Choosing a Time and Place for Calibration of Lidar-Camera Systems

Terry Scott, Akshay A. Morye, Pedro Piniés, Lina M. Paz, Ingmar Posner, Paul Newman

***Abstract*— We propose a calibration method that automatically estimates the extrinsic calibration between a sensor pose-graph from natural scenes. The sensor pose-graph represents a system of sensors comprising of lidars and cameras, without sensor co-visibility constraints. The method addresses the fact that each scene contributes differently to the calibration problem by introducing a diligent scene selection scheme. The algorithm searches over all scenes to extract a subset of exemplars, whose joint optimisation yields progressively better calibration estimates. This non-parametric method requires no knowledge of the physical world, and continuously finds scenes that better constrain the optimisation parameters. We explain the theory, implement the method, and provide detailed performance analyses with experiments on real-world data.**

I. Introduction

It is impossible to overstate the importance of good calibration. Within the mobile robotics domain, the use of complementary sensors like lidars and cameras is ubiquitous, which makes spatial calibration between lidars and cameras an imperative and very challenging, problem.

Accurately calibrating multi-modal sensors allows photo-realistic renditions of the environment within which a robot traverses, such as that shown in Fig. 1. Manual measurement of the true sensor pose is prone to errors due to many reasons, primary among which is that the sensor's protective casing often occludes the sensing element. Conventionally, calibration of lidar-camera systems is performed using dedicated targets and manual selection of lidar-camera feature correspondences the majority of which typically require simultaneous observation by the sensors under consideration.

In this paper, we propose a generic calibration algorithm, and apply it to calibrate a sensor system consisting of 2D lidars and cameras, where the sensor fields-of-view (FoVs) are not required to overlap. This relaxation on sensor co-visibility makes the problem *even harder* to solve. To complement the calibration framework, we propose a method of *diligent* scene selection that does not treat all observed scenes as equal. This is particularly important in a data-driven approach such the one we propose here.

A. Literature Review

The existing paradigm of extrinsic calibration techniques is to employ the use of known calibration targets like fiducial markers or planar checkerboards [1]–[3]. An early implementation of this technique is described in [1] where Zhang et al. [1] minimise the reprojection error of checkerboard features observed simultaneously by a camera and 2D lidar. Also using a checkerboard, the authors of [3] calibrate a 2D lidar to a camera by initially computing the relative transform using six measurements of the target, followed by solution refinement using RANSAC-based least-squares.

This work was supported by the Technology Strategy Board UK, under Ref. No. 101699 for the project "High-speed railway asset mapping system using enhanced 3D imaging and automated visual analytics.", and the EPSRC Programme Grant, under Ref. No. EP/M019918/1 for the project "Mobile Robotics: Enabling a Pervasive Technology of the Future"

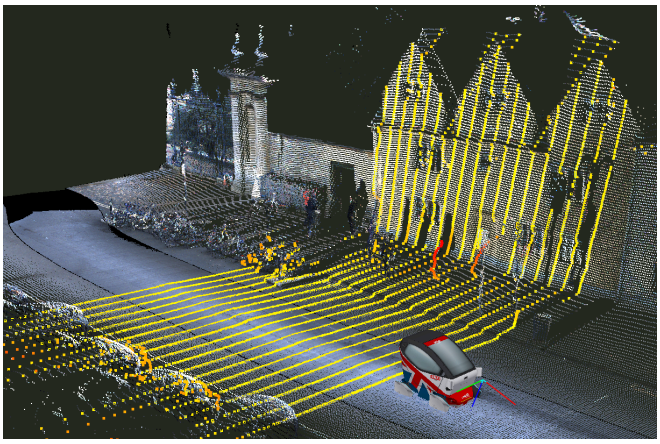

Fig. 1: Lidar-Camera Calibration: An autonomous vehicle is shown travelling through a coloured 3D point cloud created by fusing data from a 2D lidar and a multi-camera setup. Vehicle pose is obtained from a stereo camera mounted on the front (facing-forward), and a 3D point cloud is created from a lidar mounted at the back (push-broom). Lidar data is coloured using images from side-facing monocular cameras. This photo-realistic reproduction of the environment is only possible due to 'good' calibration. We generate a swathe of 2D lidar scans (shown as yellow points) to mimic sensor fields-of-view overlap as a function of vehicle motion.

An extension of [1] for application to 3D lidars is described in [4]. The checkerboard plane is extracted in both the laser and camera data, and the relative position of each sensor is determined by aligning the plane normals obtained from multiple scans. Unnikrishnan et al. [5] provide a 3D lidar-camera calibration toolbox that iteratively minimises a geometric planar constraint-based nonlinear least-squares cost function. In another method for calibrating 3D lidars, Mirzaei et al. [6] decouple the problem of intrinsic and extrinsic calibration into sub-problems. Initial estimate refinement is carried out for each sub-problem by minimising a batch nonlinear least-squares cost function. Geiger et al. [7] provide a method that uses multiple planar checkerboards to calibrate a multi-beam lidar to multiple cameras, using only a single camera image. This significantly reduces the data processing overhead.

Interestingly, the methods listed above are only applicable in the presence of known calibration targets and require these targets to be in the fields-of-view (FoV) of all sensors. Some methods that perform calibration without the presence of fiducial targets have been proposed recently. A target-

less calibration technique is proposed by Scaramuzza et al. [8], wherein the parameters are computed by applying the perspective-from-n-points (PnP) algorithm [9] on manually selected point correspondences between camera pixels and lidar points. Levinson et al. [10] relax the need for user input by discovering and correlating edges in an omnidirectional camera image to discontinuities observed in 3D lidar range measurements. Pandey et al. [11] maximise the Mutual Information by registering reflectance values obtained from a Velodyne 64-beam lidar to camera pixel intensities.

It has been shown that natural scenes can cause calibration to be unreliable or fail completely in some instances [10], [12], [13]. To the best of our knowledge, there has been no explicit attempt to address this issue. Our automatic scene selection method ensures that failure cases are excluded from the scene set used for calibration.

B. Motivation and Contribution

Sensor co-visibility is essential to apply the methods listed above. This imposes limitations on designing sensor configurations for a given application. In addition, 2D lidars are smaller, cheaper and produced on a larger scale than 3D lidars thus making them favourable for commercial robotics.

Our motivation is life-long calibration. During the lifetime of a robot, calibration may change during operation, which could lead to a need for periodic recalibration. Target-based calibration becomes impractical for working robots. We propose an automatic and diligent calibration method, where calibration is performed from natural scenes collected in the robot's workspace.

The accuracy of data-driven calibration methods is reliant on sensor observations. Each scene contributes differently towards solving the calibration problem. For example, data collected in an urban environment has a higher probability of entailing structural information like lines and planes than if the data were collected in an off-road environment, where it may be relatively featureless (e.g. sky, desert, etc.), or noisy (e.g. foliage, etc.). We tackle this by proposing a scene selection scheme, that searches over all available scenes and extracts a subset of exemplars that are 'ideal' for calibration.

The proposed method ranks individual scenes based on a continually updated objective function profile. We exploit the fact that the objective function embeds scene appearance and structural information, which would make it suitable for calibration. In this way, we eschew the need for a dedicated one-shot calibration phase, akin to that of [7], and opt instead for a continual policy of improvement. This is advantageous as it means we do not need to design a calibration scene that is specifically suited for the particular geometry of the system. We demonstrate that suitable scenes are discovered in an unsupervised manner over the operational lifetime of the robotic platform.

In this paper, we employ these *diligently* selected scenes to calibrate a sensor pose-graph using an extension our previous work [14]. Therein, we described an automatic, targetless, scene-induced method for extrinsic calibration of push-broom 2D lidars with a stereo camera. A 2D lidar would have to be configured in such a way that it produces a 3D swathe from ego-motion. The extension allows for any number of sensors to be integrated thus providing a complete lidar-camera system configuration for practically any robot.

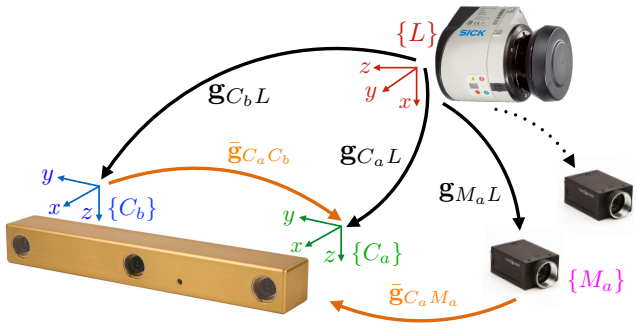

Fig. 2: Sensor Pose Graph: A setup with multiple local sensor frames of reference; the lidar frame $\{L\}$ (red), the camera frames $\{C_a\}$ (green) and $\{C_b\}$ (blue), associated with each camera of a stereo camera unit, and the camera frame $\{M_a\}$ (pink) associated with a monocular camera. The solid curved arrows denote the 6-DoF transforms between the individual frames, where the black arrows denote the transforms to be optimised, and the orange arrows denote inter-camera transform measurements. The dashed black curved arrow indicates the ability to augment an existing sensor pose-graph with additional sensors.

II. Problem and Solution Overview

For ease of understanding, we first recapitulate the calibration problem from [14]. Subsequently, wherever necessary, the calibration problem is extended to handle calibration of additional sensors.

In [14], the global calibration problem is decoupled into sub-problems as a closed-loop, hierarchical relationship between two alternating optimiser levels, distributed over a lower-level and an upper-level. The lower-level solves $N_C N_L$ problems, with N_C being the number of cameras and N_L being the number of lidars. The upper-level implements a nonlinear least-squares refinement step to minimise the error between the solutions computed by the $N_C N_L$ lower-level optimisers. The upper and lower-level optimisers are linked with a quadratic penalty term [15]. Optimisation stops when time allotted for optimisation elapses, or a user-defined threshold is satisfied.

A. Notation

A six degrees-of-freedom (6-DoF) rigid-body transform that registers entities defined in source frame $\{A\}$ to destination frame $\{B\}$ is described by matrix $\mathbf{G}_{BA} \in \mathrm{SE}(3)$. Matrix $\mathbf{G}_{BA}$ is parameterised by a tuple $\mathbf{g}_{BA} \in \Re^6$, where $\mathbf{g}_{BA} = (t_x, t_y, t_z, \theta, \rho, \psi)$, with t_x, t_y, and t_z being the relative translation components in metres, and θ, ρ, and ψ being the relative rotational components in radians, i.e. roll, pitch, and yaw angles, respectively.

Fig. 2 depicts a case where $N_C = 3$ and $N_L = 1$. The tuples $\mathbf{g}_{C_aL}$ and $\mathbf{g}_{C_bL}$ define transforms that register an entity defined in lidar frame $\{L\}$, to a frame associated with each camera of the stereo camera unit. The tuple $\mathbf{g}_{M_aL}$ defines transforms that register an entity defined in lidar frame $\{L\}$, to a frame associated with a monocular

camera. The tuples $\bar{\mathbf{g}}_{C_aC_b}$ and $\bar{\mathbf{g}}_{C_aM_a}$ define measured rigid-body transforms, that register entities defined in $\{C_b\}$ and $\{M_a\}$ to $\{C_a\}$, respectively. The transforms $\mathbf{g}_{C_aL}$, $\mathbf{g}_{C_bL}$, and $\mathbf{g}_{M_aL}$ are denoted by solid black curved arrows, while the inter-camera transform measurements $\bar{\mathbf{g}}_{C_aC_b}$ and $\bar{\mathbf{g}}_{C_aM_a}$ are denoted by solid orange curved arrows. The black dashed curved arrows show that an existing sensor pose graph can be augmented with additional sensor nodes. Note that each additional sensor node M_p would require the knowledge of the tuple $\bar{\mathbf{g}}_{C_aM_p}$.

B. Swathe Generation Background

Vehicle motion is necessary for lidar swathe generation [16]. Our approach is related to the cross-calibration method proposed by Napier et al. [12], and is detailed in [14].

We exploit vehicle motion to generate a swathe of lidar scans for generating a 3D point cloud. By assuming the consequent overlap between the FoV of both sensor modalities as a result of vehicle motion, we project the generated swathe into the relevant camera's image plane to compute a similarity measure between sensor observations.

We use a stereo camera to obtain the vehicle trajectory via visual odometry (VO). From the method in [14], a point $\mathbf{p} = [x, y, z]^\top$, when observed in the lidar frame $\{L^j\}$ at time j, and in the camera frame $\{C^k\}$ at time k, can be registered to a common frame of reference $\{R\}$ by:

$$^R\mathbf{p} = \mathbf{g}_{RC^j} \oplus \mathbf{g}_{C^jL^j} \oplus {}^{L^j}\mathbf{p}, \quad (1)$$

$$^R\mathbf{p} = \mathbf{g}_{RC^k} \oplus {}^{C^k}\mathbf{p}. \quad (2)$$

The symbols $\oplus$ and $\ominus$ are composition operators. Eqns. (1) and (2) show that any $\mathbf{p} \in \Re^3$ observed in both sensors at different times can be projected to $\{R\}$, if accurate estimates for camera pose at times j and k can be obtained from VO, and if an optimised rigid-body transform $\mathbf{g}_{C^jL^j}$ is available. Thus, swathe generation is a function of the extrinsic calibration parameters to be optimised.

This paper focuses on optimising $\mathbf{g}_{C^jL^j}$.[1] In the following section, we describe the scene selection process to facilitate life-long calibration.

III. Diligent Scene Selection

Not all scenes are equal. Calibration targets are designed to provide regular, strongly discernible signals in both the camera image and lidar modalities. In this section we discuss the general characteristics of scenes that make them more or less suitable than others. More importantly, we also need to specify what it means for a scene to be *suitable*.

We start by imagining an ideal cost function, which may resemble that in the left-hand plot of Fig. 3, which is locally convex, with a distinct minimum and a large second derivative at that minimum. With these desirable properties, let us first select a calibration cost function.

[1]For brevity, we will drop the time-index from the notation wherever possible in subsequent sections.

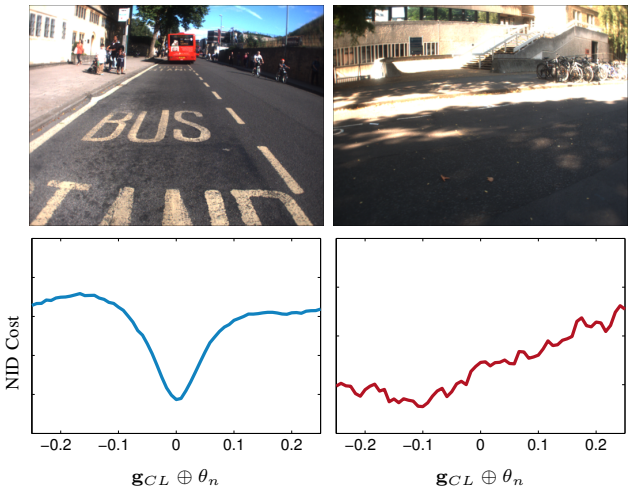

Fig. 3: Good and Bad Scenes: Top row (left) shows uniformly and well lit scene with distinct road markings, and an image under shade (right), with overexposed regions in the sun. Shapes of their corresponding cost function in which we seek a global minimum are on the bottom row.

A. Normalised Information Distance (NID)

Normalised Information Distance $D(X, Y)$ is a measure of the statistical correlation between two discrete random variables X and Y. We define X to be the reflectance value for a point $\mathbf{p} \in \Re^3$ observed by lidar L, and Y to be the image intensity value of the pixel to which $\mathbf{p}$ is projected to, in camera C. For X and Y, NID is defined as:

$$D(X, Y) = \frac{2H(X, Y) - H(X) - H(Y)}{H(X, Y)}, \quad (3)$$

$$\text{with,} \quad H(X) = -\sum_{x \in \mathcal{X}} p_x \log(p_x), \quad (4)$$

$$H(Y) = -\sum_{y \in \mathcal{Y}} p_y \log(p_y), \quad (5)$$

$$H(X, Y) = -\sum_{\substack{x \in \mathcal{X} \\ y \in \mathcal{Y}}} p_{xy} \log(p_{xy}). \quad (6)$$

In Eqns. (4), (5), and (6), p_x is the marginal pdf of X, p_y is the marginal pdf of Y, p_{xy} is the joint pdf of $\{X, Y\}$, and $H(X)$, $H(Y)$, and $H(X, Y)$ denote the entropy of X, the entropy of Y, and the joint entropy of $\{X, Y\}$, respectively. The symbols $\mathcal{X}$ and $\mathcal{Y}$ are the alphabets of X and Y.

NID is a true metric [17]. Thus, it is symmetric, non-negative, and bounded, i.e. $0 \leq D(X, Y) \leq 1$, where smaller values indicate greater similarity between the distributions. NID satisfies the triangle inequality, i.e. $D(X, Y) + D(Y, Z) \geq D(X, Z)$, and $D(X, Y) = 0 \iff X = Y$.

These properties make NID an attractive metric while designing optimisation problems. Posed as a localisation problem, the authors of [18] use the NID as a metric to match live camera images with a prior 3D map generated from cameras and laser data. Herein, we use the NID as a metric for the calibration problem.

B. The 'Ideal' Scene

In real-world environments, sensors are often exposed to signals that are not necessarily co-observed. The camera

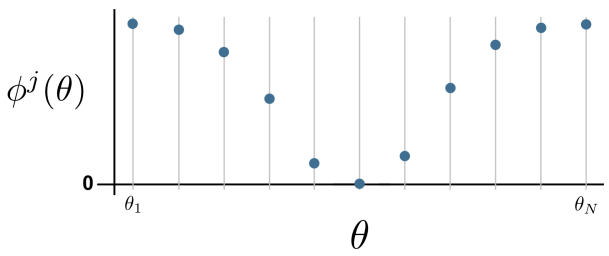

Fig. 4: Scene Descriptor: A visual representation of the scene descriptor $\phi^j(\theta)$ which is formed by sampling around the located calibration $\mathbf{g}_{CL}$, and finding the corresponding value according to Eqn. (7). This example corresponds to left-hand graph of Fig. 3.

observes light from secondary sources, which give rise to numerous irregularities like specular reflection, bloom, over-exposure and shadows, all of which will be absent from lidar reflectance (although lidars can encounter specular reflection from its own source). The immediate effect these have is on the mutual information between the lidar reflectance and camera image; NID increases, raising the latent minimum.

Instead of attempting to classify a scene based on its visual and geometric cues, which could differ drastically depending on the lidar-camera configuration, we opt for an analysis of the objective function that the lidar-camera pair produces. This is advantageous as it allows our method to generalise over any given sensor configuration or environment.

To represent the NID cost function produced by the j-th scene s^j as a discrete vector, we draw samples $\theta_{\{1,\dots,N\}}$ from the 6-DoF parameter space around the optimal $\mathbf{g}_{CL}$. We use these N samples to form a descriptor $\boldsymbol{\phi} \in \Re^N$, where the n-th element of $\boldsymbol{\phi}$ is:

$$\phi^j_n(\theta_n) = D^j\big(X, Y; \mathbf{g}_{CL} \oplus \theta_n\big) - D^j\big(X, Y; \mathbf{g}_{CL}\big). \quad (7)$$

The term $D^j(X, Y; \mathbf{g}_{CL})$ removes the NID offset to produce comparable descriptors for each scene, such that $\boldsymbol{\phi}(\mathbf{0}) = \mathbf{0}$. A visual representation of the resulting $\phi^j(\theta)$ from Eqn (7) is shown in Fig. 4. The descriptor aggregate $\bar{\boldsymbol{\phi}}$ is obtained by finding the mean over multiple scenes:

$$\bar{\boldsymbol{\phi}}(\boldsymbol{\theta}) = \frac{1}{S}\sum_{j=1}^{S} \boldsymbol{\phi}^j(\boldsymbol{\theta}). \quad (8)$$

Since $\bar{\boldsymbol{\phi}}$ is constructed using a sample of multiple frames, it gives us an insight into the nature of the data. Subsequently, for each scene, we compute a relative similarity measure $\mathcal{K}^j$ with respect to $\bar{\boldsymbol{\phi}}$. This score is obtained for each 6-DoF parameter, α, individually, and is defined as:

$$\mathcal{K}^j_\alpha(\bar{\boldsymbol{\phi}}_\alpha, \boldsymbol{\phi}^j_\alpha) = \frac{\bar{\boldsymbol{\phi}}_\alpha \cdot \boldsymbol{\phi}^j_\alpha}{\| \bar{\boldsymbol{\phi}}_\alpha \|^2}. \quad (9)$$

The scalar $\mathcal{K}^j_\alpha > 1$ indicates that s^j is likely closer to our 'ideal', and $\mathcal{K}^j_\alpha < 1$ represents a flatter, and less confident cost basin. The $\mathcal{K}$ value allows us to rank every scene in the dataset. We do this according to the minimum $\mathcal{K}$ of the α parameters for a single scene, so that:

$$\underline{\mathcal{K}}^j = \min(\mathcal{K}^j). \quad (10)$$

Thus, given $\mathbf{g}_{CL}$, and the subset S used to generate $\bar{\boldsymbol{\phi}}$, we can define the 'ideal' scene as being that which has the highest value of $\underline{\mathcal{K}}^j$. The selected scenes are used to perform calibration, which is described in the following section.

IV. Calibration Methodology

In this section, we recapitulate and extend the calibration optimisation process described in [14]. We show how the method from [14] can be applied when an existing sensor pose-graph is augmented with additional sensor nodes.

We assume that all sensors have known intrinsic parameters. To exploit the constraints afforded to us by a multi-camera setup, we measure the inter-camera transforms. Alternately, these may be provided by the manufacturer in the case of a stereo-camera, or by CAD models of the platform.

A. Calibration via Alternating Optimisation

Given a scene observed by camera C_p and lidar L, we seek to minimise the dissimilarity between the pixel intensity values associated with the observed scene in camera C_p, and the reflectance values associated with the lidar points measured by the lidar L, projected in the p-th camera's image plane. Let $\{C_a\}$ and $\{C_b\}$ be the frames associated with each camera of a stereo camera unit, and let $\{M_a\}$ be the frame associated with an additional camera (see Fig. 2).

The constrained optimisation problem to calibrate a lidar to the camera system described above is defined as:

$$\begin{aligned}
&\min_{\substack{\mathbf{g}_{C_aL}, \mathbf{g}_{C_bL}, \\ \mathbf{g}_{M_aL}}} && f_a\big(\mathbf{g}_{C_aL}\big) + f_b\big(\mathbf{g}_{C_bL}\big) + f_m\big(\mathbf{g}_{M_aL}\big) \\
&\text{s.t.} && \mathbf{g}_{C_aL} \ominus \mathbf{g}_{C_bL} = \bar{\mathbf{g}}_{C_aC_b}, \text{ and} \\
& && \mathbf{g}_{C_aL} \ominus \mathbf{g}_{M_aL} = \bar{\mathbf{g}}_{C_aM_a}, \\
&\text{where,} && f_a \equiv \frac{1}{S_a}\sum_{s_a=1}^{S_a} D^{s_a}\big(X_a, Y_a; \mathbf{g}_{C_aL}\big), \\
& && f_b \equiv \frac{1}{S_b}\sum_{s_b=1}^{S_b} D^{s_b}\big(X_b, Y_b; \mathbf{g}_{C_bL}\big), \\
& && f_m \equiv \frac{1}{S_m}\sum_{s_m=1}^{S_m} D^{s_m}\big(X_m, Y_m; \mathbf{g}_{M_aL}\big).
\end{aligned} \quad (11)$$

The terms X_a, X_b, X_m, Y_a, Y_b, and Y_m denote measurements of the discrete random variables X and Y made by, or projected to, the camera indicated by the subscripts a, b, and m, respectively. As explained in III, the scalars S_a, S_b, and S_m are the number of diligently selected images used to widen the cost function convergence basin [11], [12]. Each camera may be provided a different set of scenes. The transforms $\bar{\mathbf{g}}_{C_aC_b}$ and $\bar{\mathbf{g}}_{C_aM_a}$ are assumed to be known, and $\mathbf{g}_{C_aL}$, $\mathbf{g}_{C_bL}$, and $\mathbf{g}_{M_aL}$ are the transforms to be optimised.

In practice, $\bar{\mathbf{g}}_{C_aC_b}$ and $\bar{\mathbf{g}}_{C_aM_a}$ are obtained from the manufacturer, and may be known to a pre-defined uncertainty measure. We utilise this to terminate the equality constraints in (11) via quadratic penalty terms, and derive the following

unconstrained optimisation problem:

$$\begin{aligned}
&\underset{\substack{\mathbf{g}_{C_aL},\mathbf{g}_{C_bL},\\ \mathbf{g}_{M_aL}}}{\text{min.}} && f_a\big(\mathbf{g}_{C_aL}\big) + f_b\big(\mathbf{g}_{C_bL}\big) + f_m\big(\mathbf{g}_{M_aL}\big) + \mathcal{E} \\
&\text{where,} && \mathcal{E} = \big\|\mathbf{e}_{ab}\big\|^2_{\mathbf{P}_{ab}} + \big\|\mathbf{e}_{am}\big\|^2_{\mathbf{P}_{am}} \\
&\text{with,} && \mathbf{e}_{ab} = (\mathbf{g}_{C_aL} \ominus \mathbf{g}_{C_bL}) \ominus \bar{\mathbf{g}}_{C_aC_b}, \text{ and} \\
& && \mathbf{e}_{am} = (\mathbf{g}_{C_aL} \ominus \mathbf{g}_{M_aL}) \ominus \bar{\mathbf{g}}_{C_aM_a}.
\end{aligned} \tag{12}$$

The scalars $\|\mathbf{e}_{ab}\|^2_{\mathbf{P}_{ab}} = \mathbf{e}_{ab}^\top [\mathbf{P}_{ab}]^{-1}\mathbf{e}_{ab}$, and $\|\mathbf{e}_{am}\|^2_{\mathbf{P}_{am}} = \mathbf{e}_{am}^\top [\mathbf{P}_{am}]^{-1}\mathbf{e}_{am}$, are the squared *Mahalanobis* distances parameterised by normal distributions $\mathcal{N}(\mathbf{0}, \mathbf{P}_{ab})$ and $\mathcal{N}(\mathbf{0}, \mathbf{P}_{am})$, respectively. If the covariances $\mathbf{P}_{ab}$, $\mathbf{P}_{am} = \mathbf{0}$, then the problem in Eqn. (12) satisfies the strict equality constraints from Eqn. (11).

As explained in [14], the mutual independence of f_a, f_b, and f_m can be used to implement fast, hierarchical optimisation by decoupling Eqn. (12) into sub-problems to be solved on different CPU nodes.

To decouple Eqn. (12), we augment it with additional variables and constraints, and define:

$$\begin{aligned}
&\underset{\substack{\mathbf{g}_{C_aL},\mathbf{g}_{C_bL},\mathbf{g}_{M_aL},\\ \hat{\mathbf{g}}_{C_aL},\hat{\mathbf{g}}_{C_bL},\hat{\mathbf{g}}_{M_aL}}}{\text{min.}} && f_a\big(\hat{\mathbf{g}}_{C_aL}\big) + f_b\big(\hat{\mathbf{g}}_{C_bL}\big) + f_m\big(\hat{\mathbf{g}}_{M_aL}\big) + \mathcal{E} \\
&\text{s.t.} && \hat{\mathbf{g}}_{C_aL} = \mathbf{g}_{C_aL},\ \hat{\mathbf{g}}_{C_bL} = \mathbf{g}_{C_bL}, \text{ and} \\
& && \hat{\mathbf{g}}_{M_aL} = \mathbf{g}_{M_aL}.
\end{aligned} \tag{13}$$

Using penalty terms to terminate the equality constraints in (13), we derive another unconstrained optimisation problem:

$$\begin{aligned}
\underset{\substack{\mathbf{g}_{C_aL},\mathbf{g}_{C_bL},\mathbf{g}_{M_aL},\\ \hat{\mathbf{g}}_{C_aL},\hat{\mathbf{g}}_{C_bL},\hat{\mathbf{g}}_{M_aL}}}{\text{min.}} \quad & \mathcal{E}\big(\mathbf{g}_{C_aL}, \mathbf{g}_{C_bL}, \mathbf{g}_{M_aL} : \bar{\mathbf{g}}_{C_aC_b}, \bar{\mathbf{g}}_{M_aC_b}\big) \\
& + f_a\big(\hat{\mathbf{g}}_{C_aL}\big) + \big\| \underbrace{\hat{\mathbf{g}}_{C_aL} \ominus \mathbf{g}_{C_aL}}_{\mathbf{e}_{aL}} \big\|^2_{\mathbf{P}^i_{aL}} \\
& + f_b\big(\hat{\mathbf{g}}_{C_bL}\big) + \big\| \underbrace{\hat{\mathbf{g}}_{C_bL} \ominus \mathbf{g}_{C_bL}}_{\mathbf{e}_{bL}} \big\|^2_{\mathbf{P}^i_{bL}} \\
& + f_m\big(\hat{\mathbf{g}}_{M_aL}\big) + \big\| \underbrace{\hat{\mathbf{g}}_{M_aL} \ominus \mathbf{g}_{M_aL}}_{\mathbf{e}_{mL}} \big\|^2_{\mathbf{P}^i_{mL}}
\end{aligned} \tag{14}$$

In (14), the error terms $\mathbf{e}_{aL}$, $\mathbf{e}_{bL}$ and $\mathbf{e}_{mL}$ are parameterised by $\mathcal{N}(\mathbf{0}, \mathbf{P}^i_{aL})$, $\mathcal{N}(\mathbf{0}, \mathbf{P}^i_{bL})$, and $\mathcal{N}(\mathbf{0}, \mathbf{P}^i_{mL})$, respectively. The superscript i on $\mathbf{P}^i_{aL}$, $\mathbf{P}^i_{bL}$, and $\mathbf{P}^i_{mL}$ is an optimisation iteration index that forces $\mathbf{P}^i_{aL}, \mathbf{P}^i_{bL}, \mathbf{P}^i_{mL} \to \mathbf{0}$ as $i \to \infty$. This design choice emphasises feasible solutions. The extremity of the penalty levied on infeasible solutions is determined by $\mathbf{P}^i_{aL}$, $\mathbf{P}^i_{bL}$, and $\mathbf{P}^i_{mL}$ [19]. As $\mathbf{P}^i_{aL}$, $\mathbf{P}^i_{bL}$, and $\mathbf{P}^i_{mL}$ decrease, the unconstrained problem in (14) accurately replicates the constrained problem in (13).

Eqn. (14) is solved using an alternating optimisation algorithm [20]. For each camera C_p, the penalty term $\|\mathbf{e}_{pL}\|^2_{\mathbf{P}^i_{pL}}$ is a function of $\mathbf{g}_{C_pL}$ and $\hat{\mathbf{g}}_{C_pL}$. Thus, to solve (14), we use the penalty terms $\|\mathbf{e}_{aL}\|^2_{\mathbf{P}^i_{aL}}$, $\|\mathbf{e}_{bL}\|^2_{\mathbf{P}^i_{bL}}$, and $\|\mathbf{e}_{mL}\|^2_{\mathbf{P}^i_{mL}}$ to link the two alternating optimisation levels.

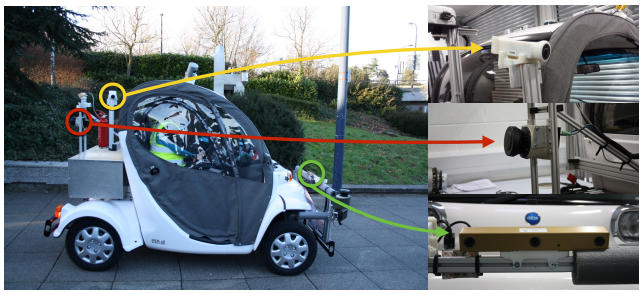

Fig. 5: Setup: The sensors to be calibrated mounted on the platform such that they do not have overlapping FoV. We use the stereo camera mounted at the front of the vehicle to estimate vehicle pose using VO. The data used for calibration was collected in Milton Keynes, UK.

1) Lower-level Optimisers: For each $p = 1 \dots N_C$, independent optimisers first optimise $\hat{\mathbf{g}}_{C_pL}$ by solving:

$$\underset{\hat{\mathbf{g}}_{C_pL}}{\text{min.}}\ f_p\big(\hat{\mathbf{g}}_{C_pL}\big) + \big\|\mathbf{e}_{pL}\big(\hat{\mathbf{g}}_{C_pL} : \mathbf{g}_{C_pL}\big)\big\|^2_{\mathbf{P}^i_{pL}} \tag{15}$$

Eqn. (15) is one of three sub-problems decoupled from Eqn. (14), i.e. the second, third, or fourth row of (14), based on the camera index. We name Eqn. (15) as a lower-level optimiser. There are $N_C N_L$ lower-level optimisers, one for each lidar-camera pair. While solving (15), $\mathbf{g}_{C_pL}$ is held constant.

2) Upper-level Optimiser: The problem defined in (14) is decoupled into three lower-level optimisers, which provide optimised solutions of $\hat{\mathbf{g}}_{C_aL}$, $\hat{\mathbf{g}}_{C_bL}$, and $\hat{\mathbf{g}}_{M_aL}$. We utilise these lower-level solutions as known constants and define the upper-level optimiser as:

$$\begin{aligned}
\underset{\substack{\mathbf{g}_{C_aL},\mathbf{g}_{C_bL},\\ \mathbf{g}_{M_aL}}}{\text{min.}} \quad & \mathcal{E}\big(\mathbf{g}_{C_aL}, \mathbf{g}_{C_bL}, \mathbf{g}_{M_aL} : \bar{\mathbf{g}}_{C_aC_b}, \bar{\mathbf{g}}_{M_aC_b}\big) \\
& + \big\|\mathbf{e}_{aL}\big(\mathbf{g}_{C_aL} : \hat{\mathbf{g}}_{C_aL}\big)\big\|^2_{\mathbf{P}^i_{aL}} \\
& + \big\|\mathbf{e}_{bL}\big(\mathbf{g}_{C_bL} : \hat{\mathbf{g}}_{C_bL}\big)\big\|^2_{\mathbf{P}^i_{bL}} \\
& + \big\|\mathbf{e}_{mL}\big(\mathbf{g}_{C_mL} : \hat{\mathbf{g}}_{C_mL}\big)\big\|^2_{\mathbf{P}^i_{mL}}
\end{aligned} \tag{16}$$

Note that (16) is also a sub-problem decoupled from Eqn. (14). The alternating optimisation formulation is used to solve the problem in (14) as a set of hierarchical, closed-loop, and sequential optimisation problems. The solutions of the lower-level optimisers drive the upper-level optimiser, and vice versa. These equations can naturally be extended to include any number of sensors.

V. Experiments and Results

This section details the results of the *alternating calibration* method coupled with *diligent scene selection*. The robotic platform and its configuration is also explained.

A. Setup

The platform in Fig. (5) has a PointGrey Research (PGR) Bumblebee XB3 stereo camera mounted in the front, facing forward, and tilted downward by approximately $18°$. The SICK LMS-151 2D lidar is positioned at the back of the vehicle, such that it is tilted back by approximately $9°$. The lidar scans are generated in a plane that is offset by

Fig. 6: 'Ideal' Scenes: Top row shows five scenes with highest $\underline{\mathcal{K}}^j$ values. These are the 'ideal' scenes. Bottom row shows five scenes with lowest $\underline{\mathcal{K}}^j$.

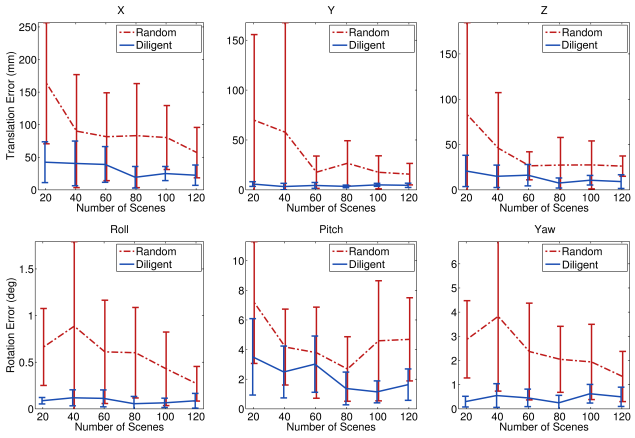

Fig. 7: Scene Selection Evaluation: Mean squared error exhibited when using a randomly selected subset of scenes, and when using our opportunistic scene selection method. Results are shown for each 6-DoF parameter individually. Error bars indicate $\pm 1\sigma$.

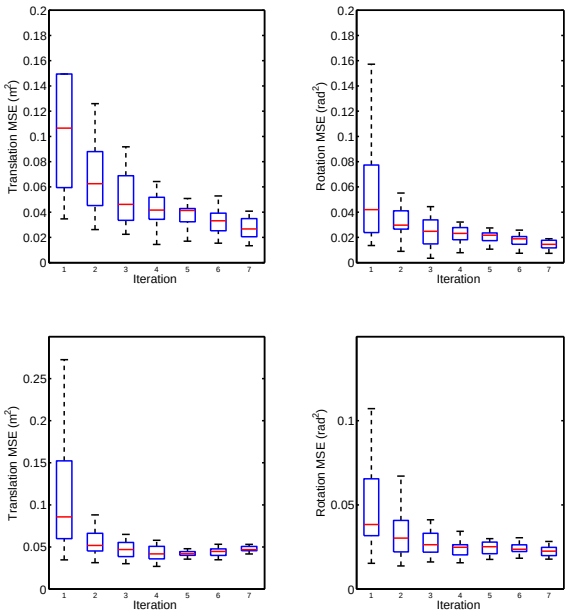

Fig. 8: Error Analysis: Box plots showing per iteration MSE distribution in translation (left) and rotation (right). The top and bottom rows indicate the laser-to-camera error observed for calibration estimates $\mathbf{g}_{C_a L}$ (Bumblebee left) and $\mathbf{g}_{M_a L}$ (Grasshopper) respectively. The lower and upper bounding boxes indicate the 25^{th} and 75^{th} percentiles; the red line within the box is the median, and the dashed line indicates the extent of the extremes.

approximately $9°$ with respect to the plane, orthogonal to the direction of horizontal planar motion. The point cloud generated using such a configuration is illustrated in Fig. 1.

The vehicle has two PGR Grasshopper monocular cameras on the sides, one facing right (highlighted by a solid yellow arrow in Fig. 5), and the other facing left (not shown in Fig. 5). No sensors have overlapping fields of view.

PGR provide sub-millimetre accuracy for the inter-camera transform between the individual cameras of the stereo unit [21]. We manually measure the inter-camera transforms between the camera-base frame (left camera of the stereo unit) and the individual monocular cameras, and use the measured transform as a known parameter with a measure of uncertainty. As explained in IV, we can exploit an inter-camera transform provided by the manufacturer or otherwise, as a known parameter to perform lidar-camera calibration.

For evaluating the proposed method, we obtain reference values by using an accurate (less than 0.5mm translation error) motion tracking system [22] to locate the plane of a checkerboard observed by each sensor. Thus, the lidar and each camera of a multi-camera unit can both be calibrated very accurately with respect to the frame of the motion tracker. In Sec. V-C, we use these values as 'ground truth'.

B. Scene Selection Evaluation

The scene selection procedure described in III is performed once at the beginning of the calibration procedure. The results are obtained using a 10km road dataset and approximately 1000 individual scenes.

1) Accuracy and variance: We evaluate the scene selection method by comparing it with a random selection process. Ground truth in this experiment is the accurately known inter-camera calibration of the Bumblebee stereo camera. The calibration of the laser to the left and right cameras respectively are used to estimate this transform which we compare to that supplied by the manufacturer.

The calibration error for translation and rotation, with respect to ground truth, is presented in Fig. 7. The mean squared error and variance are computed for both random and *diligent* selection, repeated for increasing scene subset

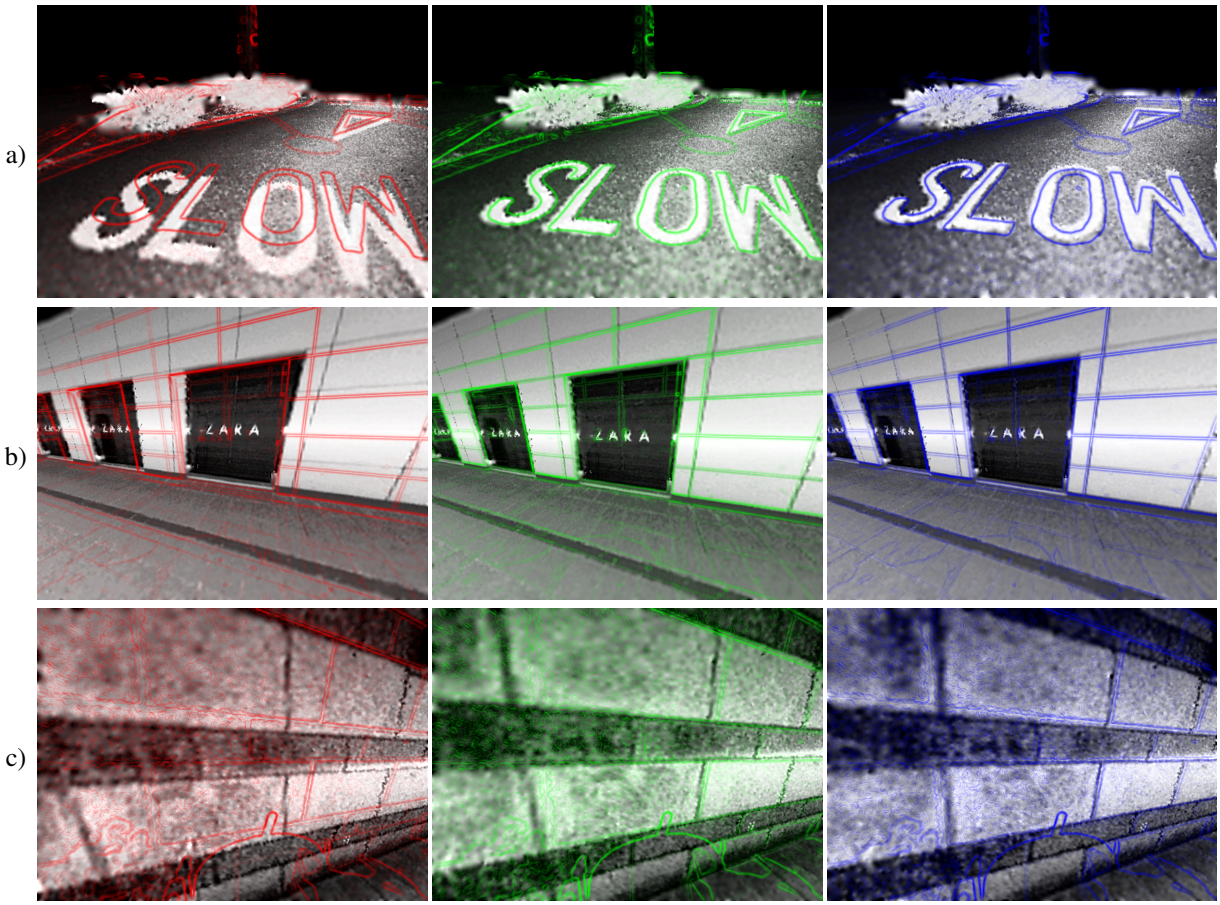

Fig. 9: Edges of a scene image superimposed on a synthetic image constructed using a projected point cloud. Row correspond to a different camera used; a) front stereo camera, $\mathbf{g}_{C_a L}$ b) right camera, $\mathbf{g}_{M_a L}$ and c) left camera, $\mathbf{g}_{M_b L}$. The columns correspond to the calibration parameters of a random initial seed, using our proposed framework and of the 'ground truth', respectively.

sizes. The N *diligently* selected scenes are those from the dataset that have the highest scene score $\underline{\mathcal{K}}^j$. We observe that *diligently* selecting scenes always produces a more accurate calibration. Moreover, Fig. 7 shows that only a few scenes are required to substantially improve the calibration over using a large random subset.

2) Scene scoring: In Eqn. (10), we introduced a measure $\mathcal{K}^j$ which ranks scenes according to the shape of the cost function they produce. Fig. 6 shows the top five and bottom five according to $\underline{\mathcal{K}}^j$. The top five scenes exhibit features with strong edges and high contrast while the opposite can be said of the bottom five.

C. Calibration Performance Evaluation

Following the procedure in II-B and III, we generate point clouds for 12 different diligently selected scenes for each camera, each 10m long. Fig. 8 shows the results from an optimisation, with a fixed number iterations for the lower and upper-level optimisers. The lower-level optimiser runs for 200 iterations, and the upper-level for 7 iterations.

The covariance $\mathbf{P}_{pL}$ for camera p is initialised with standard deviations of $\sigma_t = 1$m and $\sigma_\phi = 1.5$ radians for the translation and the rotation parameters, respectively. These covariances are used in the quadratic penalty in Eqn. (14) to link the upper and lower levels, and are initialised with relatively high values to simulate low initial confidence in the lower-level. In accordance with [19], these covariances are updated at each iteration i such that $\mathbf{P}_{pL}^{i+1} = \omega\ \mathbf{P}_{pL}^{i}$, with $\omega \in (0, 1)$.

The performance of our proposed method, when compared to our motion capture system's 'ground truth', is presented in Figure 8. The variance and the mean square error (MSE) are computed by performing *alternating* calibration with 30 different initial seeds for the calibration parameters. Rotational error is calculated using a rotational difference metric [23]. The top and bottom rows correspond to laser-to-camera calibration for the left Bumblebee camera and right Grasshopper respectively. With each iteration, the error and uncertainty decreases until convergence, after approximately seven iterations. This result is representative of the other cameras in the test system.

Fig. 9 provides a qualitative illustration of the solution computed by the *alternating* method. Each image is a synthetic image generated by projecting the point cloud into the camera frame with given calibration parameters. The pixels onto which the points fall are coloured with the laser reflectance values. The actual camera image is processed

with an edge filter, the result of which is superimposed over the synthetic image. The alignment of edges provides visual aid for gauging the quality of camera-laser calibration. The left column shows the result obtained using an initial seed. The middle column shows the alignment using a solution estimated by our *alternating* and *scene selection* methods. The right column provides visual conformation of the validity of our methods by showing the alignment given by the 'ground truth' calibration. Each row depicts a scene observed from the from stereo camera, and the right and left monocular cameras, respectively.

VI. Conclusion and Future Work

In previous work we proposed a target-less, automatic, data-driven method for calibrating a 2D push-broom lidar to a stereo camera, using a hierarchical, closed-loop, alternating optimisation algorithm, distributed over two optimisation levels. The lower-level solves camera-to-laser registration using a *Normalised Information Distance*-based (NID) cost functions, while the upper-level implements a nonlinear least-squares pose-graph refinement step.

In this paper we present an extension to that framework, which allows multiples cameras to be be folded into a single calibration procedure. We show the advantages of the upper level graph optimisation to calibrate specific sensor configurations with no overlapping observations of the same local scene. As additional contribution, we designed a data-driven approach that discerns the NID cost function generated by a different scenes and rank individual scenes using a similarity measure. We propose a cost descriptor that implicitly characterise the intrinsic information of the gathered observations. Our experiments show how the selected scenes can optimise the effectiveness of the calibration results providing more accurate estimates using a few scenes.

We provide a detailed performance analysis after running the optimisation on real-world data collected over hundreds of meters. Performance of the proposed method is evaluated against calibration parameters obtained from a highly-accurate commercial calibration system.

As for most real-world applications, the function to be minimised for the proposed method is only locally convex, and is data-dependent. Each image used for calibration may provide a different amount of information. This can affect the basin of convergence for the selected cost function. Thus, learning a calibration cost function from the data, and utilising information-based image selection and weighting schemes are interesting problems for future research.

In [12], calibration is performed by creating a synthetic lidar image through interpolation of lidar reflectance values. This lidar reflectance interpolation step is computationally expensive, and we believe, is unsuitable for extension to online calibration approaches. Extending the proposed method to an online implementation is within future scope.

References

[1] Q. Zhang and R. Pless, "Extrinsic calibration of a camera and laser range finder (improves camera calibration)," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 3. IEEE, 2004, pp. 2301–2306.

[2] L. Huang and M. Barth, "A novel multi-planar lidar and computer vision calibration procedure using 2d patterns for automated navigation," in *IEEE Intelligent Vehicles Symposium (IVS)*, Xi'an, China, June 2009, pp. 117–122.

[3] O. Naroditsky, A. Patterson, and K. Daniilidis, "Automatic alignment of a camera with a line scan lidar system," in *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, May 2011, pp. 3429–3434.

[4] G. Pandey, J. R. McBride, S. Savarese, and R. Eustice, "Extrinsic calibration of a 3d laser scanner and an omnidirectional camera," in *IFAC Symposium on Intelligent Autonomous Vehicles (IAV)*, Sep 2010.

[5] R. Unnikrishnan and M. Hebert, "Fast extrinsic calibration of a laser rangefinder to a camera," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Carnegie Mellon University, 2005.

[6] F. M. Mirzaei, D. G. Kottas, and S. I. Roumeliotis, "3d lidar-camera intrinsic and extrinsic calibration: Identifiability and analytical least-squares-based initialization," in *The International Journal of Robotics Research (IJRR)*, Sep 2012, pp. 452–467.

[7] A. Geiger, F. Moosmann, O. Car, and B. Schuster, "Automatic camera and range sensor calibration using a single shot," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 3936–3943.

[8] D. Scaramuzza, A. Harati, and R. Siegwart, "Extrinsic self calibration of a camera and a 3d laser range finder from natural scenes," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2007, pp. 4164–4169.

[9] L. Quan and Z. Lan, "Linear n-point camera pose determination," in *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 21, no. 8, Xi'an, China, Aug 1999, pp. 774–780.

[10] J. Levinson and S. Thrun, "Automatic online calibration of cameras and lasers." in *Robotics: Science and Systems*, 2013.

[11] G. Pandey, J. R. McBride, S. Savarese, and R. Eustice, "Automatic extrinsic calibration of vision and lidar by maximizing mutual information," in *Journal of Field Robotics (JFR)*, Sep 2014, pp. 1–27.

[12] A. Napier, P. Corke, and P. Newman, "Cross-calibration of push-broom 2d lidars and cameras in natural scenes," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, Karlsruhe, Germany, May 2013.

[13] R. Wang, F. P. Ferrie, and J. Macfarlane, "Automatic registration of mobile lidar and spherical panoramas," in *Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 2012, pp. 33–40.

[14] T. Scott, A. A. Morye, P. Piniés, L. M. Paz, I. Posner, and P. Newman, "Exploiting Known Unknowns: Scene Induced Cross-Calibration of Lidar-Stereo Systems," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, Germany, 2015.

[15] S. Boyd and L. Vandenberghe, "Duality," in *Convex Optimization*, 2004, pp. 215–287.

[16] I. Baldwin and P. Newman, "Localising transportable apparatus," Apr 2013, wO Patent App. PCT/GB2012/052,381. [Online]. Available: http://www.google.com/patents/WO2013045917A1?cl=en

[17] T. M. Cover and J. A. Thomas, "Entropy, relative entropy, and mutual information," in *Elements of Information Theory*, vol. 2, 1991, pp. 1–55.

[18] A. Stewart and P. Newman, "Laps - localisation using appearance of prior structure: 6-dof monocular camera localisation using prior pointclouds," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, Minnesota, USA, May 2012.

[19] D. P. Bertsekas, "The method of multipliers for equality constrained problems," in *Constrained Optimization and Lagrange Multiplier Methods*, 1996, pp. 95–157.

[20] J. C. Bezdek and R. J. Hathaway, "Convergence of alternating optimization," in *Neural, Parallel and Scientific Computations (NPSC)*, vol. 11, no. 4, Dec 2003, pp. 351–368.

[21] P. G. Research. (2014) Bumblebee2 and xb3 datasheet. [Online]. Available: http://www.ptgrey.com/support/downloads/10132/

[22] (2014) Phoenix technologies inc. vz4000. [Online]. Available: http://www.ptiphoenix.com/?prod-trackers-post=vz4000

[23] D. Q. Huynh, "Metrics for 3d rotations: Comparison and analysis," *Journal of Mathematical Imaging and Vision*, vol. 35, no. 2, pp. 155–164, 2009.