AUTOCALIBRATION OF LIDAR AND OPTICAL CAMERAS VIA EDGE ALIGNMENT

*Juan Castorena**

New Mexico State University
Klipsch School of Electrical Engineering
Las Cruces, NM 88003
jecastor@nmsu.edu

Ulugbek S. Kamilov, Petros T. Boufounos

Mitsubishi Electric Research Laboratories
201 Broadway
Cambridge, MA 02139
{*kamilov, petrosb*}*@merl.com*

ABSTRACT

We present a new method for joint automatic extrinsic calibration and sensor fusion for a multimodal sensor system comprising a LIDAR and an optical camera. Our approach exploits the natural alignment of depth and intensity edges when the calibration parameters are correct. Thus, in contrast to a number of existing approaches, we do not require the presence or identification of known alignment targets. On the other hand, the characteristics of each sensor modality, such as sampling pattern and information measured, are significantly different, making direct edge alignment difficult. To overcome this difficulty, we jointly fuse the data and estimate the calibration parameters. In particular, the joint processing evaluates and optimizes both the quality of edge alignment and the performance of the fusion algorithm using a common cost function on the output. We demonstrate accurate calibration in practical configurations in which depth measurements are sparse and contain no reflectivity information. Experiments on synthetic and real data obtained with a three-dimensional LIDAR sensor demonstrate the effectiveness of our approach.

***Index Terms*—** Multimodal calibration, depth superresolution, intersensor registration, sensor fusion, total variation.

1. INTRODUCTION

As an increasing number of sensors and sensor modalities are used to acquire scenes, consolidation or fusion of the sensor data is becoming increasingly important. Data fusion exploits the distinct sensor modalities to provide complementary information about the environment, overcome hardware limitations, or reduce data uncertainty due to each individual sensor. An essential step in data fusion is extrinsic calibration, which determines the geometric parameters of each sensor, such as position and orientation, with respect to the other ones. During fusion, the calibration parameters are used to compute the geometric transformation that maps the output of each sensor to a common frame of reference.

The fusion problem we consider in this paper is depth superresolution: low-resolution depth data from a LIDAR sensor is fused with an image from an optical camera to produce a higher-resolution depth image [1–4]. In particular, we are interested in calibrating and fusing the output of a LIDAR sensor that provides a three-dimensional (3D) point-cloud of depth measurements with an optical camera observing the same scene, as illustrated in Figure 1.

The key insight in our paper is that fusion and calibration are complementary processes and improvements in one can be used to boost the performance of the other. Thus, a joint calibration and fusion method significantly improves the final output. In contrast to existing approaches, which typically address either calibration or fusion, but not both, the joint processing we describe works towards a common objective: improving the final output of the pipeline. This objective is reached by jointly optimizing over the calibration parameters and the fusion output, using the appropriate cost function in each case.

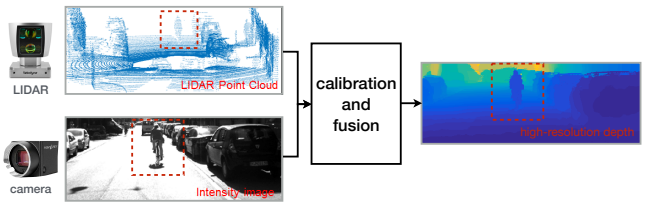

Fig. 1. High-resolution depth image obtained after calibrating and fusing intensity image with a sparse depth-map. The highlights in the figure show the biker.

In the next section we present a brief background and discuss in detail the key elements of our approach, especially in relation to existing approaches. Section 3 formulates and develops our approach, including the joint optimization algorithm. Section 4 presents experimental results on real and synthetic data validating our approach. Section 5 concludes and discusses our findings.

2. BACKGROUND

Most calibration approaches, including the one described in this paper, are fundamentally based in identifying and matching features acquired from the two sensors and determining the calibration parameters, i.e., a common geometric frame of reference, using those features. The most common approaches require the existence of known targets in the scene, which are used for feature matching. These targets are typically placed in the scene during a calibration stage and their signatures in the acquired scenes are used to establish correspondences across modalities. For example, [5] exploits a planar checkerboard pattern and uses nonlinear least squares optimization to calibrate a 2D laser scanner with a single optical camera. Follow up work extended and refined this approach [6–8], while others considered using other types of targets such as right-angled triangles [9], circles [10], or white-to-black transitions [11].

Unfortunately, such target-based methods are typically performed offline, separately from subsequent processing, and cannot correct calibration errors that occur during the operation of the sen-

*The first author performed this work while at Mitsubishi Electric Research Laboratories.

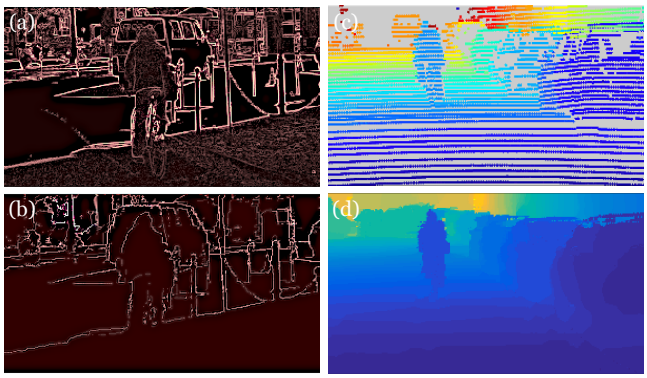

Fig. 2. Normalized gradient magnitudes at the correct calibration parameters corresponding to the (a) intensity and (b) fused depth images from the KITTI dataset [17]. There is a significant correlation between the edge images at the correct calibration parameters. Result of estimating a (d) high-resolution depth image from the (c) projected depth point-cloud from a single frame.

sors due to drift caused, for example, by vibration or due to changes in the sensor configuration. Furthermore, these methods become impractical in mass-produced systems, especially if manufacturing tolerances result in variations of the calibration parameters among systems to affect further processing and data fusion. In such cases, these method would require individual calibration of each system, increasing the manufacturing cost.

Accordingly, more recent work attempts automatic calibration using features present in the observed scene, without dedicated targets. For example, in [12] the calibration parameters are selected to minimize the χ^2 distance between LIDAR reflectivity and camera image intensity. More recently, in [13] the calibration algorithm determines parameters that maximize the mutual information between LIDAR reflectivity and camera image intensity. If reflectivity values are not available or not reliable from the LIDAR, it is also possible to directly correlate the depth discontinuity at each LIDAR point with edges in the intensity image [14, 15] or use the mutual information between the signals [16].

The main drawback of existing automated calibration methods that do not require a target is that, quite often, the sampling density and geometry in one modality do not match the sampling density and geometry in the other, thus making comparisons difficult. For example, camera images typically comprise a dense two dimensional array of pixel values. In contrast, LIDAR data are typically represented as a sparse point cloud, heavily subsampled in the vertical direction, and often do not include reflectivity values. Identifying, for example, edges in the latter is not straightforward.

In this paper, we propose an alternative approach that can calibrate the sensors while simultaneously superresolving the sparse depth from the LIDAR. Specifically, we design a cost functional penalizing the misalignment between the high-resolution depth and intensity images and propose a simulated-annealing-based optimization algorithm for finding the optimal calibration parameters. Our approach is based on the observation that the depth and intensity edges are more likely to be aligned at the correct calibration parameters (see Figure 2). The high-resolution depth image is obtained by solving total-variation (TV) regularized least squares problem [18]. We experimentally demonstrate the ability of our method to accurately calibrate the sensors without additional laser reflectivity information on both simulated and experimentally measured data in the context of autonomous vehicles. We believe that our method could enhance the applicability of sensor fusion algorithms that rely on accurate multimodal calibration of sensors.

Algorithm 1 Joint calibration and sensor fusion

input: initial guess $\boldsymbol{\theta}^0$, depth samples $\boldsymbol{\psi}$, and intensity image $\mathbf{x}$
set: $t \leftarrow 1$
repeat
  $T_t \leftarrow \text{temperature}\{T_{t-1}\}$
  Pick a random neighbour: $\boldsymbol{\theta}^t \leftarrow \text{neighbour}\{\boldsymbol{\theta}^{t-1}\}$
  Update high-resolution depth $\boldsymbol{\phi}_{\boldsymbol{\theta}^t}$ by solving (8).
  if $P\left(\mathcal{F}(\boldsymbol{\theta}^t) > \mathcal{F}(\boldsymbol{\theta}^{t-1}), T\right) > \text{rand}(0,1)$
    or $\mathcal{F}(\boldsymbol{\theta}^t) < \mathcal{F}(\boldsymbol{\theta}^{t-1})$
    $\widehat{\boldsymbol{\theta}} \leftarrow \boldsymbol{\theta}^t$ and $\widehat{\boldsymbol{\phi}} \leftarrow \boldsymbol{\phi}_{\boldsymbol{\theta}^t}$
until: stopping criterion
return: $\widehat{\boldsymbol{\theta}}$ and $\widehat{\boldsymbol{\phi}}$

3. PROPOSED APPROACH

3.1. Problem Formulation

Extrinsic calibration of a LIDAR with a camera attempts to determine a perspective projection of the frame of reference of one system to the frame of reference of the other. The formulation we consider in this paper considers a transformation with 6 degrees of freedom (DOF): three angle rotations (roll, pitch and yaw) and three translations (along the x, y and z axes). We compactly represent those using the respective parameters $\boldsymbol{\theta} = (\theta_{\text{roll}}, \theta_{\text{pitch}}, \theta_{\text{yaw}}, \theta_x, \theta_y, \theta_z)$.

For convenience, the global frame of reference typically coincides with the frame of reference of one of the sensor. In our formulation, the camera geometry provides the global frame of reference. Of course, once the calibration is known, the geometry can be trivially mapped to any frame of reference, including that of the LIDAR. We denote the recorder camera image in its frame of reference using $\mathbf{u} \in \mathbb{R}^{N_x \times N_y}$, where N_x and N_y are the number of horizontal and vertical pixels respectively, for a total of $N = N_x N_y$ pixels. Note that in the remainder of this paper we often use $n \in \{1, \ldots, N\}$ to index the elements of $\mathbf{u}$ and other similarly sized matrices, essentially vectorizing them.

LIDAR data are typically represented as a point cloud, i.e., a set of (x, y, z) Cartesian coordinates from which a LIDAR pulse reflection has been recorded. We denote the set of those points using $\boldsymbol{\psi} \in \mathbb{R}^{M \times 3}$, where M is the number of recorded reflections and each point coordinates are represented in a row of $\boldsymbol{\psi}$. Often LIDAR systems represent data in a polar coordinate system instead of a cartesian one. Transformation of one to the other is trivial; we use Cartesian coordinates in this paper.

We should also note that LIDAR systems will only record reflections they detected. Lack of points along a particular direction may mean either that there is no object present along that direction, or that the LIDAR system never illuminated that direction. A number of LIDAR products only return the reflections recorded, also known as *hits* and not the *misses*, i.e., the pulses that were never reflected. Our development does not assume a particular LIDAR scanning geometry or schedule, and, therefore, we assume that the empty space has just not been illuminated. If information about misses is available, it can be easily used to enhance performance.

Given transformation parameters $\boldsymbol{\theta}$, we can map the 3-D LIDAR point cloud onto the reference 2D image plane of the camera using a

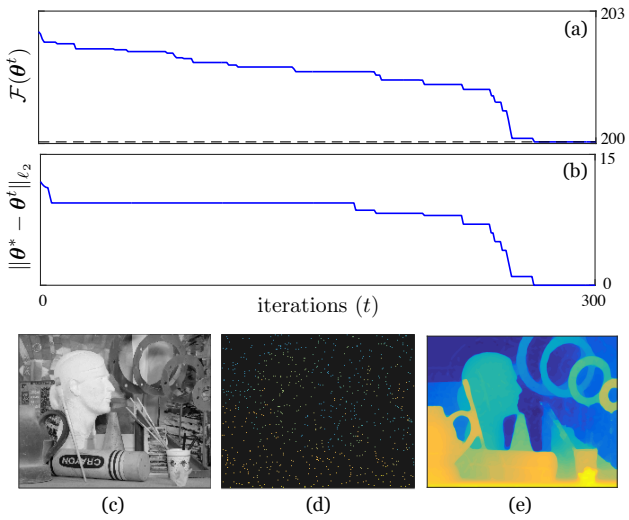

Fig. 3. Calibration experiment on the Middlebury dataset [19]: (a) evolution of the cost function; (b) evolution of the error between the true and estimated parameters; (c) intensity image; (d) sparse depth measurements; (e) fused depth image.

linear 3D rigid body transformation transformation.

$$\mathcal{P}_{\boldsymbol{\theta}}(\boldsymbol{\psi}) = \left(\begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{\psi}^T \\ \mathbf{1}^T \end{bmatrix} \right)^T, \tag{1}$$

where $\boldsymbol{\psi}^T$ denotes the transpose of $\boldsymbol{\psi}$, $\mathbf{R}$ is the rotation matrix from 3D rotation group SO(3), characterized by the roll, pitch and yaw angles, and $\mathbf{t} \in \mathbb{R}^3$ is specified by the translations θ_x, θ_y and θ_z.

The goal of automatic calibration is to estimate the correct transformation parameters $\boldsymbol{\theta}$ directly from the camera and LIDAR data $\widehat{\boldsymbol{\theta}} = f_{\text{calibration}}(\mathbf{u}, \boldsymbol{\psi})$.

The LIDAR data, in contrast to the camera image, significantly subsample the observed scene. Given calibration parameters, the goal of fusion in the context of this work is to estimate a high-resolution depth map $\boldsymbol{\phi}_{\boldsymbol{\theta}} \in \mathbb{R}^{N_x \times N_y}$, where each element of $\boldsymbol{\phi}$ indicates the depth of the corresponding pixel with respect to the camera frame of reference $\widehat{\boldsymbol{\phi}} = f_{\text{fusion}}(\mathbf{u}, \boldsymbol{\psi}, \boldsymbol{\theta})$.

3.2. Automatic calibration

To estimate the calibration parameters, our approach exploits the output of the fusion process. Specifically, we use $\boldsymbol{\phi}_{\boldsymbol{\theta}} = f_{\text{fusion}}(\mathbf{u}, \boldsymbol{\psi}, \boldsymbol{\theta})$ to compactly denote the high-resolution depth image estimated given the calibration parameters $\boldsymbol{\theta}$. The calibration parameters are estimated by minimizing a cost function on $\boldsymbol{\phi}_{\boldsymbol{\theta}}$ which penalizes miss-alignments between the gradients of the projected high-resolution depth map and the intensity image

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} \{\mathcal{F}(\boldsymbol{\theta})\}, \tag{2}$$

where,

$$\mathcal{F}(\boldsymbol{\theta}) = \sum_{k \in \{x,y\}} \frac{\mathcal{A}_k(\boldsymbol{\phi}_{\boldsymbol{\theta}})}{\mathcal{N}_k(\boldsymbol{\phi}_{\boldsymbol{\theta}})}. \tag{3}$$

In particular, the numerator $\mathcal{A}_k(\boldsymbol{\phi}_{\boldsymbol{\theta}})$, measures the weighted ℓ_1 total variation (TV) of the high-resolution depth image, evaluated only at the points in which the LIDAR performs real measurements.

$$\mathcal{A}_k(\boldsymbol{\phi}_{\boldsymbol{\theta}}) = \sum_{n \in \Omega_{\boldsymbol{\theta}}} w_{k,n} |[\boldsymbol{\nabla}_k \boldsymbol{\phi}_{\boldsymbol{\theta}}]_n|, \tag{4}$$

where k is the horizontal or vertical direction, x or y, $\boldsymbol{\nabla}_k$ is the gradient operating on $\boldsymbol{\phi}_{\boldsymbol{\theta}}$ along direction k. The weights $w_{k,n}$ are specified using the camera image, $w_{k,n} = \exp(-\gamma\, |[\boldsymbol{\nabla}_k \mathbf{u}]_n|)$, thus promoting edge alignment between the camera image and the high-resolution depth map, as controlled by the fixed parameter $\gamma \geq 0$.

The set $\Omega_{\boldsymbol{\theta}} \neq \emptyset$ denotes the subset of points where we have obtained depth measurements from the LIDAR. Restricting the cost to those points ensures that we consider only the regions where the depth image is reliable. Note that, even though the function is only computed in those points, computing the high-resolution image is required in order to be able to calculate the vertical and horizontal gradients at these points.

Given an edge in the intensity image, a large γ does not penalize a corresponding edge in the depth map. Thus, only edges in the depth map that do not have corresponding edges in the intensity map, i,e, not properly aligned, are penalized. A small γ, on the other hand, reduces the effect of the intensity image, thus penalizing edges in the depth map similarly, irrespective on whether the intensity map has a corresponding edge.

The denominator is a necessary normalization factor that takes into account the difference in number of points in $\Omega_{\boldsymbol{\theta}}$ that are included in the evaluation given different parameter sets $\boldsymbol{\theta}$

$$\mathcal{N}_k(\boldsymbol{\phi}_{\boldsymbol{\theta}}) = \left(\sum_{n \in \Omega_{\boldsymbol{\theta}}} w_{k,n} \right) \left(\sum_{n \in \Omega_{\boldsymbol{\theta}}} |[\boldsymbol{\nabla}_k \boldsymbol{\phi}_{\boldsymbol{\theta}}]_n| \right). \tag{5}$$

3.3. Sensor Fusion

Given a fixed parameter vector $\boldsymbol{\theta}$, fusion estimates $\widehat{\boldsymbol{\phi}}_{\boldsymbol{\theta}}$ using the image $\mathbf{u}$ and the projection of the LIDAR data to the camera image frame $\mathcal{P}_{\boldsymbol{\theta}}(\boldsymbol{\psi})$.

Our approach uses an ℓ_2 penalty to promote consistency of the resulting depth map with the projected data. In other words, we measure data fidelity using

$$\mathcal{D}(\boldsymbol{\phi}) = \frac{1}{2} \|\mathcal{P}_{\boldsymbol{\theta}}\{\boldsymbol{\psi}\} - \mathbf{H}\boldsymbol{\phi}\|_{\ell_2}^2 \tag{6}$$

where $\mathbf{H} : \mathbb{R}^N \to \mathbb{R}^M$ is a masking operator which selects only the indices where we have data in the projection $\mathcal{P}_{\boldsymbol{\theta}}\{\boldsymbol{\psi}\}$.

Since the depth data are significantly subsampled, we regularize the problem using the weighted isotropic total variation (TV) [18] of the depth

$$\mathcal{R}(\boldsymbol{\phi}) = \sum_{n=1}^{N} w_n \|[\boldsymbol{\nabla}\boldsymbol{\phi}]_n\|_{\ell_2}, \tag{7}$$

which promotes sharper edges, according to the weights w_n [20].

The resulting fusion algorithm minimizes the following optimization

$$\hat{\boldsymbol{\phi}}_{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\phi} \in \boldsymbol{\Phi}} \{\mathcal{D}(\boldsymbol{\phi}) + \lambda \mathcal{R}(\boldsymbol{\phi})\}, \tag{8}$$

where $\boldsymbol{\Phi} \subseteq \mathbb{R}^N$ is used to enforce certain physical constraints on the depth, such as non-negativity, and the fixed parameter $\lambda > 0$ controls the amount of regularization.

The fusion algorithm exploits the camera data to improve the fusion output through the weights w_n. Specifically, the weights can be chosen as a function of the gradient of the image at each pixel, such

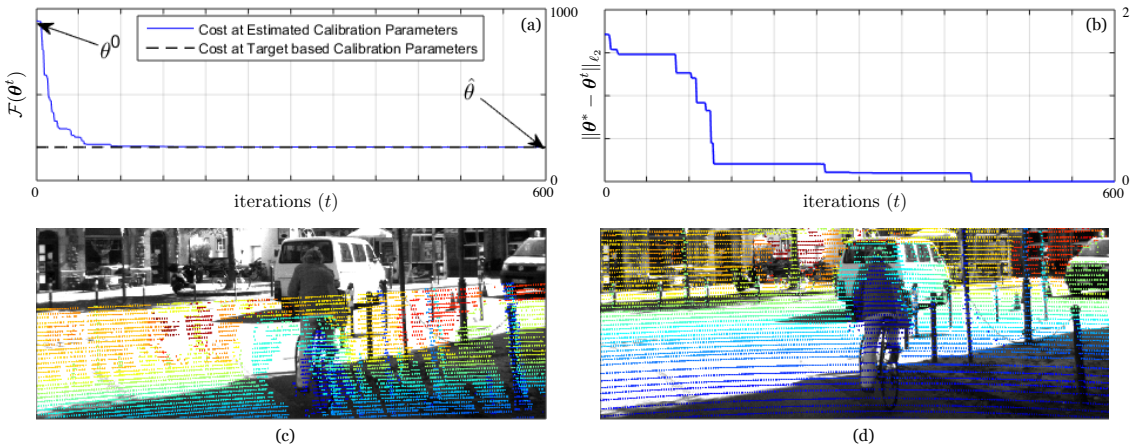

Fig. 4. Calibration experiments on the KITTI dataset: (a) evolution of the cost function; (b) evolution of the error between the true and estimated parameters; (c) projected depth at the initial calibration parameters; (d) projected depth at the final calibration parameters.

that corresponding edges in the depth image are not penalized significantly [21]. If the data are not properly calibrated, this weighting may introduce spurious edges and further confuse subsequent calibration. Thus, while calibration is performed the weights are chosen as $w_n = 1$. Once the calibration parameters are estimated, a final fusion step is performed $w_n = \exp(-\tau\|[\nabla \mathbf{x}]_n\|_{\ell_2})$ for some constant $\tau > 0$. Figure 2 illustrates the result of intensity and depth fusion at the correct calibration parameters $\boldsymbol{\theta}$.

3.4. Algorithm

Since the cost functional (3) is non-convex, we rely on a simulated annealing based approach to find the calibration parameters and solve (2). The overall method is summarized in Algorithm 1. The high-resolution depth in step 6 of the algorithm is obtained by iteratively solving (8) with a fast iterative shrinkage/thresholding algorithm (FISTA) [22]. We initialize FISTA with the result of a linear interpolator, which allows it to solve (8) within very few iterations.

4. EXPERIMENTS

To validate our algorithm we use the Middlebury dataset [19] which provides dense 2D depth-maps and corresponding 2D intensity images. As shown in Figure 3, we first subsample the dense depth-map by taking only 2% of the total number of pixels. To test automatic calibration, this subsampled version is first randomly translated and rotated before the algorithm is used for calibration, i.e., to determine and undo the translation and rotation, and fusion.

The results of applying algorithm 1 to this dataset are summarized in Figure 3. Figure 3(c) and 3(e) show the intensity image and the high-resolution reconstruction of depth, respectively. Figure 3(a) and 3(b) summarizes the progression of the objective function and of the ℓ_2 error of the calibration parameters with respect to the true parameters as a function of iterations. Note that our approach overall performed pretty well for all the images in this dataset.

For additional validation, we test the algorithm on the KITTI dataset [17]. This dataset contains intensity images $\mathbf{u}$ and 3D point clouds $\boldsymbol{\psi}$ from urban and rural scenarios. In addition, the dataset provides extrinsic calibration parameters $\boldsymbol{\theta}$ that can be used to verify the ground truth and measure the error of our algorithms. The method used to generate these calibration parameters is a target based automatic calibration which is run prior to acquisition.

In the experiments, we initialize the calibration algorithm using a random perturbation added to the ground truth calibration parameters, as these are provided by the dataset. The perturbation is randomly generated using a uniform distribution between $\pm$ 10 degrees in the case of the rotational parameters, $\pm$ 70 pixels in the case of the θ_x, θ_y translations and $\pm$ 10 pixels in the case of θ_z translation.

Our initialization simulates a coarse manual initialization with significant tolerance and should be sufficient to illustrate the effectiveness of our approach. LIDAR optical cameras are often mounted close-by with small tolerances. Furthermore, if calibration is required, it is usually to update an existing calibration due to small change in the orientation of the sensors.

An example of the initial projection of the point-cloud in one of our experiments is shown in Figure 4(c). This initial projection was used as an input along with $\boldsymbol{\psi}$ and $\mathbf{X}$ on our calibration algorithm. The progression of (3) and of the ℓ_2 error between the solution and the ground truth parameters provided in KITTI is shown as a function of iterations in Figures 4(a) and 4(b), respectively. Note that both the objective function and ℓ_2 error significantly decrease as a function of iterations. Figure 4(d) illustrates the projection at the calibration parameters $\hat{\boldsymbol{\theta}}$ found by minimizing (3).

5. CONCLUSION AND DISCUSSION

Our paper presents a joint automatic calibration and fusion approach for multimodal sensors comprising of 2D/3D LIDAR and optical cameras. This approach is based on the fundamental realization that subsequent processing steps, such as sensor fusion, can be used to inform and significantly improve the calibration step in a signal processing pipeline.

Our approach is based on implicitly matching and preserving the edges present in the two modalities using a weighted total variation norm. This norm is able to exploit the additional information extracted by the fusion process and aid calibration but, also, inform the fusion process in generating a high-fidelity depth image. Our experimental results confirm our intuition and validate our approach. Of course, this approach is immediately applicable to calibration of other imaging modalities, both 2D and 3D.

6. REFERENCES

[1] J. Park, H. Kim, Y.-W. Tai, M. S. Brown, and I. Kweon, "High quality depth map upsampling for 3D-TOF cameras," in *Proc. IEEE Int. Conf. Comp. Vis.*, Barcelona, Spain, November 6-13, 2011, pp. 1623–1630.

[2] Y. Li, T. Xue, L. Sun, and J. Liu, "Joint example-based depth map super-resolution," in *Proc. IEEE Int. Con. Multi.*, Melbourne, VIC, Australia, July 9-13, 2012, pp. 152–157.

[3] D. Ferstl, C. Reinbacher, R. Ranftl, M. Ruether, and H Bischof, "Image guided depth upsampling using anisotropic total generalized variation," in *Proc. IEEE Int. Conf. Comp. Vis.*, Sydney, NSW, Australia, December 1-8, 2013, pp. 993–1000.

[4] M.-Y. Liu, O. Tuzel, and Y. Taguchi, "Joint geodesic upsampling of depth images," in *Proc. IEEE CVPR*, Portland, OR, USA, June 23-28, 2013, pp. 169–176.

[5] Q. Zhang, "Extrinsic calibration of a camera and laser range finder," in *Proc. IEEE Int. Conf. Int. Robot. Sys.*, Sendai, Japan, 2004, pp. 2301–2306.

[6] C. Mei and P. Rives, "Calibration between a central catadioptric camera and a laser range finder for robotic applications," in *Proc. IEEE Int. Conf. Rob. Autom.*, Orlando, FL, USA, 2006, pp. 532–537.

[7] G. Pandey, J. R. McBride, S. Savarese, and R. M. Eustice, "Extrinsic calibration of a 3D laser scanner and an omnidirectional camera," in *IFAC Symposium on Intelligent Autonomous Vehicles*, Leece, Italy, 2010, pp. 336–341.

[8] L. Zhou and Z. Deng, "Extrinsic calibration of a camera and a LIDAR based on decoupling the rotation from the translation," in *Proc. IEEE Int. Veh. Sym.*, Madrid, Spain, 2012, pp. 642–648.

[9] G. Li, Y. Liu, L. Dong, X. Cai, and D. Zhou, "An algorithm for extrinsic parameters calibration of a camera and a laser range finder using line features," in *Proc. IEEE Int. Conf. Int. Robot. Sys.*, San Diego, CA, USA, 2007, pp. 3854–3859.

[10] F. Rodriguez, V. Fremong, and P. Bonnifait, "Extrinsic calibration between a multi-layer LIDAR and a camera," in *Proc. IEEE Int. Conf. Mult. Sensor Fusion and Integration for Intelligent Syst.*, Seoul, South Korea, 2009, pp. 214–219.

[11] O. Naroditsky, A. Patterson, and K. Daniilidis, "Automatic alignment of a camera with a line scan LIDAR system," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, Shanghai, China, May 9-13, 2011, pp. 3429–3434.

[12] N. Williams, K.-L. Low, C. Hantak, M. Pollefeys, and A. Lastra, "Automatic image alignment for 3D environment modeling," in *Proc. IEEE SIBGRAPI*, Curitiba, PR, Brazil, October 17-20, 2004, pp. 388–395.

[13] G. Pandey, J.R. McBride, S. Savarese, and R.M. Eustice, "Automatic extrinsic calibration of vision and lidar by maximizing mutual information," *Journal of Field Robotics*, vol. 32, no. 5, pp. 1–27, August 2014.

[14] J. Levinson and S. Thrun, "Automatic online calibration of cameras and lasers," in *Robotics: Science and Systems*, Berlin, Germany, June 24-28, 2013, pp. 29–36.

[15] S. Bileschi, "Fully automatic calibration of lidar and video streams from a vehicle," in *Proc. IEEE Int. Conf. Comp. Vis. Workshops (ICCV WKSHP)*, Kyoto, Japan, September 27-October 4, 2009, pp. 1393–1400.

[16] Z. Taylor and J. Nieto, "A mutual information approach to automatic calibration of camera and lidar in natural environments," in *Proc. Austral. Conf. Robot. Autom.*, Victoria University of Wellington, New Zealand, Dec 3-5, 2012.

[17] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *Int. J. of Rob. Res.*, 2013.

[18] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, vol. 60, no. 1–4, pp. 259–268, November 1992.

[19] D. Scharstein and C. Pal, "Learning conditional random fields for stereo," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[20] U. S. Kamilov, I. N. Papadopoulos, M. H. Shoreh, D. Psaltis, and M. Unser, "Isotropic inverse-problem approach for two-dimensional phase unwrapping," *J. Opt. Soc. Am. A*, vol. 32, no. 6, pp. 1092–1100, June 2015.

[21] M. P. Friedlander, H. Mansour, R. Saab, and O. Yilmaz, "Recovering compressively sampled signals using partial support information," *IEEE Trans. Inf. Theory*, vol. 58, no. 2, pp. 1122–1134, February 2012.

[22] A. Beck and M. Teboulle, "Fast gradient-based algorithm for constrained total variation image denoising and deblurring problems," *IEEE Trans. Image Process.*, vol. 18, no. 11, pp. 2419–2434, November 2009.