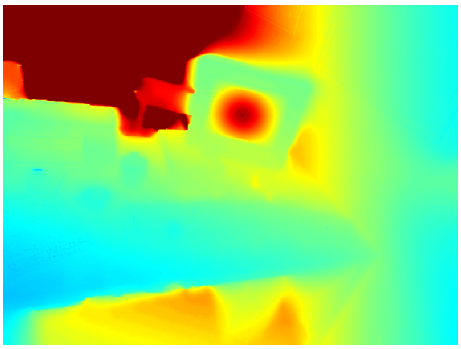
(a) Gray image from reference frame $I_r(\mu)$



(b) Ground truth of the depth map from reference frame $D_r(\mu)$

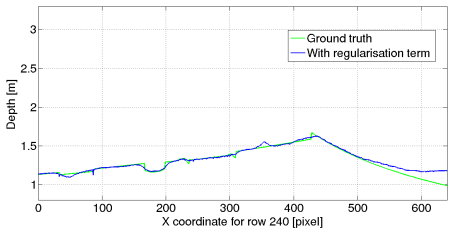


(c) Initial Depth map. It is noisy due to regions of low texture.



(d) Depth map after 200 iterations of primal-dual algorithm.

(e) Absolute error in depth using the data term and regularization term.

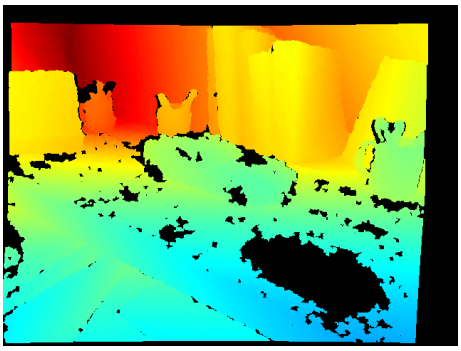


(f) Depth for row 240 using the data term and regularization term.

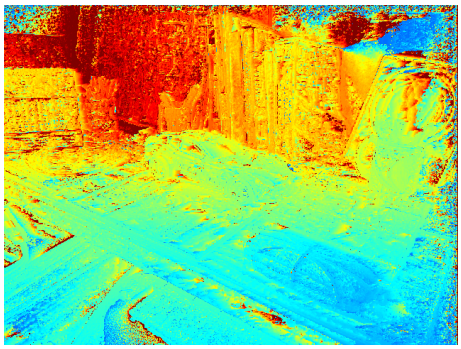
Fig. 3. (a) Gray image from reference frame, (b) ground truth of the depth map from reference frame, (c) initial depth map built using the photometric error, (d) refined depth map after 200 iterations of primal-dual algorithm, (e) absolute error of the solution and (f) depth for row 240 compared to ground truth

For the scale we consider the average distance of each point to the centroid. We suppose a supporting plane (blue points in fig. 5(b)) in order to estimate two of the three angles that define the

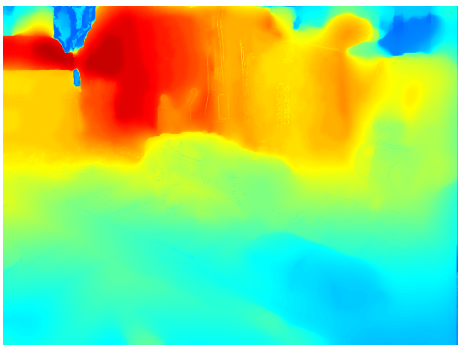
(a) Gray image for the reference frame $I_r(\mu)$, real data.



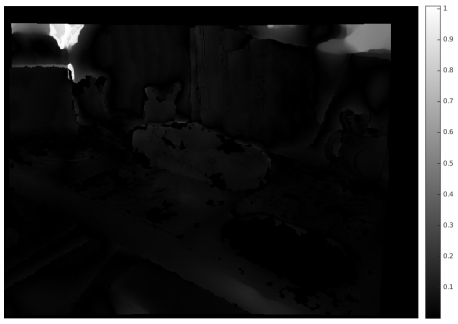
(b) Depth from kinect $D_r(\mu)$. Used for estimating accuracy of solution.



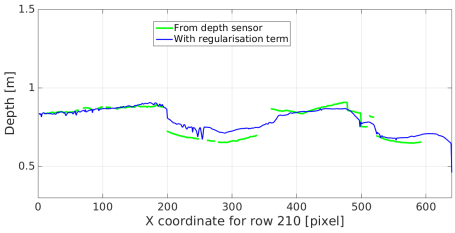
(c) Initial Depth map. It is noisy due to regions of low texture.



(d) Depth map after 200 iterations of primal-dual algorithm.



(e) Absolute error in depth using the data term and regularization term.



(f) Depth for row 210 using the data term and regularization term.

Fig. 4. (a) Gray image from reference frame (real data), (b) depth got with kinect sensor for reference frame (used for estimating depth accuracy of solution), (c) initial depth map built using the photometric error, (d) refined depth map after 200 iterations of primal-dual algorithm, (e) absolute error of the solution and (f) depth for row 210 compared to ground truth

initial orientation of the car. The third angle is found with exhaustive search. Summarizing, the initial position is $\boldsymbol{t_{wo}} = [0.7433 \quad 0.0065 \quad 0.0264]$, the initial orientation corresponds to (rotations over fixed axis) $\alpha_x = 14.3506°$, $\alpha_y = -17.8619°$, and $\alpha_z = -31.6360°$. The initial scale is $s = 0.2222$, and the initial shape is the one associated to the reference model employed in the model alignment process $\boldsymbol{\chi} = [0.3895 \quad 1.6162]$. Figure 5(c) shows the initial conditions for the model.

For refining the initial estimation we carry out two cycles with the sequence: 15 iterations for pose and 5 iterations for scale. At the end of this sequence, 40 iterations have been done and very close pose and scale estimations are obtained (see fig. 5(d)). With these estimations we can perform exhaustive search over the $N_m = 49$ models of cars used for learning the latent space. The latent variable with the 3D level set that produces the minimum energy $\boldsymbol{\chi} = [-0.0659 \quad 0.2060]$ is used as initial value in the following refinement process. Finally, we carry out three cycles with the sequence: 10 iterations for shape, 10 iterations for pose and 5 iterations for scale, getting a refinement in pose and scale for a more approximated shape (see fig. 5(e)). The final scale is $s = 0.2325$, and the final latent variable is $\boldsymbol{\chi} = [-0.0763 \quad 0.1037]$.

The final position is $\boldsymbol{t_{wo}} = [0.7614 \quad 0.0173 \quad 0.0393]$ and the final orientation is $\alpha_x = 13.3060°$, $\alpha_y = -19.9214°$ and $\alpha_z = -37.1394°$. Finally, we create a synthetic depth map by reading the depth buffer from the current camera pose (see fig. 5(f)). The evolution of the energy for this alternating optimization is shown in fig. 6.

Once we have two depth maps: one from the optimal 3D model and the other one built with a monocular camera, we integrate this data for getting an enhanced depth map. Figure 7(a) shows the built depth map using algorithm of eq. (22) while fig. 7(b) shows the resulting depth map after 100 iterations of the algorithm for merging shape prior data of eq. (31).

Note that the most significant changes are presented in the car area where the depth map built with the monocular camera and the depth map from the optimal 3D model interact and integrate. Outside the car area just depth smoothing is carried out. Figure 7(c) compares depth data through the x-slice for row 210. In this figure we can see the smoothing effect and the improvement in accuracy in the car area.

The alternative approach, that makes simultaneously depth map building and shape prior integration (supposing that we already have a shape prior), produces similar results than running both algorithms sequentially.

Figure 8(a) shows the initial depth map got by solving the data term (see eq. (7)), while fig. 8(b) shows the results of the simultaneous depth refinement and shape prior integration defined in eq. (39). In fig. 8(c) we can see a comparison of depth data through the x-slice for row 210.

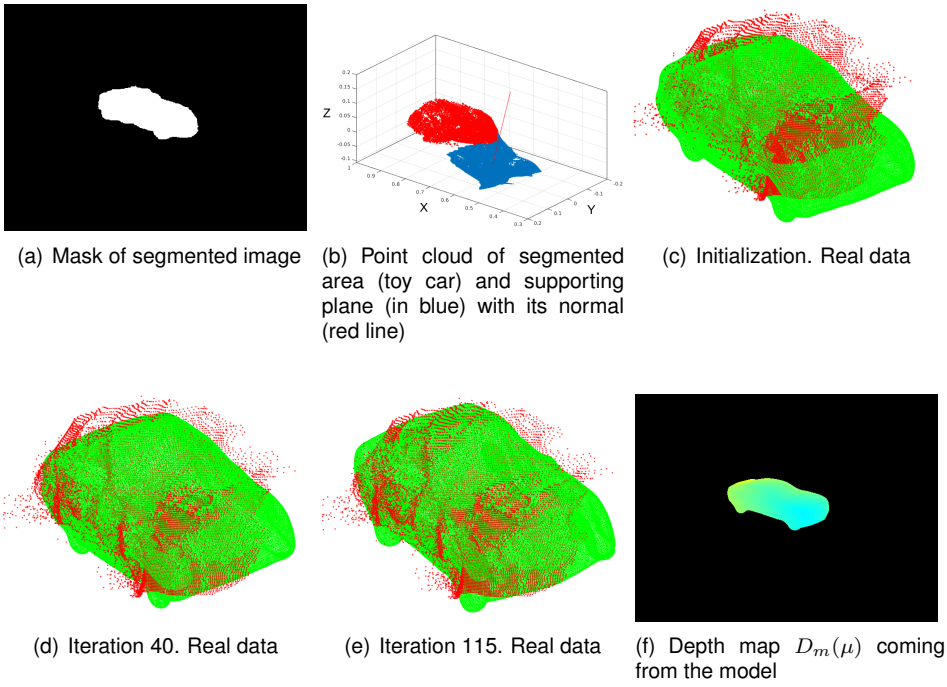
For quantifying these improvements we compute the error in depth of the built depth map, of the depth map estimated with a sequential building and shape prior integration and of the depth map estimated with a simultaneous refinement and shape prior integration, considering only the segmented area (car area) and taking the depth coming from the sensor as reference. The comparison is summarized in table 1.

We found that the RMSE, the mean error and the median error diminish when shape prior data is integrated in a sequential or simultaneous way, although the maximum values (in both algorithms) and the standard deviation (in the simultaneous algorithm) increase a little bit due to mismatches in the borders of the car. Moreover, comparing these values we can say that the simultaneous algorithm produces depth maps with higher accuracy in the car area than the sequential one.

Finally, we present the processing time for the main steps of the three algorithms analyzed in this work: depth map building, shape prior integration and simultaneous depth map building and shape prior integration.

Since they share the same structure, with the primal-dual algorithm for solving the optimization problems, the processes are similar, as is shown in table 2.

An iteration for building a depth map takes 18.452ms so the computation of the initial depth map and 200 iterations for refining it takes 4.7529s. An iteration for integrating the shape prior data into



(a) Mask of segmented image

(b) Point cloud of segmented area (toy car) and supporting plane (in blue) with its normal (red line)

(c) Initialization. Real data

(d) Iteration 40. Real data

(e) Iteration 115. Real data

(f) Depth map $D_m(\mu)$ coming from the model

Fig. 5. Steps in the process for estimating the optimal 3D model and a synthetic depth map from the current camera pose. (a) Mask of the segmented car, (b) Point cloud of the segmented car and supporting plane, (c) Initial state of the 3D model, (d) 3D model for iteration 40, (e) 3D model for iteration 115 and (f) depth map from the model

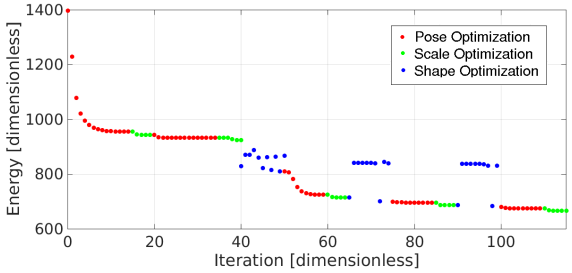


Fig. 6. Evolution of the energy for pose, scale and shape optimization

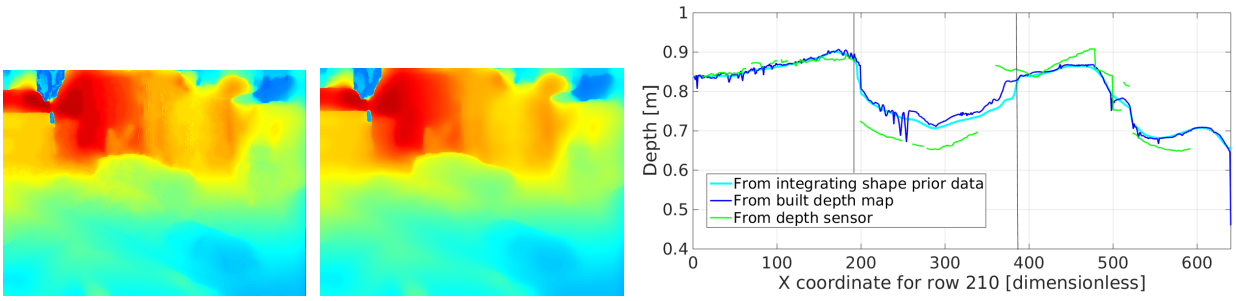
the built depth map takes 15.694ms so the total time for 100 iterations is 1.5694s.

The time for building and merging shape prior data sequentially is 6.3223s. On the other hand, an iteration of the algorithm that makes simultaneously depth refinement and shape prior integration takes 25.287ms. Considering the time for estimating the initial depth map and 200 iterations of the simultaneous algorithm, the resulting depth map takes 6.1199s.

5 Conclusion

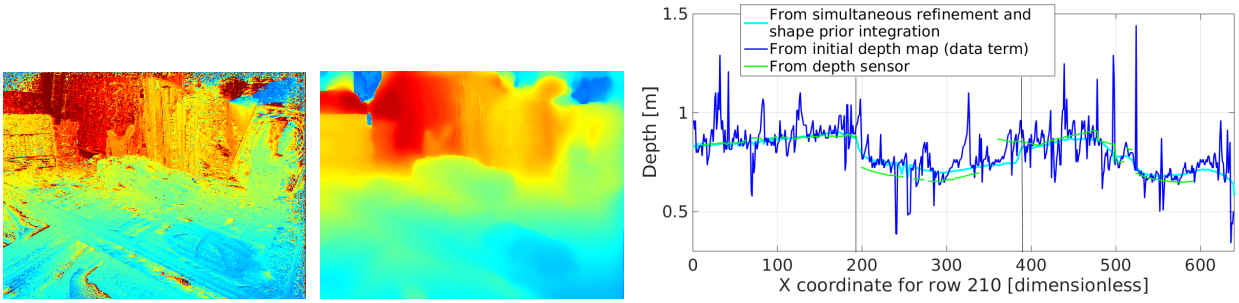
We have developed a system that builds a depth map with a monocular camera and integrates shape prior data, in sequential and simultaneous way, for improving its accuracy. The depth map is built by minimizing an energy functional, composed of a data term and a regularization term, using a decoupling term, the primal-dual algorithm and exhaustive search. The models are represented as 3D level-sets that are compressed and reduced in dimensions for improving the search of the optimal model. The energy function aligns a point cloud of the segmented area associated to the car and the level-set embedding function of a model

ISSN 2007-9737



(a) Built depth map $D_s(\mu)$ (b) Sought depth map $D_f(\mu)$ for iteration 100 (c) Depth for row 210. The vertical lines encompass the car

Fig. 7. (a) Built depth map using a monocular camera, (b) Resulting depth map after 100 iterations for merging the built depth map and the shape prior data, (c) Depth for row 210; data of the built depth map (dark blue), the smoother and complete solution resulting from merging shape prior data (cyan) and the incomplete depth coming from the sensor (green)



(a) Initial depth map computed with eq. (7) (b) Depth map after 200 iterations of the algorithm for simultaneous depth refinement and shape prior integration (c) Depth for row 210. The vertical lines encompass the car

Fig. 8. (a) Initial depth map computed by solving eq. (7), (b) Resulting depth map after 200 iterations of the algorithm for refining and shape prior integration, (c) Depth for row 210; data of the built depth map (dark blue), the smoother and complete solution resulting from merging shape prior data (cyan) and the incomplete depth coming from the sensor (green)

hypothesis. The optimization is done w.r.t. pose, scale and shape. Once the alignment is done, a synthetic depth map coming from the optimal model is created and integrated to the built depth map (sequential way).

In the simultaneous way, the energy functional for building a depth map is modified by adding a term that constraints the solution to be similar to the synthetic depth map coming from the shape prior. Finally, the improvement in accuracy is quantified. The results are satisfactory:

1. The mean error of the depth map created with a monocular camera (using a synthetic depth map as reference) is 0.0953m, when both the data term and the regularization term are used in the optimization.
2. When the shape prior is integrated into the built depth map the mean error in the

Table 1. Comparison in accuracy in depth in the car area using the depth map built with the monocular camera and the depth map resulting of merging the shape prior data

	Built depth map	With shape prior data	built and shape prior
RMSE [m]	*0.0624*	*0.0572*	*0.0531*
Max. error [m]	0.2038	0.2195	0.2316
Min. error [m]	3.4114e-04	1.0117e-04	5.7745e-05
Mean error [m]	0.0596	0.0545	0.0487
Median error [m]	0.0582	0.0537	0.0452
Standard dev. [m]	0.0187	0.0176	0.0212

Table 2. Processing time for depth map building DMB, shape prior integration SPI and both depth map building and shape prior integration simultaneously DMB-SPI

Process-Algorithm	Time[ms] DMB	Time[ms] SPI	Time[ms] DMB-SPI
Creation initial depth map			
Eq. (7)	1062.5	—	1062.5
Update of ϱ			
First line of eq. (22)	6.627		
First line of eq. (31)		6.501	
First line of eq. (39)			6.648
Update of r			
Second line of eq. (31)	—	5.373	
Second line of eq. (39)	—		5.415
Update of y			
Second line of eq. (22)	3.350		
Third line of eq. (31)		3.247	
Third line of eq. (39)			3.366
Update of z			
Eq. (23)	6.293	—	6.352
Remaining processes	2.182	0.573	3.506
TOTAL ITERATION	18.452	15.694	25.287

segmented area diminishes from 0.0596m to 0.0545m for the sequential algorithm and from 0.0596m to 0.0487m for the simultaneous algorithm, and the data for both cases is smoother, closer to the object's shape and preserves discontinuities.

The processing time in commodity graphics hardware is 6.3223s for the sequential algorithm, with 200 iterations for building a depth map and then 100 iterations for merging shape prior data. The processing time is 6.1199s for the simultaneous algorithms with 200 iterations for building a depth map and merging shape prior data at the same time (times do not consider the estimation of the shape prior).

As future work we leave the implementation of the algorithm for estimating the optimal model (shape prior) in commodity graphics hardware. Moreover, we leave as future work the fusion of several enhanced depth maps into a volumetric structure in order to make a dense reconstruction of the scene and quantify the improvement in geometry of the reconstructed 3D model.

Acknowledgment

The authors would like to thank Fundación CEIBA for the financial support that has made the development of this work possible.

References

1. **Chambolle, A. & Pock, T. (2011).** A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, Vol. 1, No. 40, pp. 120–145.

2. **Concha, A., Hussain, M. W., Montano, L., & Civera, J. (2014).** Manhattan and piecewise-planar constraints for dense monocular mapping. *Robotics: Science and Systems X, University of California, Berkeley, USA, July 12-16, 2014*.
3. **Concha, A., Hussain, W., Montano, L., & Civera, J. (2015).** Incorporating scene priors to dense monocular mapping. *Autonomous Robots*, Vol. 39, No. 3, pp. 279–292.
4. **Dame, A., Prisacariu, V. A., Ren, C. Y., & Reid, I. (2013).** Dense reconstruction using 3D object shape priors. *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1288–1295.
5. **Felzenszwalb, P. F., Girshick, R. B., McAllester, D., & Ramanan, D. (2010).** Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 32, No. 9, pp. 1627–1645.
6. **Handa, A., Newcombe, R. A., Angeli, A., & Davison, A. J. (2012).** Real-time camera tracking: When is high frame-rate best? *Proc. of the European Conference on Computer Vision (ECCV)*.
7. **Klein, G. & Murray, D. W. (2007).** Parallel tracking and mapping for small AR workspaces. *Proc. Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality*.
8. **Kluckner, S., Pock, T., & Bischof, H. (2010).** Exploiting redundancy for aerial image fusion using convex optimization. In **Goesele, M., Roth, S., Kuijper, A., Schiele, B., & Schindler, K.**, editors, *Pattern Recognition: 32nd DAGM Symposium, Darmstadt, Germany*. Springer Berlin Heidelberg, pp. 303–312.
9. **Krenzin, J. & Hellwich, O. (2016).** *Reduction of Point Cloud Artifacts Using Shape Priors Estimated with the Gaussian Process Latent Variable Model*. Springer International Publishing, Cham, pp. 273–284.
10. **Muller, M. F. (1993).** A scaled conjugate gradient algorithm for fast supervised learning. *Neural networks*, Vol. 6, No. 4, pp. 525–533.
11. **Mur-Artal, R. & Tardós, J. D. (2017).** ORB-SLAM2: an open-source SLAM system for monocular, stereo and RGB-D cameras. *IEEE Transactions on Robotics*, Vol. 33, No. 5, pp. 1255–1262.
12. **Newcombe, R. A., Lovegrove, S. J., & Davison, A. J. (2012).** DTAM: Dense tracking and mapping in real-time. , No. Department of Computing, Imperial College London, UK.
13. **Piniés, P., Paz, L. M., & Newman, P. (2015).** Dense mono reconstruction: Living with the pain of the plain plane. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Seattle, WA, USA.
14. **Prisacariu, V. A., Segal, A. V., & Reid, I. (2013).** Simultaneous monocular 2D segmentation, 3D pose recovery and 3D reconstruction. *Proceedings of the 11th Asian Conference on Computer Vision - Volume Part I*, ACCV'12, Springer-Verlag, Berlin, Heidelberg, pp. 593–606.
15. **Rudin, L. I., Osher, S., & Fatemi, E. (1992).** Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, Vol. 60, No. 1-4, pp. 259–268.
16. **Slesareva, N., Bruhn, A., & Weickert, J. (2005).** Optic flow goes stereo: A variational method for estimating discontinuitypreserving dense disparity maps. *Proc. 27th DAGM Symposium*, pp. 33–40.
17. **Steinbruecker, F., Pock, T., & Cremers, D. (2009).** Large displacement optical flow computation without warping. *Proc. Int. Conf. Computer Vision*, Kyoto, Japan.
18. **Stuhmer, J., Gumhold, S., & Cremers, D. (2010).** Real-time dense geometry from a handheld camera. *Proceedings of the DAGM Symposium on Pattern Recognition*, pp. 11–20.
19. **Wu, C. (2011).** VisualSFM: A visual structure from motion system. Accessed 2017-05-25.
20. **Zach, C., Pock, T., & Bischof, H. (2007).** A duality based approach for realtime TV-l1 optical flow. *Ann. Symp. German Association Patt. Recogn*, pp. 214–223.
21. **Zhu, M. (2008).** *Fast Numerical Algorithms for Total Variation Based Image Restoration*. Doctoral thesis, University of California.

Article received on 20/09/2018; accepted on 11/11/2019.
Corresponding author is Andrés Díaz.