Line-based Extrinsic Calibration of Range and Image Sensors

Peyman Moghadam, Michael Bosse, Robert Zlot

Fig. 1: Overview of the proposed method: lines are extracted from (a) an image and (b) from a point cloud of the same scene. (c) By registering 2D and 3D lines the corresponding photorealistic representation can be generated.

Abstract—Creating rich representations of environments requires integration of multiple sensing modalities with complementary characteristics such as range and imaging sensors. To precisely combine multisensory information, the rigid transformation between different sensor coordinate systems (i.e., extrinsic parameters) must be estimated. The majority of existing extrinsic calibration techniques require one or multiple planar calibration patterns (such as checkerboards) to be observed simultaneously from the range and imaging sensors. The main limitation of these approaches is that they require modifying the scene with artificial targets. In this paper, we present a novel algorithm for extrinsically calibrating a range sensor with respect to an image sensor with no requirement of external artificial targets. The proposed method exploits natural linear features in the scene to precisely determine the rigid transformation between the coordinate frames. First, a set of 3D lines (plane intersection and boundary line segments) are extracted from the point cloud, and a set of 2D line segments are extracted from the image. Correspondences between the 3D and 2D line segments are used as inputs to an optimization problem which requires jointly estimating the relative translation and rotation between the coordinate frames. The proposed method is not limited to any particular types or configurations of sensors. To demonstrate robustness, efficiency and generality of the presented algorithm, we include results using various sensor configurations.

I. INTRODUCTION

In recent years, there has been tremendous progress in the development of mobile mapping systems. Such systems have many potential applications ranging from robotics (e.g., SLAM), security (change detection), emergency response (disaster sites), to gaming (virtual 3D reality), and virtual tourism (E-Heritage conservation).

Current mobile mapping systems are primarily based on range sensors such as lidar. These systems can acquire three-dimensional geometric data about a scene with high accuracy and high resolution in day or night. In addition, some lidar scanners can report the surface reflectivity values at the wavelength of the laser beam. However, such unimodal systems typically do not provide rich and consistent visual appearance information. One possibility is to combine 3D data with alternative sensing modalities (e.g., visual, thermal infrared, hyperspectral, etc.) to create richer environment models. Furthermore, by integrating multisensory information the reliability and accuracy of mobile mapping systems can be improved as a result of the complementary characteristics of each sensing modality.

In order to precisely fuse information obtained from different sensing modalities, it is critical to represent them in a common reference frame. For such a purpose, the rigid transformation between different sensor coordinate systems should be obtained. The process of estimating the six degree of freedom (6DoF) transformation (i.e., the relative rotation and translation) between sensor coordinate systems is called the extrinsic calibration.

The extrinsic range-image calibration techniques can be categorized into two main groups. Most of the existing techniques require external artificial targets to be observed simultaneously from the range and imaging sensors. A standard external target is the planar calibration pattern (i.e., a checkerboard pattern), which is commonly used for intrinsic camera calibration [1], [2], [3], [4], [5], [6]. This approach usually requires views of the checkerboard from several poses for an accurate extrinsic calibration result.

Zhang and Pless [1] first proposed the extrinsic calibration of a perspective camera and 2D laser range finder using a checkerboard. This approach uses point-to-plane geometric constraints for extrinsic calibration. It parameterizes the calibration pattern by its unit normal vector and its distance to the camera coordinate frame. Since the laser points should lie on this plane, the problem can be refined as minimizing the reprojection errors and solved iteratively using non-linear optimization solutions.
The Unnikrishnan and Herbert method [2] solves the extrinsic calibration problem using plane-to-plane geometric constraints. A plane is fitted to the checkerboard in each coordinate frame and parameterized as a normal vector and a distance with respect to its origin coordinate frame. Later, the rigid body transformation between the two reference frames is estimated by minimizing the difference in orientation and distance of the planes observed in each of the two coordinate systems.

The major drawback of these approaches is that they require the placement of artificial calibration targets in the scene. Moreover, a large set of images and range measurements has to be acquired at different poses of the calibration target to precisely estimate the rigid transformation between the imaging and range sensors.

A second group of extrinsic camera-laser calibration techniques do not require any planar calibration pattern and use natural scene features to determine the rigid transformation between the coordinate frames [7], [8], [9]. The algorithm we propose in this paper belongs to this group of extrinsic calibration techniques. Scaramuzza et al. [7] proposed an extrinsic calibration technique for a 3D laser range finder and an omnidirectional camera based on point correspondences from natural scenes. This approach requires the user to manually select a set of point correspondences (point-to-point geometric constraints) in images and their associated range data. Once the point correspondences are known, the rigid transformation between the two reference frames can be determined using the well-known PnP (Perspective from n Points) algorithm followed by iterative least-squares refinement. However, it is much more difficult to manually identify 3D point features in range data than to identify their correspondences in the images. On the other hand, due to uncertainties in the range data, the 3D position of a selected point may contain measurement errors, which degrades the reliability of the calibration procedure.

Recently, Pandey et al. [8] proposed an automatic, targetless extrinsic calibration method for a 3D laser scanner and a camera. The calibration algorithm uses a mutual information (MI) framework to estimate the statistical dependence between the laser scanner reflectivity values and intensity information from the camera image data (point-to-point geometric constraints). This approach cannot apply directly to range sensors without associated reflectivity information.

Stomas et al. [9] present an alternative approach by decoupling the estimation of camera translation from rotation with respect to the range data. Rotation is estimated by matching 2D image vanishing points with major scene directions extracted from 3D range data (point-to-point geometric constraints). Next, the camera translation with respect to the range sensor is calculated by matching the corresponding clusters of parallel 2D and 3D linear features. This approach [9] bears the most similarity to the method proposed in this paper. However, it is mainly restricted to urban environments that have at least two major vanishing points in the scenes (e.g., building facades).

To address the abovementioned shortcomings, we propose a novel targetless extrinsic calibration method between 3D range and 2D imaging systems. The proposed method uses line-to-line geometric constraints to precisely determine the rigid transformation between the coordinate frames. We formulate the extrinsic calibration as an optimization problem between line segments extracted in the 3D point clouds and 2D images.

The contributions of this work can be summarized as follows: We present a novel extrinsic range-image calibration technique which exploits strong line-to-line geometric constraints to jointly estimate the 6DoF transformation between the coordinate systems. The proposed method requires no special calibration pattern: a single image and range acquisition with enough linear features is sufficient (discussed in Section II). Our method is not limited to any particular types or configurations of range and imaging sensors. We demonstrate results using various sensor configurations in Section III. The proposed algorithm has been already utilized to estimate the relative rigid transformation between range sensor and thermal infrared image sensing modality for 3D thermal mapping applications [10].

The remainder of this paper is organized as follows. In Section II, we explain the methodology of the proposed 2D-3D registration based on the line-to-line geometric constraints. Section III demonstrates the accuracy, effectiveness, and generality of the proposed technique based on the results from experiments on several datasets. Finally, we conclude our paper in Section IV.

II. METHODOLOGY

The proposed extrinsic calibration algorithm is comprised of three main components. The first component extracts 3D lines (plane intersection and boundary lines) from the point cloud. Then, edge detection followed by line extraction are performed to find 2D line segments in the input image. Only 2D line segments corresponding to the 3D lines are retained. Given a set of 3D and 2D line correspondences, the proposed algorithm uses a nonlinear least squares optimization to find a transformation that minimizes the distance between the projected 3D line points and the image lines. Figure 1 provides an overview of the system. In the following sections we present each step in more detail.

A. 2D LINE EXTRACTION

For 2D image line extraction, we employ a Canny edge detector followed by standard morphological operations (i.e., dilation and erosion) to extract dominant edge features in the input images. The next step is to determine if there are any line segments in the image, which are extracted from the edge features using a Hough Transform. The algorithm ignores any line segments that are shorter than a predefined threshold. An example result is illustrated in Figure 1a.

B. 3D LINE EXTRACTION

Two types of 3D lines are extracted from the point cloud data: line segments at the intersections of pairs of planes, and linear features at the boundaries of planes. Plane intersections
are determined in a similar manner to our approach in a previous publication [11]. First, planar surfaces are estimated using a region growing process, whereby local neighborhoods are formed around each lidar point using a \( kd \)-tree fixed radius search algorithm. A surface normal and local planarity metric for the point neighborhood can be computed through eigenanalysis of the second-order moments matrix. The region growing algorithm then proceeds to merge adjacent surfaces if the surface normal directions and along-normal (point-to-plane) distances are within defined tolerances. Lines of intersection are then computed for adjacent pairs of planes containing a sufficient number of points. The endpoints of each line segment are defined according to the extents of the points from the two planes projected onto the line.

Plane intersection lines are not always sufficient to reliably determine a scale for calibration. For instance, in the case where all of the plane intersection lines belong to a pencil of lines intersecting at the corner point of three intersecting planes, the translation towards the intersection point would not be observable.

Such ambiguities can be mitigated by adding additional lines, such as those at range discontinuities (or boundary lines), and generally resulting in a more accurate and reliable calibration. The boundary line segments for a given plane are the edges from the convex polygon obtained by forming the 2D convex hull of the point set (projected onto the plane). Edges close to plane intersection lines are discarded from the boundary line segment set. An example result of the 3D line extraction algorithm is shown in Figure 1b.

C. Registration

Given a corresponding set of 2D image lines, and 3D world lines, the goal is to find the 6DoF pose transformation of the image with respect to the world. At a high level, we compute an error vector using each line correspondence and use a nonlinear least squares optimization to find a transformation that minimizes the \( L_2 \) norm of the error vector. We assume that the endpoints of the lines do not contain reliable information (due to occlusions, camera field of view clipping, and coarse laser resolution), therefore the error metric is mostly invariant to the positions of the endpoints on the lines. However, we use the endpoints of the lines to parameterize the lines and determine the points from which to measure the error between the image and world line. Each 2D image line is represented by its endpoints \( m_1 \) and \( m_2 \) which are 3-dimensional homogeneous vectors \( (x, y, 1)^T \) in the image plane. The line equation:

\[
I^T m = d
\]

defines the line where \( d = 0 \), \( m \) is any point on the line and \( I \) can be computed from the endpoints (or any two distinct points on the line) as \( I = m_1 \times m_2 \). When \( I \) is normalized such that its first two components are a unit vector, then Equation 1 computes the signed distance \( d \) of a point \( m \) to the line \( I \).

The 3D world lines are similarly represented by their endpoints \( M_1 \) and \( M_2 \), which are 4D homogeneous vectors \((x, y, z, 1)^T\). Any point on the 3D line can be parameterized by a linear combination of the two endpoints:

\[
M = \alpha M_1 + (1 - \alpha) M_2
\]

The 3D point can then be projected into the image plane using the camera intrinsic matrix \( K \) and the current estimate of the camera’s pose \((R, T)\) with respect to the world frame:

\[
m_p = K [R^T, -R^T T] M
\]

The projected point is normalized by its third component such that the first two components are in units of pixels.

The first step in computing the error metric is to find the points on the 3D lines that are projected closest to the endpoints of the 2D image line, by solving for \( \alpha \):

\[
m = m_{p1} \alpha + m_{p2} (1 - \alpha)
\]

\[
\alpha = \frac{(m_{p1} - m_{p2})^T (m - m_{p2})}{\|m_{p1} - m_{p2}\|^2}
\]

To prevent 3D line points from being extrapolated outside of the endpoints, we clamp \( \alpha \) to the range \([0, 1]\).

The two elements in the error vector \( E \) for each line correspondence are then defined as the distance between the projected 3D line points and the image line:

\[
E_{ij} = \frac{I_j^T m_{pij}}{\|I_j\|^2} m_{pij}
\]

\[
m_{pij} = K [R^T, -R^T T] (\alpha_j M_{i1} + (1 - \alpha_j) M_{i2})
\]

where \( i \) indexes the correspondence, and \( j \in \{1, 2\} \) indexes the line endpoint, and \( \tilde{z} = [0 \ 0 \ 1]^T \) is used to normalize \( m_{pij} \) by its third component.

The nonlinear least squares optimization uses the Jacobian of the error vector \( E \) with respect to the optimization parameters \( x \), which contains the corrections to the camera pose \((R, T)\):

\[
x = [x_r, x_t]
\]

\[
[R, \hat{T}] = \left[ e^{(x_r)}, x_t \right] \oplus [R, T]
\]
line segments correspond to the same image line and the matching error for each segment can be computed just as in Equation 6, however the errors are reweighted such that the total contribution is invariant to the number of segments the line is broken up into:

$$E_{ij} = \frac{1}{\sqrt{N_i}} \left( \frac{1}{2} m_{pij}^T \frac{1}{2} m_{pij} \right)$$

(10)

where \(N_i\) is the number of segments in line \(i\), and \(j \in \{1, \cdots, N_i + 1\}\) indexes the points from the 3D line.

### III. Experimental Results

#### A. Experimental Setups

In order to demonstrate generality and robustness of the proposed extrinsic calibration algorithm, we conduct a series of experiments using variety of sensor configurations. We present results from different range sensors such as a 3D laser scanner (a Velodyne HDL-64E), a Microsoft Kinect RGB-D sensor, and a spring-mounted Hokuyo UTM-30LX [12]. For imaging sensors, we test the algorithm using both a monochrome and color camera.

First, we assess our proposed method based on the datasets provided by Geiger et al. [13]. The datasets contain several scan-image pairs in both indoor and outdoor settings. The range-camera experimental configuration setups [13] are illustrated in Figure 2. The first setup contains dense 3D data generated by a Kinect range sensor and high resolution monochrome cameras with resolution of 1280×1024 (denoted as configuration 1, 2, 3). The second setup consists of a sparse 3D point cloud acquired by the Velodyne HDL-64E with a trinocular camera (both color and monochrome) with resolution of 1392×512 (denoted as configuration 4, 5, 6).

For the second dataset, a handheld 3D mobile mapping system called Zebedee is used [12]. Zebedee consists of a Hokuyo UTM-30LX lidar scanner and a Microstrain 3DM-GX3 IMU mounted on a spring (Figure 3). The operating concept behind Zebedee is that the natural motion of the operator is converted to a non-deterministic (predominantly pitching rotational) motion about the spring thereby extending the lidar scanner’s field of view from 2D to 3D. The Hokuyo scanner operates at a scanning rate of 40 Hz generating 1080 points per scan at 0.25° angular resolution within a 270° 2D field of view and up to a maximum range of 30 m. The non-deterministic scanner motion typically produces a field of view perpendicular to the scan plane of approximately 150–170° at a frequency of around 1 Hz. A specialized SLAM algorithm accurately estimates the motion of the scanner given the lidar and IMU measurements (the spring also ensures that surfaces in the environment are re-observed at an appropriate frequency), producing a six degree of freedom trajectory estimate and a corresponding 3D point cloud [12]. The version of Zebedee hardware used in these experiments has been enhanced by the addition of a Point Grey Firefly MV global shutter CMOS camera with resolution of 640×480 color images at 30 fps and 40° horizontal field of view. The camera is rigidly mounted to the sensor head, and therefore its velocity can be quite significant. To mitigate the effects of motion blur, we configure the camera to have a fast shutter speed, though we are also investigating methods of correcting the motion blur (e.g., the algorithm by Joshi et al. [14]).

#### B. Evaluation

In the first experiment, we evaluate the accuracy of the proposed method with respect to the number of corresponding sets of lines. We start by randomly selecting \(n \in \{3, P - 1\}\) line pairs, where \(P\) is the total number of line correspondences extracted from a range-image set. These \(n\) 2D-3D line pairs are then used to estimate the extrinsic parameters. Next, we employ the estimated extrinsic parameters to project the \(m\) remaining 3D lines \(m = P - n\) onto the image plane and compute the reprojection error values in pixels. The reprojection error is calculated based on the distance in pixels between each 2D line and its corresponding matched projected 3D line onto the image plane. We repeat this procedure 10 times for each \(n\) and calculate the root mean square of reprojection errors. The results using the two Kinect-monochrome configurations (configurations 1 and 3) [13] are represented as box-plots in Figure 4. As it described in Section II-C a minimum of three lines are required to constrain all six DoF’s. However, if the three lines are not distinct (i.e., they intersect at least on one common point), the optimization will be weakly constrained. As can be seen from the results, the reprojection error...
It should also be noted that although the datasets provided by Geiger et al. [13] contain planar checkerboards, we are not extracting any information regarding the checkerboard patterns.

Next, in order to evaluate the effectiveness and robustness of the proposed approach over sparse 3D point clouds, we consider the Velodyne-trinocular camera setting datasets sourced from Geiger et al. [13]. Detecting straight 3D lines from sparse range data requires interpolation and can lead to noisy measurements. However, we can demonstrate that the proposed method is robust to noisy occlusion boundaries in sparse datasets by extracting multi-segment 3D lines instead of estimating single straight boundary lines. We first extract a convex polygon of each 3D plane and then fit a line segment to each consecutive vertex on the convex hulls. Since all the 3D line segments correspond to a single 2D image line, the total contribution from them in the minimization problem are reweighted using Equation 10 such that it is invariant to the number of line segments.

Figure 6 demonstrates an example of 2D and 3D lines segments extracted from a color camera and the Velodyne HDL-64E along with a reprojection of the 3D lines onto the image.
Fig. 7: Photorealistic 3D map of indoor office environment generated using Zebedee. The trajectory is colored by time: blue at the start transitioning to red at the end. Zoomed regions showing the detail level of photorealistic 3D maps. It took around five minutes to map the entire floor (20 m).

plane and the photorealistic 3D model of the same scene. In this experiment, a total number of eight image lines are extracted corresponding to fourteen 3D line segments (three distinct plane intersection lines and eleven boundary lines segments). The root mean square of reprojection errors using only plane intersection lines is 6.71 pixels while adding noisy multi-segment 3D lines reduces the reprojection error to 1.85 pixels.

We also numerically compare our results with the Geiger et al. [13] approach. The Geiger et al. [13] method requires mounting multiple checkerboard patterns on planar surfaces in the scene (e.g., Figure 6) to be able to extrinsically calibrate the camera with respect to the range sensor.

Table I presents the total root mean square reprojection errors in number of pixels for six different datasets using the extrinsic parameters estimated using the method proposed in this paper and the best estimated extrinsic parameters using the Geiger et al. [13] method. Based on these results, the proposed method outperforms the Geiger et al. [13] approach for all of the configurations. While the Geiger et al. [13] method belongs to the group of techniques require external artificial targets to be observed simultaneously from the range and imaging sensors, our proposed method does not rely on locating checkerboard corners or patterns, thus making it suitable for the calibration of sensors in-situ or applying it to any previously recorded range-image datasets without any specific calibration stage.

For the final experiment, we test the proposed method on the Zebedee handheld 3D mobile mapping system. Since Zebedee requires motion to extend the lidar scanner’s field of view from 2D to 3D, the timing latency between the range sensor and camera measurements is critical and needs to be addressed properly. Thus, we add the unknown latency between the laser and camera into the same optimization problem to jointly estimate a 6DoF transformation and latency that minimizes the reprojection errors. Extended analysis of validation of this optimization is beyond the scope of this paper and will be described in our future work.

Figure 7 shows an example of photorealistic 3D maps generated by Zebedee in an indoor office environment after registration between range and camera sensors with sensor trajectory overlaid. The point cloud is down-sampled for
visualization. However, the zoomed regions show more detailed views of the dense 3D colorized point clouds. Most points from the ceiling have been removed for clarity.

The entire extrinsic calibration procedure is implemented as a Matlab toolbox and it will be released to the community. In the current implementation, the user needs to manually select the corresponding 2D lines with respect to the extracted 3D lines. However, in future work we would like to implement an automatic matching technique based on a probabilistic RANSAC method.

IV. CONCLUSIONS

We have presented a novel extrinsic calibration algorithm which uses natural linear features in the scene to determine the rigid transformation between range and image sensors’ coordinate systems. The approach does not require any modification into the natural scene. First, a set of 2D and 3D line segments are extracted from image and range data respectively. Given 2D-3D line correspondences, the proposed method uses a nonlinear least squares optimization to find a 6DoF transformation that minimizes the reprojection errors of the 2D-3D line pairs. The minimum number of lines required for optimization is three distinct line pairs. However, the more lines that are utilized, the more observable the estimation of 6DoF transformation becomes. Furthermore, we have extended the algorithm to handle sparse lidar data by extracting a convex polygon of each 3D plane and fit line segments to each consecutive vertex on the convex hulls. Later, the optimization problem is modified to incorporate the one-to-many correspondences situation by re-weighting the error metric by the number of line segments. The capability of robustly detecting natural linear features and not requiring any calibration targets in the scene enable us to apply the proposed method to many sensor configurations (sparse and dense range data) as well as any previously recorded range-image dataset without any calibration sequence. Currently, we are investigating more robust algorithms for image line extraction. For future work, we would like to remove the requirement of human intervention by incorporating automatic correspondence of 2D-to-3D lines using iterative matching techniques.

ACKNOWLEDGMENTS

The authors thank Paul Flick from the CSIRO Autonomous Systems Laboratory for his technical support. We also thank the Karlsruhe Institute of Technology for providing the Kinect and Velodyne datasets.

REFERENCES