Visually Bootstrapped Generalized ICP

Gaurav Pandey
Department of Electrical Engineering & Computer Science
University of Michigan, Ann Arbor, MI 48109
Email: pgaurav@umich.edu

James R. McBride
Research and Innovation Center
Ford Motor Company, Dearborn, MI 48124
Email: jmcbride@ford.com

Silvio Savarese
Department of Electrical Engineering & Computer Science
University of Michigan, Ann Arbor, MI 48109
Email: silvio@umich.edu

Ryan M. Eustice
Department of Naval Architecture & Marine Engineering
University of Michigan, Ann Arbor, MI 48109
Email: eustice@umich.edu

***Abstract*— This paper reports a novel algorithm for bootstrapping the automatic registration of unstructured 3D point clouds collected using co-registered 3D lidar and omnidirectional camera imagery. Here, we exploit the co-registration of the 3D point cloud with the available camera imagery to associate high dimensional feature descriptors such as scale invariant feature transform (SIFT) or speeded up robust features (SURF) to the 3D points. We first establish putative point correspondence in the high dimensional feature space and then use these correspondences in a random sample consensus (RANSAC) framework to obtain an initial rigid body transformation that aligns the two scans. This initial transformation is then refined in a generalized iterative closest point (ICP) framework. The proposed method is completely data driven and does not require any initial guess on the transformation. We present results from a real world dataset collected by a vehicle equipped with a 3D laser scanner and an omnidirectional camera.**

I. Introduction

One of the basic tasks of mobile robotics is to automatically create 3D maps of the unknown environment. To create realistic 3D maps, we need to acquire visual information from the environment, such as color and texture, and to precisely map it onto range information. To accomplish this task, the camera and 3D laser range finder need to be extrinsically calibrated [1] (i.e., the rigid body transformation between the two reference systems is known). The extrinsic calibration allows us to associate texture to a single scan, but if we want to create a full 3D model of the entire environment, we need to automatically align hundreds or thousands of multiple scans using scan matching techniques.

The most common method of scan matching is popularly known as iterative closest point (ICP) and was first introduced by Besl and McKay [2]. In their work, they proposed a method to minimize the Euclidean distance between corresponding points to obtain the relative transformation between the two scans. Chen and Medioni [3] further introduced the point-to-plane variant of ICP owing to the fact that most of the range measurements are typically sampled from a locally planar surface. Similarly, Alshawa [4] introduced a line-based matching variant called iterative closest line (ICL). In ICL line features are extracted from the range scans and aligned to obtain the rigid body transformation. Several other variants of the ICP algorithm have also been proposed and can be found in the survey paper by Rusinkiewicz and Levoy [5].

One of the main reasons for the popularity of ICP-based methods is that it solely depends on the 3D points and does not require extraction of complex geometric primitives. Moreover, the speed of the algorithm is greatly boosted when it is implemented with kd-trees [6] for establishing point correspondences. However, most of the deterministic algorithms discussed so far do not account for the fact that in real world datasets, when the scans are coming from two different time instances, we never achieve exact point correspondence. Moreover, scans are generally only partially overlapped—making it hard to establish point correspondences by applying a threshold on the point-to-point distance.

Recently, several probabilistic techniques have been proposed that model the real world data better than the deterministic methods. Biber et al [7] applies a probabilistic model by assuming that the second scan is generated from the first through a random process. Haehnel and Burgard [8] apply ray tracing techniques to maximize the probability of alignment. Biber [9] also introduced an alternate representation of the range scans, the normal distribution transform (NDT), where they subdivide a 2D plane into cells and assign a normal distribution to each cell to model the distribution of points in that cell. They use this density to match the scans and therefore no explicit point correspondence is required. Segal et al [10] proposed to combine the iterative closest point and point-to-plane ICP algorithms into a single probabilistic framework. They devised a generalized framework that naturally converges to point-to-point or point-to-plane ICP by appropriately defining the sample covariance matrices associated with each point. Their method exploits the locally planar structure of both participating scans as opposed to just a single scan as in the case of point-to-plane ICP. They have shown promising results with full 3D scans acquired from a Velodyne laser scanner.

Most of the ICP algorithms described above are based on 3D point clouds alone and very few incorporate visual

information into the ICP framework. Johnson and Kang [11] proposed a simple approach incorporating color information in the ICP framework by augmenting the three color channels to the 3D coordinates of the point cloud. Although this technique adds color information to the ICP framework, it is highly prone to registration errors. Moreover, the three RGB channels are not the best representation of visual information of the scene. Recently, Akca et al [12] proposed a novel method of using intensity information for scan matching. They proposed the concept of a quasisurface, which is generated by scaling the normal at a given 3D point by its color, and then matching the geometrical surface and the quasisurfaces in a combined estimation model. This approach works well when the environment is structured and the normals are well defined.

All of the aforementioned methods use the color information directly, i.e., they are using the very basic building blocks of the image data (RGB values), which does not provide strong distinction between the points of interest. However, there has been significant development over the last decade in the feature point detection and description algorithms employed by the computer vision and image processing community. We can now characterize any point in the image by high dimensional descriptors such as the scale invariant feature transform (SIFT) [13] or speeded up robust features (SURF) [14], as compared to just RGB values alone. These high dimensional features provide a better measure of correspondence between points as compared to the Euclidean distance. The extrinsic calibration of 3D lidar and omnidirectional camera imagery allows us to associate these robust high dimensional feature descriptors to the 3D points.

Once we have augmented the 3D point cloud with these high dimensional feature descriptors we can then use them to align the scans in a robust manner. We first establish point correspondence in the high dimensional feature space using the image-derived feature vectors and then use these putative correspondences in a random sample consensus (RANSAC) [15] framework to obtain an initial rigid body transformation that aligns the two scans. This initial transformation is then refined in a generalized ICP framework as proposed by Segal et al [10].

The outline of the rest of the paper is as follows: In section II we describe the proposed method of automatic registration of the 3D scans. We divide the method into two parts, a RANSAC framework to obtain the initial transformation from SIFT correspondences and a refinement of this initial transformation via a generalized ICP framework. In section III we present results showing the robustness of the proposed method and present a comparison of our method with the unenhanced generalized ICP algorithm. Finally, in section IV we summarize our findings.

II. Methodology

In our previous work [1] we presented an algorithm for the extrinsic calibration of a 3D laser scanner and an omnidirectional camera system. The extrinsic calibration of the two

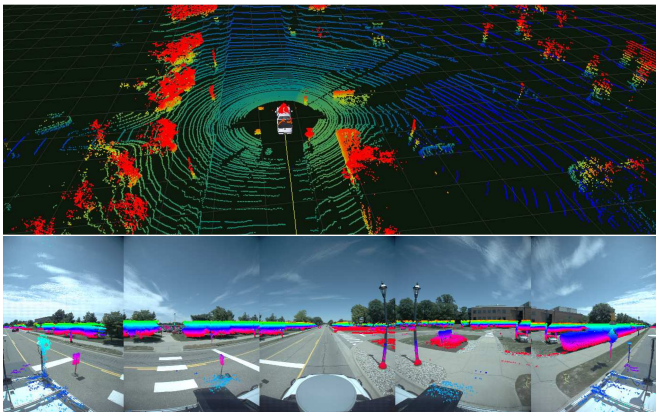


Fig. 1. The top panel is a perspective view of the Velodyne 3D lidar range data, color-coded by height above the ground plane. The bottom panel shows the above ground plane range data projected into the corresponding image from the Ladybug3 camera.

sensors allows us to project 3D points onto the corresponding omnidirectional image (and vice versa) as depicted in Fig. 1. This co-registration allows us to calculate high dimensional feature descriptors in the omnidirectional image (in this paper we use SIFT) and associate them to a corresponding 3D lidar point that projects onto that pixel location. Since only few 3D points are projected onto interesting parts of the image (i.e., where visual feature points are detected), only a subset of the 3D points will have a feature descriptor assigned to them. To be consistent throughout the text we have adopted the notation below for describing the different attributes of a co-registered camera-lidar scan, here referred to as Scan A.

1) X_A: $\{\mathbf{x}_a^i \in \mathbb{R}^3, i = 1, ...n\}$ set of 3D lidar points.
2) U_A: $\{\mathbf{u}_a^i \in \mathbb{R}^2, i = 1, ...n\}$ set of reprojected pixel coordinates associated with 3D lidar points.
3) S_A: $\{\mathbf{s}_a^i \in \mathbb{R}^{128}, i = 1, ...m\}$ set of extracted SIFT descriptors.
4) Y_A: $\{\mathbf{y}_a^i \in \mathbb{R}^3, i = 1, ...m\}$ subset of 3D lidar points that are assigned a SIFT descriptor, $\mathrm{Y_A} \subset \mathrm{X_A}$.
5) V_A: $\{\mathbf{v}_a^i \in \mathbb{R}^2, i = 1, ...m\}$ subset of reprojected pixel coordinates that have a SIFT descriptor, $\mathrm{V}_A \subset \mathrm{U}_A$.

Once we have augmented the 3D point cloud with the high dimensional feature descriptors, we then use them to align the scans in a two step process. In the first step, we establish putative point correspondence in the high dimensional feature space and then use these correspondences within a RANSAC framework to obtain a coarse initial alignment of the two scans. In the second step, we refine this coarse alignment using a generalized ICP framework [10]. Fig. 2 depicts an overview block-diagram of our algorithm.

The novel aspect of our work is in how we derive this initial coarse alignment. Our algorithm is completely data driven and does not require the use of external information (e.g., odometry). The initial alignment is intrinsically derived from the data alone using visual feature/lidar primitives available in the co-registered sensing modality. Note that initialization is typically the weakest link in any ICP-based methodology. By adopting our RANSAC framework, we are

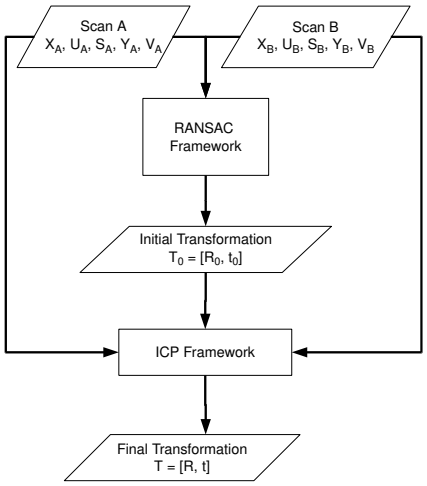


Fig. 2. Block-diagram depicting the two step scan alignment process.

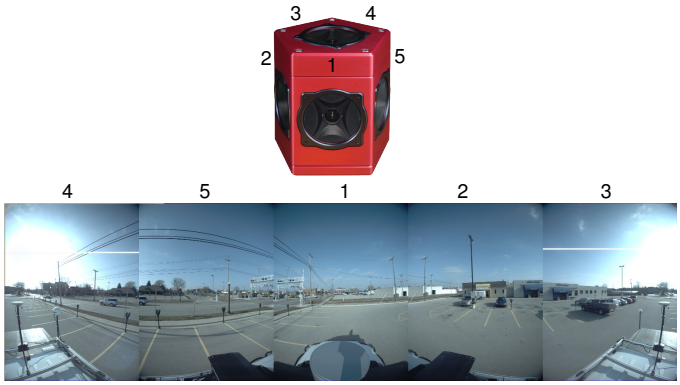


Fig. 3. A depiction of the Ladybug3 omnidirectional camera system and a sample image showing the field of view of cameras 1 through 5.

able to extend the convergence of generalized ICP over three times beyond the inter-scan distance that it normally breaks down. In the following, we explain our two-step algorithm in detail and discuss our novel concept of a camera consensus matrix (CCM).

A. RANSAC Framework

In the first part of our algorithm, we estimate a rigid body transformation that approximately aligns the two scans using putative visual correspondences. We do so by matching the SIFT feature sets, S_A and S_B, across the two scans and make the assumption that the matched 3D feature points, Y_A and Y_B, correspond to the same 3D point in Euclidean space. If we have three correct point correspondences, then we can calculate the rigid body transformation that aligns the two scans using the method proposed by Arun et al [16]. However, if there exist outliers in the correspondences obtained by matching SIFT features, then this transformation will be wrong. Hence, we adopt a RANSAC framework [15] whereby we randomly sample three point correspondence pairs and iteratively compute the rigid body transformation until we find enough consensus or exceed a preset maximum number of iterations based upon a probability of outliers.

The difficult aspect of this task is establishing a good set of putative correspondences so as to get a sufficient number of inliers. In our work we used the Point Grey Ladybug3 for our omnidirectional camera system [17]. The Ladybug3 has six 2-Megapixel (1600×1200) cameras, five positioned in a horizontal ring and one positioned vertically. Each sensor of the omnidirectional camera system has a minimally overlapping field of view (FOV) as depicted in Fig. 3. The usable portion of the omnidirectional camera system essentially consists of five cameras spanning the 360° horizontal FOV. Unless we use prior knowledge on the vehicle's motion, we do not know *a priori* which camera pairs will overlap between the first and second scans. Hence, a simple global correspondence search over the entire omnidirectional image set will not give robust feature correspondence. Instead, in order to improve our putative feature matching, we exploit a novel camera consensus matrix that intrinsically captures the geometry of the omnidirectional camera system in order to establish a *geometrically consistent* set of putative point correspondences in SIFT space.

1) Camera Consensus Matrix: If the motion of the camera is known, then robustness to incorrect matches can be achieved by restricting the correspondence search to localized regions. Since we do not assume that we know the vehicle motion *a priori*, we first need to estimate these localized regions based upon visual similarity. To do so, we divide the FOV of the omnidirectional camera into n equally spaced regions. In our case we chose $n = 5$ because the five sensors of the omnidirectional camera naturally divide the FOV into five equispaced regions.[1] Once the FOV is partitioned we need to identify the cameras that have the maximum overlap between the two instances when the scans are captured. In our work, we assume that the motion of the vehicle is locally planar (albeit unknown).

For a small forward translational motion of the vehicle (Fig. 4) the maximum FOV overlap between scans A and B occurs for the following pairs of cameras: {1-1, 2-2, 3-3, 4-4, 5-5}. Similarly, for large forward translational motion the maximum overlap of camera 1 of scan A can be with either of {1, 2 or 5} of scan B (i.e., the forward looking cameras) (Fig. 4), whereas for the remaining four cameras of scan A the maximum overlap is obtained between {2-3, 3-3, 4-4, 5-4} of scan B. This overlap of the cameras is captured in a matrix called the camera consensus matrix (CCM). The CCM is a [5 × 5] binary matrix where each element $\mathrm{C}(i, j)$ defines the correspondence consensus of the i^{th} camera of scan A with the j^{th} camera of scan B, where 0 means no consensus and 1 means maximum consensus between the regions.

Similar to our translational motion example, we can also obtain the CCM for pure rotation of the vehicle about the yaw axis by circularly shifting the columns of the identity matrix as depicted in Fig. 5. Moreover, we can calculate the CCM matrices resulting from the combined rotational

[1]Note that in the case of catadioptric omnidirectional camera systems, the entire panoramic image can be divided into smaller equispaced regions.

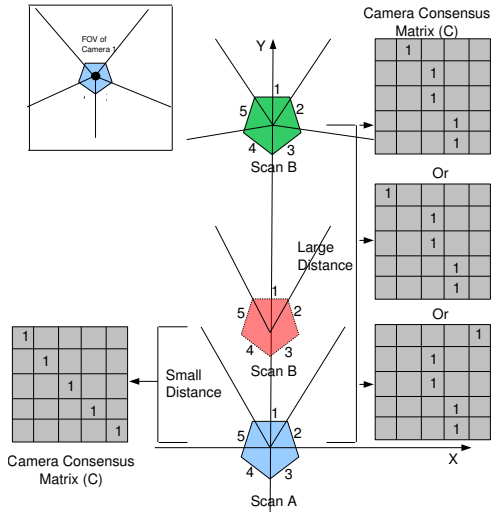


Fig. 4. Top view of the omnidirectional camera system depicting the intersecting FOV of individual camera sensors as the omnidirectional camera-rig moves forward along the Y axis. For small translational motion (blue to red), the FOV of the cameras between scan A and scan B does not change much, thereby giving maximal overlap with the same sensors and is described by the identity CCM matrix shown on the left. For large forward translational motion (blue to green), the FOV of the individual camera sensors does change and what was visible in camera 1 of scan A can now be visible in either of the forward looking cameras $\{1, 2 \text{ or } 5\}$ of scan B, resulting in the sample CCM matrices shown on the right.

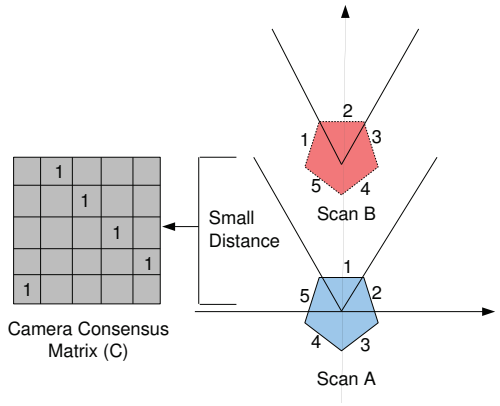


Fig. 5. Top view of the omnidirectional camera system depicting the intersecting FOV of individual camera sensors as the camera-rig rotates about the yaw axis. Here we have shown one possible discrete rotation such that the FOV of each sensor is circularly shifted by one unit, resulting in the sample CCM shown on the left. In this case, five such discrete rotations are possible.

and translational motion of the vehicle by circularly shifting the CCM matrices from Fig. 4. Each resulting binary CCM represents a consistent geometry hypothesis of the camera motion and can be considered as a set of basis matrices spanning the entire space of possible CCMs arising due to the discrete planar vehicle motion assumed here. We vectorize these basis matrices by stacking the rows together into a vector, denoted $\mathbf{h}_i$, where each $\mathbf{h}_i$ corresponds to a valid geometry configuration CCM hypothesis.

2) Camera Constrained Correspondence Search: To use the concept of the CCM to guide our image feature matching, we first need to empirically compute the measured CCM arising from the visual similarity of the regions of scan A and scan B using the available image data. Each element of the empirically derived CCM is computed as the sum of the inverse SIFT score (i.e., squared Euclidean distance) of the matches established between camera i of scan A and camera j of scan B. This yields a measure of visual similarity between the two regions:

$$\tilde{\mathrm{C}}(i,j) = \sum_{k} 1/s_k, \tag{1}$$

where s_k is the SIFT matching score of the k^{th} match. This matrix is then normalized across the columns so that values are within the interval $[0, 1]$ to comply with our notion that 0 means no consensus and 1 means maximum consensus:

$$\hat{\mathrm{C}}(i,j) = \tilde{\mathrm{C}}(i,j) / \max(\tilde{\mathrm{C}}(i)). \tag{2}$$

Here $\max(\tilde{\mathrm{C}}(i))$ denotes the maximum value in the i^{th} row of the matrix $\tilde{\mathrm{C}}$.

This matrix $\hat{\mathrm{C}}$ is then vectorized to obtain the corresponding camera consensus vector $\hat{\mathbf{c}}$. To determine which ideal CCM hypothesis is most likely, we project this vector to all the hypothesis basis vectors $\mathbf{h}_i$ and calculate the orthogonal error of projection:

$$e_i = \|\hat{\mathbf{c}} - \mathbf{h}_i \frac{\hat{\mathbf{c}} \cdot \mathbf{h}_i}{\|\hat{\mathbf{c}}\| \|\mathbf{h}_i\|}\| \tag{3}$$

The basis vector $\mathbf{h}_i$ that has the least orthogonal error of projection yields the closest hypothesis on the CCM. This geometrically consistent camera configuration is then used for calculating the camera constrained SIFT features. Fig. 6 depicts a typical situation where the CCM yields a more robust feature correspondence as compared to the simple global correspondence search alone. The CCM-consistent putative correspondences are then used in the RANSAC framework to estimate the rigid body transformation that aligns the two scans. The complete RANSAC algorithm to estimate the rigid body transformation is outlined in Algorithm 1.

B. ICP Framework

Our method to refine the initial transformation obtained from section II-A is based upon the generalized ICP (GICP) algorithm proposed by Segal et al [10]. The GICP algorithm is derived by attaching a probabilistic model to the cost function minimization step of the standard ICP algorithm outlined in Algorithm 2. In this section we review the GICP algorithm as originally described in [10].

The cost function at line 13 of the standard ICP algorithm is modified in [10] to give the generalized ICP algorithm. In GICP the point correspondences are established by considering the Euclidean distance between the two point clouds X_A and X_B. Once the point correspondences are established, the ICP cost function is formulated as a maximum likelihood estimate (MLE) of the transformation "T" that best aligns the two scans.

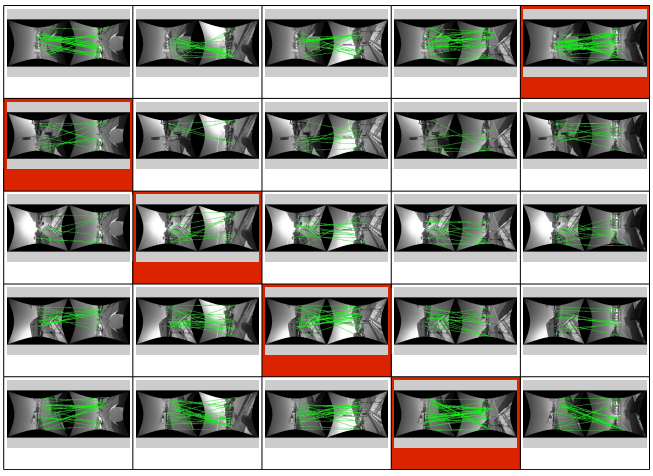
In the GICP framework the points in the two scans are assumed to be coming from Gaussian distributions, $\mathbf{x}_a^i \sim$

Algorithm 1 RANSAC Framework

input: Y_A, Y_B, S_A, S_B,
output: The estimated transformation $[R_0, t_0]$
Establish camera constrained SIFT correspondences between S_A and S_B.
Store the matches in a list L.
while iter $<$ MAXITER **do**
  Randomly pick 3 pairs of points from the list L.
  Retrieve these 3 pair of points from Y_A and Y_B.
  Calculate the 6-DOF rigid body transformation $[R, t]$ that best aligns these 3 points.
  Store this transformation in an array M, $M[iter] = [R, t]$
  Apply the transformation to Y_B to map Scan B's points into the reference frame of Scan A: $\mathrm{Y}'_B = \mathrm{R}\mathrm{Y}_B + t$
  Calculate the set cardinality of pose-consistent SIFT correspondences that agree with the current transformation (i.e., those that satisfy a Euclidean threshold on spatial proximity): $n = |(\mathrm{Y}'_B(L) - \mathrm{Y}_A(L)) < \epsilon|$
  Store the number of pose-consistent correspondences in an array N, $N[iter] = n$
  $iter = iter + 1$
end while
Find the index i that has maximum number of correspondences in N.
Retrieve the transformation corresponding to index i from M. $[R_0, T_0]$ = $M[i]$. This is the required transformation.

Algorithm 2 Standard ICP Algorithm [10]

input: Two point clouds: X_A, X_B;
An initial transformation: T_0
output: The correct transformation, T, which aligns X_A and X_B
$\mathrm{T} \leftarrow \mathrm{T}_0$
while not converged **do**
  for $i \leftarrow 1$ to N **do**
    $m_i \leftarrow FindClosestPointInB(\mathrm{T} \cdot \mathbf{x}_a^i)$
    if $\|m_i - \mathrm{T} \cdot \mathbf{x}_a^i\| <= d_{max}$ **then**
      $w_i \leftarrow 1$;
    else
      $w_i \leftarrow 0$;
    end if
  end for
  $\mathrm{T} \leftarrow \mathrm{argmin}_\mathrm{T} \sum_i w_i \|\mathrm{T} \cdot \mathbf{x}_a^i - m_i\|^2$
end while



0.7386	0.3669	0.2658	0.5588	1
1	0.6307	0.3590	0.4303	0.4800
0.6266	1	0.5160	0.7388	0.7482
0.4475	0.5219	1	0.5846	0.2976
0.6395	0.3719	0.2865	0.6600	1

Camera Consensus Matrix (CCM) based on visual similarity

0	0	0	0	1
1	0	0	0	0
0	1	0	0	0
0	0	1	0	0
0	0	0	1	0

Closest hypothesis corresponding to the CCM

Fig. 6. This figure shows the pairwise exhaustive SIFT matches obtained across the five cameras of scan A and scan B. The corresponding empirically measured CCM is shown below on the left, and the closest matching binary CCM hypothesis is shown below on the right. The blocks highlighted in red indicate the CCM-consistent maximal overlap regions. In this case, the resulting CCM hypothesis indicates a clockwise rotational motion by one camera to the right (refer to Fig. 5).

$\mathcal{N}(\tilde{\mathbf{x}}_a^i; \mathrm{C}_i^A)$ and $\mathbf{x}_b^i \sim \mathcal{N}(\tilde{\mathbf{x}}_b^i; \mathrm{C}_i^B)$, where $\tilde{\mathbf{x}}_a^i$ and $\tilde{\mathbf{x}}_b^i$ are the mean or actual points and C_i^A and C_i^B are sample based covariance matrices associated with the measured points. Now in the case of perfect correspondences (i.e., geometrically consistent with no errors due to occlusion or sampling) and correct transformation, T^*:

$$\tilde{\mathbf{x}}_b^i = \mathrm{T}^* \tilde{\mathbf{x}}_a^i. \tag{4}$$

But for an arbitrary transformation T, and noisy measurements $\mathbf{x}_a^i$ and $\mathbf{x}_b^i$, the alignment error can be defined as $\mathbf{d}_i = \mathbf{x}_b^i - \mathrm{T}\mathbf{x}_a^i$. Now the ideal distribution from which $\mathbf{d}_i^{(\mathrm{T}^*)}$ is drawn is given as:

$$\begin{aligned} \mathbf{d}_i^{(\mathrm{T}^*)} &\sim \mathcal{N}(\tilde{\mathbf{x}}_b^i - \mathrm{T}^* \tilde{\mathbf{x}}_a^i, \mathrm{C}_i^B + \mathrm{T}^* \mathrm{C}_i^A \mathrm{T}^{*\top}) \\ &= \mathcal{N}(\mathbf{0}, \mathrm{C}_i^B + \mathrm{T}^* \mathrm{C}_i^A \mathrm{T}^{*\top}). \end{aligned}$$

Here $\mathbf{x}_a^i$ and $\mathbf{x}_b^i$ are assumed to be drawn from independent Gaussians. Thus the required transformation T is the MLE computed by setting:

$$\mathrm{T} = \underset{\mathrm{T}}{\mathrm{argmax}} \prod_i p(\mathbf{d}_i^{(\mathrm{T}^*)}) = \underset{\mathrm{T}}{\mathrm{argmax}} \sum_i \log p(\mathbf{d}_i^{(\mathrm{T}^*)}) \tag{5}$$

which can be simplified to:

$$\mathrm{T} = \underset{\mathrm{T}}{\mathrm{argmin}} \sum_i \mathbf{d}_i^{\mathrm{T}} (\mathrm{C}_i^B + \mathrm{T}\mathrm{C}_i^A \mathrm{T}^T)^{-1} \mathbf{d}_i. \tag{6}$$

The rigid body transformation T given in (6) is the MLE refined transformation that best aligns scan A and scan B.

III. Results

We present results from real data collected from a 3D laser scanner (Velodyne HDL-64E) and an omnidirectional camera system (Point Grey Ladybug3) mounted on the roof of a Ford F-250 vehicle (Fig. 7). We use the pose information available from a high end inertial measurement unit (IMU) (Applanix POS-LV) as the ground truth to compare the scan alignment errors. We performed the following experiments to analyze the robustness of the bootstrapped generalized ICP algorithm.

Fig. 7. Test vehicle equipped with a 3D laser scanner and omnidirectional camera system.

A. Experiment 1

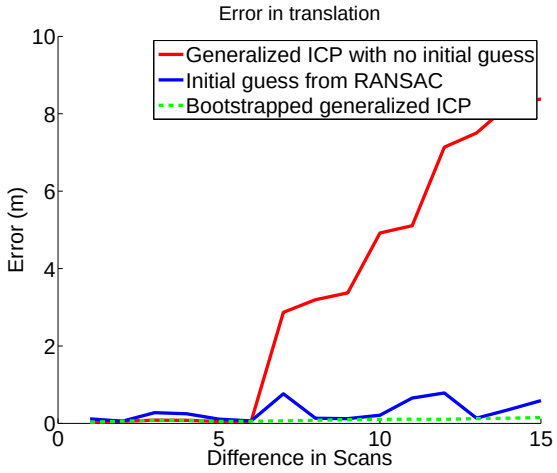
In the first experiment we selected a series of 15 consecutive scans captured by the laser-camera system in an outdoor urban environment collected while driving around downtown Dearborn, Michigan at a vehicle speed of approximately $15.6\ \mathrm{m/s}$ ($35\ \mathrm{mph}$). The average distance between the consecutive scans is approximately $0.5\ \mathrm{m}$ - $1.0\ \mathrm{m}$. In this experiment we fixed the first scan to be the reference scan and then tried to align the remaining scans (2–15) with the base scan using *(i)* the generalized ICP alone, *(ii)* our RANSAC initialization alone, and *(iii)* the bootstrapped generalized ICP algorithm seeded by our RANSAC solution. The error in translational motion between the base scan and the remaining scans obtained from these algorithms is plotted in Fig. 8. We found the plotted error trend to be typical across all of our experiments—in general the GICP algorithm alone would fail after approximately 5 or so scans of displacement when not fed an initial guess. However, by using our RANSAC framework to bootstrap seed the GICP algorithm, we were able to significantly extend GICP's convergence out past 15 scans of displacement.

We repeated this experiment for 10 sets of 15-scan pairs (i.e., 150 scans in total) from different locations in Dearborn and calculated the average translational and rotational error as a function of the intra-scan displacement. The resulting error statistics are tabulated in Table I where we see that the bootstrapped GICP is able to provide sub $25\ \mathrm{cm}$ translational error at 15 scans apart, while GICP alone begins to fail after only 5 scans of displacement.

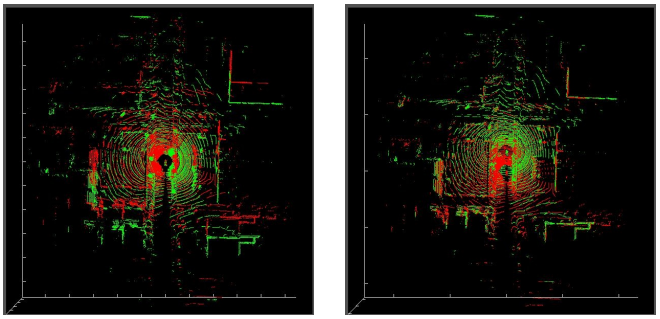
B. Experiment 2

In the second experiment, we compared the output of GICP and our bootstrapped GICP in a real-world application-driven context. For this experiment we drove a $1.6\ \mathrm{km}$ loop around downtown Dearborn, Michigan with the intent of characterizing each algorithm's ability to serve as a registration engine for localizing and 3D map building in an outdoor urban environment. For this purpose we used a pose-graph simultaneous localization and mapping (SLAM) framework where the ICP-derived pose constraints served as edges in the graph. We employed the open-source incremental smoothing and mapping (iSAM) algorithm by Kaess [18] for inference. In our experiment the pose-constraints are obtained only from the scan matching algorithm and no odometry information is used in the graph.

Fig. 9 shows the vehicle trajectory given by the iSAM algorithm (green) overlaid on top of OmniStar HP global positioning system (GPS) data ($\sim$2 cm error) for ground-truth (red). Here the pose constraints were obtained by aligning every third scan using GICP with no initial guess from odometry. As we can see in Fig. 9(b), the resulting iSAM output differs greatly from the ground truth. This mainly occurs because the generalized ICP algorithm does not converge to the global minimum when it is initialized with a poor guess, which means the pose-constraints that we



(a) Error comparison between GICP and bootstrapped GICP.



(b) GICP for Scan 10 (c) BGICP for Scan 10

Fig. 8. Graph showing the error (a) in translation as the distance between scans A and B is increased. Top view of the 3D scans aligned with the output of GICP (b) and bootstrapped GICP (c) for two scans that are 10 time steps apart. Note that the GICP algorithm fails to align the two scans when unaided by our novel RANSAC initialization step.

TABLE I

THIS TABLE SUMMARIZES THE ERROR IN SCAN ALIGNMENT. WE SHOW HERE THE TRANSLATION AND ROTATIONAL ERROR BETWEEN SCAN PAIRS {1-2, 1-5, 1-10, 1-15} OBTAINED AT DIFFERENT LOCATIONS. HERE WE HAVE USED THE POSE OF THE VEHICLE OBTAINED FROM A HIGH END IMU AS GROUND TRUTH TO CALCULATE ALL THE ERRORS.

	Generalized ICP with no initial guess						Initial guess from RANSAC						Bootstrapped generalized ICP					
Scans	T (m)		Ax (degrees)		An (degrees)		T (m)		Ax (degrees)		An (degrees)		T (m)		Ax (degrees)		An (degrees)	
	Err	Std	Err	Std	Err	Std	Err	Std	Err	Std	Err	Std	Err	Std	Err	Std	Err	Std
1-2	.047	.011	0	0	.05	.02	.15	.02	0	0	.223	.0003	.04	.010	0	0	.057	.110
1-5	.546	.173	.570	.20	1.15	.344	.20	.03	.43	.15	.230	.0001	.084	.010	.025	.090	.058	.006
1-10	6.37	.868	.710	.25	1.72	.573	.51	.09	.59	.01	.745	.0044	.145	.015	.030	.010	.057	.012
1-15	10.34	.834	1.86	.13	2.86	.057	1.02	.02	1.35	.54	1.15	.0021	.220	.008	.042	.015	.070	.017

T = Error in translation (meters); Ax = Error in rotation axis (degrees); An = Error in rotation angle (degrees)
Err = Average Error; Std = Standard Deviation

get are biased, and hence a poor input to iSAM. Fig. 9(d) shows the resulting vehicle trajectory for our bootstrapped GICP algorithm when given as input to the iSAM algorithm, which agree well with the GPS ground-truth.

IV. CONCLUSION

This paper reported an algorithm for robustly determining a rigid body transformation that can be used to seed a generalized ICP framework. We have shown that in the absence of a good initial guess, the pose information obtained from the generalized ICP algorithm is not optimal if the scan alignment is performed using the 3D point clouds alone. We have also shown that if we incorporate visual information from co-registered omnidirectional camera imagery, we can provide a good initial guess on the rigid body transformation and provide a more accurate set of point correspondences to the generalized ICP algorithm by taking advantage of high dimensional image feature descriptors. We introduced the novel concept of a camera consensus matrix and showed how it can be used to intrinsically provide a set of geometrically-consistent putative correspondences purely using the image data alone. We call this approach "visually bootstrapped GICP", and it is a completely data driven approach that does not require any external initial guess (e.g., from odometry). In the experiments performed with real world data, we have shown that the bootstrapped generalized ICP algorithm is more robust and gives accurate results even when the overlap between the two scans reduces to less than 50%.

ACKNOWLEDGMENTS

This work was supported by Ford Motor Company via the Ford-UofM Alliance (Award #N009933).

REFERENCES

[1] G. Pandey, J. McBride, S. Savarese, and R. Eustice, "Extrinsic calibration of a 3d laser scanner and an omnidirectional camera," in 7th IFAC Symposium on Intelligent Autonomous Vehicles, 2010.

[2] P. J. Besl and N. D. McKay, "A method for registration of 3-d shapes," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 14, no. 2, pp. 239–256, 1992.

[3] Y. Chen and G. Medioni, "Object modelling by registration of multiple range images," Image Vision Comput., vol. 10, no. 3, pp. 145–155, 1992.

[4] M. Alshawa, "Icl: Iterative closest line a novel point cloud registration algorithm based on linear features," ISPRS, 2007.

[5] S. Rusinkiewicz and M. Levoy, "Efficient variants of the icp algorithm," in 3DIM, 2001, pp. 145–152.

[6] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "Introduction to algorithms, second edition," 2001.

[7] P. Biber, S. Fleck, and W. Straßer, "A probabilistic framework for robust and accurate matching of point clouds," in DAGM-Symposium, 2004, pp. 480–487.

[8] D. Hhnel and W. Burgard, "Probabilistic matching for 3d scan registration," in In.: Proc. of the VDI - Conference Robotik 2002 (Robotik), 2002.

[9] P. Biber, "The normal distribution transform: A new approach to laser scan matching," in Proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS), vol. 3, 2003, pp. 2743–2748.

[10] A. V. Segal, D. Haehnel, and S. Thrun, "Generalized-icp," in Robotics: Science and Systems, 2009.

[11] A. Johnson and S. B. Kang, "Registration and integration of textured 3-d data," in Image and Vision Computing, 1996, pp. 234–241.

[12] D. Akca, "Matching of 3d surfaces and their intensities," ISPRS Journal of Photogrammetry and Remote Sensing, vol. 62, pp. 112–121, 2007.

[13] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," International Journal of Computer Vision, vol. 60, no. 2, pp. 91–110, 2004.

[14] H. Bay, T. Tuytelaars, and L. V. Gool, "Surf: Speeded up robust features," in In ECCV, 2006, pp. 404–417.

[15] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," in Communications of the ACM, Volume 24, Number 6, 1981.

[16] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-d point sets," IEEE Trans. Pattern Anal. Mach. Intell., vol. 9, no. 5, pp. 698–700, 1987.

[17] Pointgrey. (2009) Spherical vision products: Ladybug3. [Online]. Available: www.ptgrey.com/products/ladybug3/index.asp

[18] M. Kaess, A. Ranganathan, and F. Dellaert, "iSAM: Incremental smoothing and mapping," IEEE Trans. on Robotics, TRO, vol. 24, no. 6, pp. 1365–1378, Dec 2008.

(a) iSAM with generalized ICP open-loop.

(b) iSAM with generalized ICP closed-loop.

(c) iSAM with bootstrapped generalized ICP open-loop.

(d) iSAM with bootstrapped generalized ICP closed-loop.

Fig. 9. iSAM output with input pose constraints coming from generalized ICP and bootstrapped generalized ICP. Here, the red trajectory is the ground truth coming from GPS and the green trajectory is the output of the iSAM algorithm. The *start* and *end* point of the trajectory are the same and is denoted by the black dot.