Robust and Fast 3D Scan Alignment using Mutual Information

Nikhil Mehta[1], James R. McBride[2] and Gaurav Pandey[2]

Abstract— **This paper presents a mutual information (MI) based algorithm for the estimation of full 6-degree-of-freedom (DOF) rigid body transformation between two overlapping point clouds. We first divide the scene into a 3D voxel grid and define simple to compute features for each voxel in the scan. The two scans that need to be aligned are considered as a collection of these features and the MI between these voxelized features is maximized to obtain the correct alignment of scans. We have implemented our method with various simple point cloud features (such as number of points in voxel, variance of z-height in voxel) and compared the performance of the proposed method with existing point-to-point and point-to-distribution registration methods. We show that our approach has an efficient and fast parallel implementation on GPU, and evaluate the robustness and speed of the proposed algorithm on two real-world datasets which have variety of dynamic scenes from different environments.**
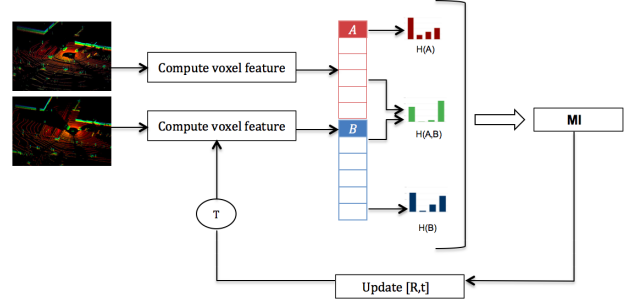
Fig. 1: An overview of mutual information (MI) based scan alignment framework. MI maximization is applied over the voxelized-features computed from two partially overlapping scans.

I. INTRODUCTION

A 3D alignment algorithm to determine the relative rigid body transformation between two partially overlapping scans is an underpinning tool for many applications in mobile robotics including localization, mapping and navigation systems. In this work, we consider a robot which obtains two 3D scans (A and B) from two poses P_0 and P_1 via a 3D laser scanner. Provided that some part of the environment is common to both scans, it is generally possible to find a rigid-body transformation T that can project the points in P_1 so that they align with P_0. The solution to the process of scan alignment (T) is parameterized by six values: three translation components (t_x, t_y and t_z) and three rotation components (θ_x, θ_y and θ_z). The reason that scan alignment problem is at the center of most navigation, mapping and localization systems, is simply because the rigid body transformation T derived from alignment is of higher quality than odometry estimate (due to wheel slippage and surface irregularities).

The primary challenge in the problem is to minimize the runtime complexity while maximizing the robustness of the solution. Most existing methods are either designed around computationally-efficient local searches which are not robust to initialization or global search methods which are computationally intense. At vehicle's dead-reckoning error, the initial estimate can be far from global maximum resulting in large errors when using local-search methods. To counter this problem, most implementations currently involve running the registration algorithm at high frequency rate resulting in a constant load on computation resources.

Our approach poses the alignment problem in a MI (mutual information) maximization framework: it finds the rigid body transformation that maximizes MI between the features of two scans. An overview of our approach is illustrated in Figure 1. We show that even primitive features like variance and number of points give results better than the other state of the art point-to-point and point-to-distribution methods for large prior errors from odometry.

The central contributions of this paper are:

- We present a robust and fast framework for scan alignment by posing the problem in a MI maximization framework with simple voxelized features.
- Unlike other 3D scan alignment methods, our approach allows us to consider the no-feature voxels (unoccupied region in scene).
- We present a detailed empirical evaluation of our method in different environments: both urban and rural scenes. We compare our approach with a point-to-point and a point-to-distribution alignment method.
- We show how the proposed approach has an efficient fast and robust GPU implementation, freeing the CPU for other important tasks.

The quality and robustness of our method along with it's ability to work in real-time, makes it ideal for mobile robotic systems in which accuracy is of high importance.

In the following section we present a brief overview of the prior work (Section II). In section III, we describe the proposed framework. Empirical evaluation and comparison with other methods along with runtime analysis is shown in section IV.

[1]N. Mehta is with the Computer Engineering Department of IIT Kanpur, India. `nikhil@cse.iitk.ac.in`

[2]G. Pandey and J. R. Mcbride are with Ford Motor Company, Dearborn, USA. `gpandey2@ford.com, jmcbride@ford.com`.

II. RELATED WORK

Iterative Closest Point (ICP) [1] is one of the most popular scan registration algorithms used to estimate the optimal transformation between two overlapping scans. In ICP, closest points between scan A and reference scan B are used to obtain a closed-form solution by optimizing the sum of squared distances (usually Euclidean distance). Performing the nearest neighbor search in ICP becomes a bottleneck due to its high computational cost, however, using a K-D Tree [2] does mitigate the problem to an extent. Iterative Dual Correspondence (IDC) [3] generates corresponding points for both rotation and translation separately, with optimization done in an alternate fashion. This improves the alignment accuracy when the initial estimate has large rotational error. Iterative Closest Line (ICL) [4], [5], [6] is a variant of ICP in which instead of matching the points in both the scans, the query points in scan B are matched to lines extracted from points in reference scan A. Generalized ICP (GICP) [7], attaches a probabilistic model in estimating the correct corresponding points by taking the covariance structure derived from the local neighborhood in the environment.

Another widely used algorithm is Normal Distribution Transform (NDT) which was initially proposed for 2D data [8] and was later extended to three-dimensions. 3D-NDT [9] is a point-to-distribution method in which the maximum likelihood estimate (MLE) of points in B is maximized over the distribution of points in a gaussian mixture model derived from A. Supervoxel based NDT (SV-NDT) [10] is an extension to 3D-NDT in which segmentation on the basis of local-spatial structure is done over A before creating a GMM model.

The problem of scan alignment with fused sensor data as input has also been extensively researched in the past two decades. Some of the work includes incorporating color information in the ICP framework ([11], [12]) by augmenting RGB color channels to 3D coordinates and by exploiting the co-registration of 3D data with the available camera imagery to associate scale invariant feature transform (SIFT) [13] or speeded up robust features (SURF) [13] features to the 3D points as in [14].

Mutual information based alignment methods was first proposed in [15] for multi-modality medical images. Since then literature has been filled with work inspired by mutual information, these include minimization of distribution of joint histograms [16], data alignment from multiple modalities (different sensors) as in [17], [18], [19], and [20]. In [21], a comprehensive survey of mutual information based techniques being used in medical images is presented.

The proposed method is closely related to FPFH [20] mainly because we use mutual information to compute the registration parameters. However, the feature selection and histogram creation method is different. In [20] high dimensional features are computed using FPFH which are then quantized into one of the precomputed codewords. There is no denying that FPFH are better descriptors of a scan than simple voxelized features like z-variance, but in a MI
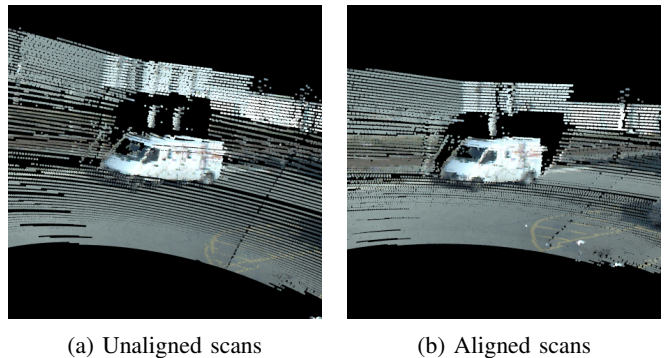

(a) Unaligned scans   (b) Aligned scans

Fig. 2: This figure shows the cropped 3D textured point cloud of a pair of scan transformed to the same frame of reference. (a) Transformation used is from the initial estimate. (b) Transformation applied is first improved/aligned using our approach.

based framework, we don't need to find correspondences between patches, therefore computing FPFH only increases the run-time complexity due to nearest neighbor search and normal computation at each point. Moreover, unlike voxelized features, FPFH fails to take into account the empty region in the scan.

III. METHODOLOGY

A. Approach

A qualitative result of the proposed MI based approach using voxelized features is shown in Figure 2. In order to estimate the correct rigid body transformation T, represented by a $[4 \times 4]$ transformation matrix, we first initialize our scene with a large 3D grid with resolution R_v $(1 \times 1 \times 1)$ that encompasses both the scans A and B. We create two voxel-point mapping tables by finding the corresponding voxel (the voxel which contains the point) for all points in scan A and B. Now we calculate simple features like (i) Variance of z-height, or (ii) number of points in voxel. It is important to note that unoccupied voxels are also included in our voxel-point mapping table with ϕ (no-feature) value; this results in a voxel-feature mapping that contains large number of voxels with feature ϕ. Intuition behind this approach is that in a perfectly aligned state not only the occupied voxels should be aligned, but unoccupied voxels should also be aligned in the overlapping region.

To obtain the statistical dependence using mutual information of two partially overlapping scans, we define an overlapping region which is a subgrid parameterized by six variables $[x^o_{min}, x^o_{max}, y^o_{min}, y^o_{max}, z^o_{min}, z^o_{max}]$ (where 'o' is the overlapping region). We consider the feature maps of this overlapping subgrid as two random variables representing the structure of the environment. These random variables are used to calculate the mutual information of both the scans. The mutual information between these two random variables is the amount of information obtained about one variable, through the information about the other variable. Therefore, under the correct rigid body transformation, we
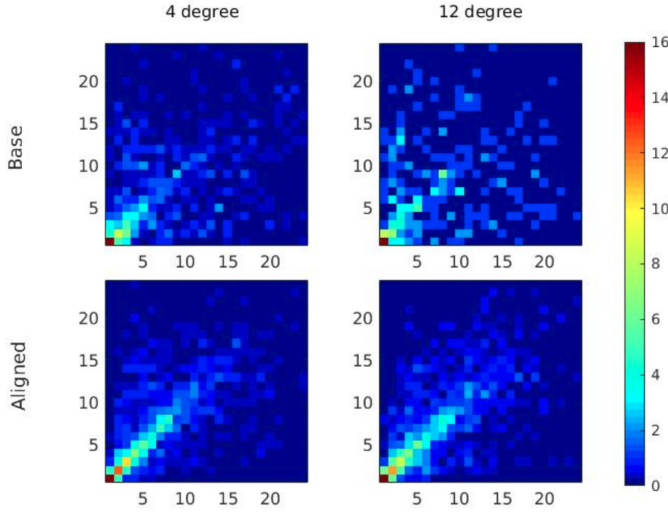
Fig. 3: This figure shows 4 joint histograms: 2 before alignment (Row 1) and 2 after alignment using the proposed method (Row 2). Each column represents two different test cases: in Column 1 the scan pair had the initial heading error of 4° and in Column 2 the scan pair had the initial heading error of 12°. X-axis and Y-axis of all the 4 joint histograms represent the variance of Z in a voxel for scan A and scan B respectively. Note how the dispersion in the joint histogram decreases (forming a straight line) after alignment in both the test cases. It can also be seen that the dispersion in joint histogram increases as we increase the heading error from Column 1 - Column 2 (4° to 12°). Best viewed in color.

can expect the information from one of the scans to give maximum amount of information about the other scan, thus rendering the MI to be maximum between the feature maps.

$$\eta = \arg\max_{\eta} MI(X, Y; \eta) \qquad (1)$$

where $\eta = [t_x, t_y, t_z, \theta_x, \theta_y, \theta_z]$ are the required transformation parameters, $[X, Y]$ are the respective random variables representing the features of scan A and B in the corresponding overlapping region and $MI(X, Y; \eta)$ is the mutual information of the overlapping voxels under the transformation η.

$$MI(X, Y; \eta) = H(X; \eta) + H(Y; \eta) - H(X, Y; \eta) \qquad (2)$$

where $H(X; \eta)$ and $H(Y; \eta)$ are the entropies of the random variables X and Y respectively, and $H(X, Y; \eta)$ is the joint entropy of the two variables at transformation η.

$$H(X; \eta) = -\sum_{x \in X} \mathsf{P}_X(x; \eta) \log \mathsf{P}_X(x; \eta) \qquad (3)$$

$$H(Y; \eta) = -\sum_{y \in Y} \mathsf{P}_Y(y; \eta) \log \mathsf{P}_Y(y; \eta) \qquad (4)$$

$$H(X, Y; \eta) = -\sum_{x \in X} \sum_{y \in Y} \mathsf{P}_{XY}(x, y; \eta) \log \mathsf{P}_{XY}(x, y; \eta) \qquad (5)$$

Algorithm 1 Alignment of two scans using MI framework

Input: Two scans A and B with some partial overlap. Initial guess of the rigid-body transformation T_o.

Output: Estimated registration parameter T

    Initialization :
1: Extract the associated voxel ids for all points in scan A
2: Create scan A feature map for all voxels
    Alignment :
3: $T_c = T_o$
4: **while** MI not converged **do**
5:     Transform scan B using current rigid body transformation T_c
6:     Extract associated voxel ids for points in scan B
7:     Create scan B feature map for all voxels
8:     Define an overlapping bounding region:
    $x^o_{min} = max(x^A_{min}, x^B_{min})$
    $y^o_{min} = max(y^A_{min}, y^B_{min})$
    $z^o_{min} = max(z^A_{min}, z^B_{min})$
    $x^o_{max} = min(x^A_{max}, x^B_{max})$
    $y^o_{max} = min(y^A_{max}, y^B_{max})$
    $z^o_{max} = min(z^A_{max}, z^B_{max})$
9:     Consider random variable X and Y as the voxelized feature map for all voxels lying inside the overlap of scan A and scan B respectively
10:   Compute the marginal and joint entropy using the equations 3, 4 and 5
11:   Calculate MI using 2
12:   Update T_c using Nelder-Mead optimization
13: **end while**
14: **return** T_c

Here $\mathsf{P}_X(x; \eta)$ is the marginal probability for $X = x$, $\mathsf{P}_Y(y; \eta)$ is the marginal probability for $Y = y$ and $\mathsf{P}_{XY}(x, y; \eta)$ is the joint probability for $X = x$ and $Y = y$ when transformation parameters are η. For simplicity, we have taken x and y to be either the variance in z or the number of points in a voxel.

In each iteration, the MI value changes and is maximized when all voxels (occupied and unoccupied) are aligned. It is important to note that with every iteration, the overlapping region will change and the correct subgrid will be found when the MI is maximum. Figure 3 represents the joint-distribution before (row 1) and after (row 2) MI maximization for two pairs of scan with different initial error. It can be seen that the joint distribution converges to a line passing through origin and becomes relatively uniform after the correct alignment.

B. Optimization

The cost function (1) is maximized at the correct value of rigid body transformation. Therefore, any optimization technique that iteratively converges to the local optimum can be used here. Some of the commonly used optimization techniques compute the gradient or hessian of the cost function ([22], [23] and [24]). There are also gradient free direct optimization methods like *pattern search* [25] or

Dataset	Environment	Total Scans	deviation(t_x)	deviation(t_y)	deviation(t_z)	deviation(θ_x)	deviation(θ_y)	deviation(θ_z)
Ford Campus Vision and Lidar	Urban	3538	0.65	2.87	0.05	0.02	0.02	0.23
Kitti Vision Benchmark Suite	Rural / Highway	4919	2.67	0.31	0.04	0.01	0.01	0.08

TABLE I: Standard deviation (in meters for translation and radians for rotation) of the ground truth for all 6 DOF is shown. This analysis involves all the scan-pairs considered in our experiments. All floating point numeric values are rounded to two decimal places. It can be seen that the variance in t_x, t_y and θ_z is more than the variance in t_z, θ_x and θ_y. Although, we have shown this only for two different robots, this is in general true for any wheeled-robot. We use this information in initializing the simplex particles (N+1 points) in the optimization phase.

simulated annealing [26]. In this work we use one such direct optimization method called *Nelder-Mead* optimization [27].

Nelder-Mead method initializes a simplex with (N+1) points on the cost surface, where N is the number of DOF (six in our case) in the optimization phase. In each iteration it tries to improve these set of points by series of steps (reflecting, expanding, shrinking or contracting) to obtain the point which minimizes the cost function. NM-Simplex is sensitive to initial simplex size [s_x, s_y, s_z, s_{roll}, s_{pitch}, s_{yaw}]. Choosing an initial simplex large can cause unnecessary steps in areas of little interest, while a small simplex can lead to a narrow search on the cost surface increasing the computation. This problem can be alleviated by making a simple assumption that most of the relative traversal will be in t_x, t_y and θ_z for wheeled robots. The intuition being that for any wheeled-robotic platform, motion is constrained in roll, pitch and z(height). Similar observation was also made in the localization method proposed in [28]. Making this assumption allows us to explore large range for t_x, t_y and θ_z on cost surface, while at the same time a constrained initial search in t_z, θ_x and θ_y. We verify this assumption, by calculating the deviation of all 6-DOF in Table I.

As this optimization technique is based on heuristics, optimization might not always lead to the optima. This is indeed a disadvantage of this optimization technique. While there exist methods that allow approximating the gradient of the cost function, however, the computation involved increases the run time of the algorithm. In the section below, we show that even after using a heuristic optimization, the results are close to the ground truth.

The complete algorithm for obtaining the rigid body transformation is given in (Algorithm 1).

IV. EXPERIMENTS AND RESULTS

We show results for two types of feature used in our MI framework: (i) Variance of z-height in voxel (MI-VARZ), and (ii) Number of points in voxel (MI-N). We have used two point cloud datasets to assess the performance of the proposed approach. Scans from both these datasets are registered using GICP, 3D-NDT, MI-N, and MI-VARZ. The comparison factors were accuracy and runtime for different initial estimates. Datasets used in our experiments were Ford Campus Vision and Lidar dataset [29] and odometry

dataset from the Kitti Vision Benchmark Suite [30]. Both the datasets consist of 3D scans collected from a test vehicle with a Lidar mounted on it and have ground truth pose information available from a highly accurate inertial navigation system (INS).

Implementation of GICP is taken from the open source version available at [31] and 3D-NDT is taken from Point Cloud Library (PCL) [32]. Grid resolution (R_v) for 3D-NDT, MI-VARZ and MI-N is set as 1m. In our implementation we chose the initial simplex for optimization as [$s_x = 8$, $s_y = 8$, $s_z = 1$, $s_{roll} = 0.1$, $s_{pitch} = 0.1$, $s_{yaw} = 0.8$].

A. Ford Campus Vision and Lidar Dataset

In this experiment we compare the translation and rotation accuracy of the proposed method with GICP and 3D-NDT on Ford Dataset. To estimate the translation accuracy, we consider equi-spaced reference scan (A) in the downtown test run and sample 10 query scans (B) which are 1-10m from A. Similarly for rotation, all B query scans that have relative rotation upto $20°$ (with translation error less than 5m) from scan A are chosen. In total 3538 scan pairs from Ford dataset were considered to plot error results shown in Figure 4.

B. Kitti Vision Benchmark Suite

In this experiment, we use the scans from Kitti dataset for testing both translation and rotation accuracy. Here also, we sample scans that have translation 1-10m and rotation upto $12°$ (plot range is reduced here as the number of scan pairs with translation error less than 5m and rotation greater than $12°$ are not enough) resulting in a total of 4919 scan pairs from a single session run. Results are shown in Figure 5.

C. Discussion of results

It can be seen, that when the initial estimate of transformation parameters are good, all 4 methods perform reasonably good for both the datasets. However, 3D-NDT fails to converge when the initial error is increased: this can be explained by the discontinuity in score function due to the jump caused when the query scan B passes one of the cell boundaries. In [33] and [34], convergence of 3D-NDT and it's dependence on voxels resolution is explained in detail.

It should be noted that GICP works better when we have a good initial guess for translation parameters. However, it fails to converge to the correct optima when the initial guess
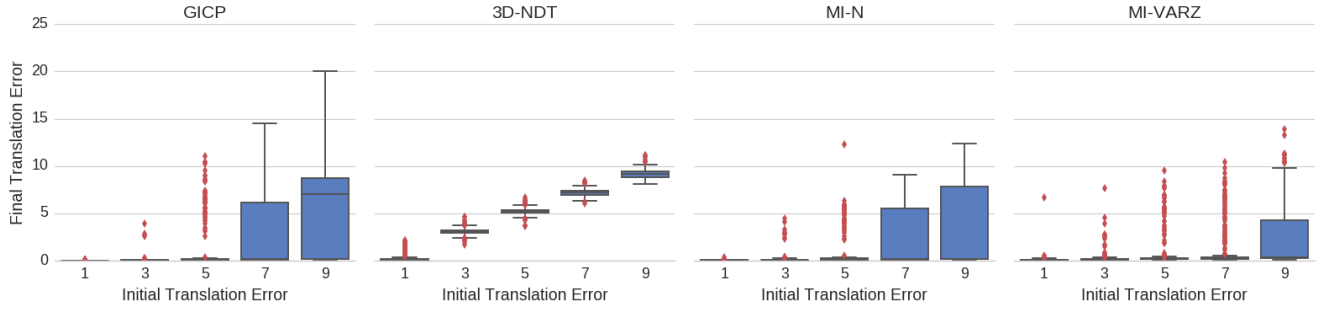
(a) Boxplot depicting the variance and median of final translation error (in meters) versus initial error in translation parameters
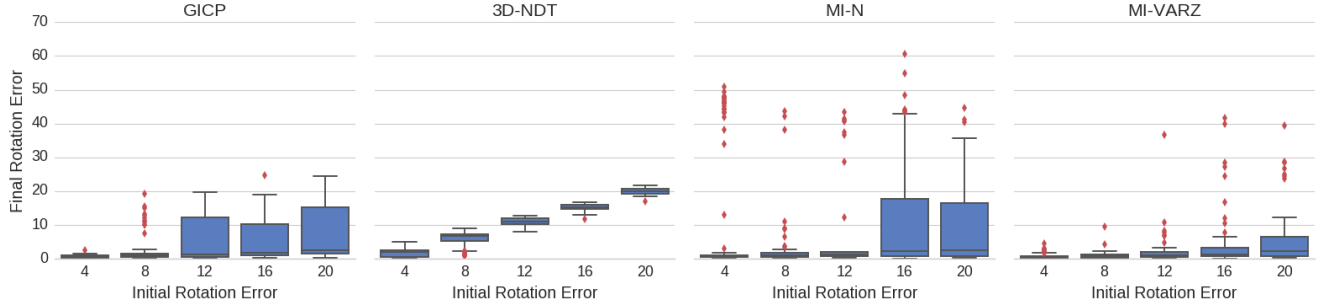
(b) Boxplot depicting the variance and median of final rotation error (in degrees) versus initial error in rotation parameters

Method	1	2	3	4	5	6	7	8	9	10
GICP	0.03	0.07	0.11	0.34	1.06	2.49	2.60	3.99	5.42	6.73
3D-NDT	0.29	1.69	3.08	4.12	5.22	6.17	7.17	8.22	9.17	10.29
MI-N	0.05	0.10	0.18	0.43	0.68	1.22	2.09	2.68	2.80	3.23
MI-VARZ	0.06	0.10	0.28	0.30	0.70	0.95	1.37	2.01	2.39	2.70

(c) Mean error (in meters) after alignment. Error in initial translation parameters is increased (from left: 1-10m).

Method	2	4	6	8	10	12	14	16	18	20
GICP	0.41	0.72	1.00	1.14	0.88	0.81	1.23	1.78	3.26	5.05
3D-NDT	0.52	1.82	4.48	5.98	8.96	11.03	13.17	15.24	17.43	19.44
MI-N	1.18	1.90	2.39	4.79	4.19	5.66	2.60	5.61	4.91	1.49
MI-VARZ	0.43	0.70	0.75	0.72	0.92	2.56	0.89	4.11	1.98	1.48

(d) Mean error (in degrees) after alignment. Error in initial rotation parameters is increased (from left: 2°-20°).

Fig. 4: Ford Campus Vision and Lidar: Comparison between the proposed method (MI-VARZ and MI-N) with GICP and 3D-NDT. (a) and (b), depict the variance and median of the error magnitude (L2 norm) versus initial error for translation and rotation. In (c) and (d) we summarize the success rate of GICP, 3D-NDT, MI-N and MI-VARZ using mean error results in translation and rotation calculated by taking the L2 norm of difference with ground truth. Error analysis in (a) and (b) is done using descriptive statistical boxplots. In each boxplot, blue box represents the range from 25th percentile to 75th percentile and the black line inside the box shows the median. Results farther than 1.5 times the error at box-edges are considered outliers and are represented as red dots outside the whiskers. Ends of the whisker below and above the box represent the minimum and maximum values respectively.

is poor. This can be explained from cost surface of GICP algorithm. As seen in Figure 6a, GICP has several local optima causing the gradient based optimization methods to fail. Figure 6c also depicts this problem, when the initial heading error is large GICP can converge to incorrect transformation. In contrast, in Figure 6b it can be seen that our MI based

method has a single optima corresponding to the correct rigid-body-transformation for the same range of initial error. It is important to note that this might not always be the case. In scan pairs with multiple dynamic objects and large initial error, the heuristic based optimization in MI-VARZ and MI-N can converge at an optima far from the correct solution.
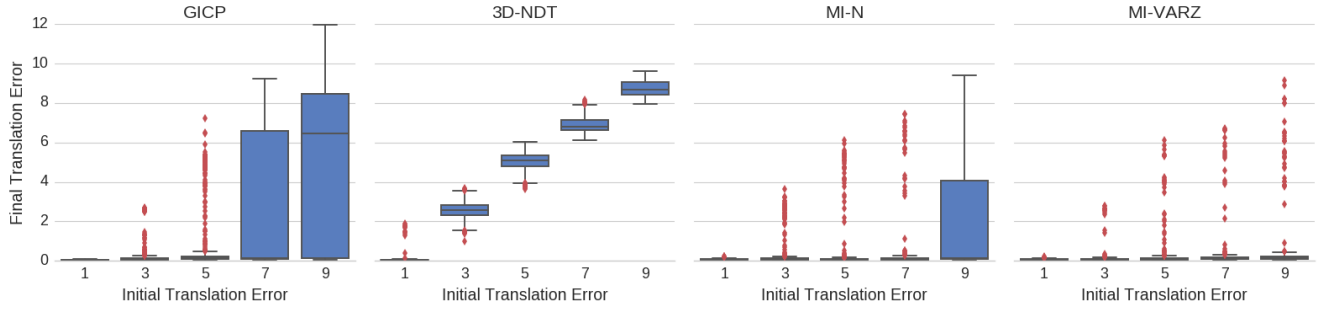
(a) Boxplot depicting the variance and median of final translation error (in meters) versus initial error in translation parameters
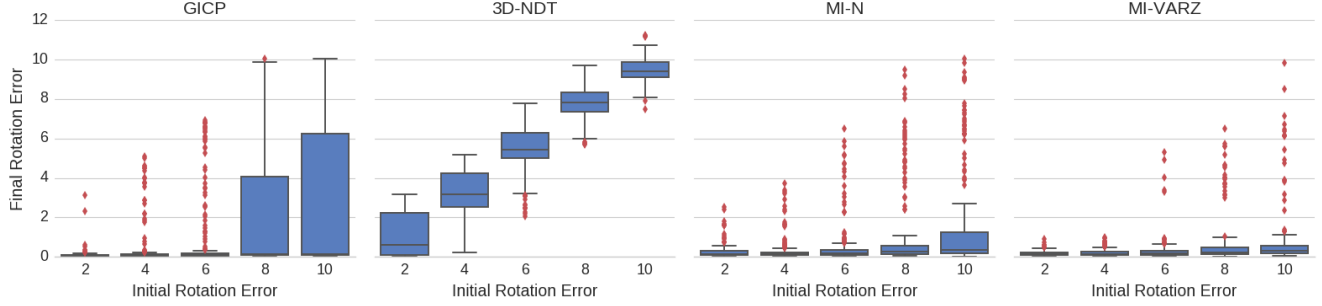
(b) Boxplot depicting the variance and median of final rotation error (in degrees) versus initial error in rotation parameters

Method	1	2	3	4	5	6	7	8	9	10
GICP	0.03	0.13	0.21	0.60	0.74	1.41	2.53	2.91	4.71	5.42
3D-NDT	0.24	1.62	2.55	3.92	5.01	5.86	6.91	7.88	8.73	10.01
MI-N	0.06	0.10	0.22	0.28	0.25	0.47	0.51	0.54	1.90	3.38
MI-VARZ	0.06	0.06	0.12	0.09	0.21	0.41	0.42	0.36	0.68	1.23

(c) Mean error (in meters) after alignment. Error in initial translation parameters in increased (from left: 1-10m).

Method	2	4	6	8	10	12
GICP	0.13	0.34	0.82	2.08	2.70	2.90
3D-NDT	1.10	3.21	5.43	7.82	9.49	11.47
MI-N	0.30	0.27	0.51	1.20	1.85	0.31
MI-VARZ	0.17	0.20	0.31	0.67	0.94	0.36

(d) Mean error (in degrees) after alignment. Error in initial rotation parameters is increased (from left: 2°-12°).

Fig. 5: Kitti Vision and Benchmark Suite: Comparison between the proposed method (MI-VARZ and MI-N) with GICP and 3D-NDT. (a) and (b), depict the variance and median in the error for different magnitude error in initial estimate for translation and rotation. In (c) and (d) we summarize the success rate of GICP, 3D-NDT, MI-N and MI-VARZ using mean error results in translation and rotation calculated by taking the L2 norm of difference with ground truth. Error analysis in (a) and (b) is done using descriptive statistical boxplots. In each boxplot, blue box represents the range from 25th percentile to 75th percentile and the black line inside the box shows the median. Results farther than 1.5 times the error at box-edges are considered outliers and are represented as red dots outside the whiskers. Ends of the whisker below and above the box represent the minimum and maximum values respectively.

Despite that, we observe that our MI based method performs better than GICP in most cases.

Both MI-VARZ and MI-N have better results for translation as we increase the prior translation error. For the Kitti Dataset, it can be seen that MI-VARZ has average translation error less than 0.5m for the initial translation error up till 8m and MI-N till 6m, which is in contrast to results from GICP where we see an average error of 0.6m at 4m of initial translation error. In the Ford dataset, MI-VARZ and MI-N have error less than 0.5m up till 4m and GICP till 3m.

As for the rotation error, in Kitti Dataset we see a similar trend where MI-VARZ performs better than GICP as we

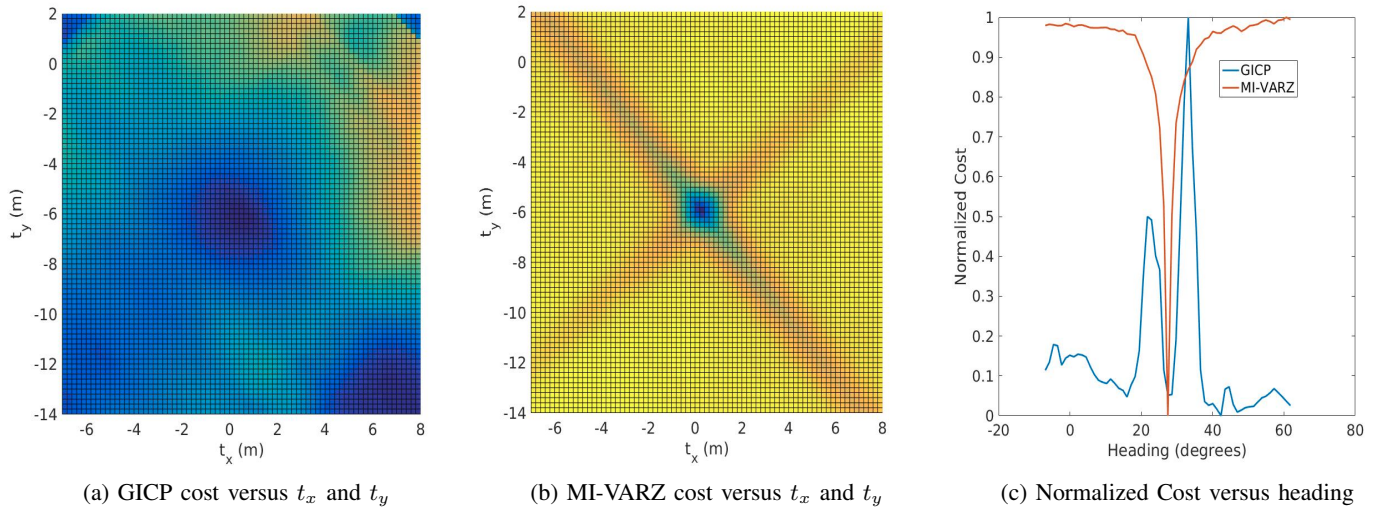| (a) GICP cost versus t_x and t_y | (b) MI-VARZ cost versus t_x and t_y | (c) Normalized Cost versus heading |

Fig. 6: Cost-function plots for GICP and MI-VARZ. In (a) and (b), we have plotted the top view of GICP and MI-VARZ cost-function surface versus the translation parameters t_x and t_y, while all other parameters are fixed to true ground-truth value. The correct translation parameters are ($t_x = 0.40, t_y = -5.99$). In (c), we compare the cost-function of GICP(Blue) and MI-VARZ(Orange) versus the heading. True heading parameter is ($\theta_z = 27.45°$). It can be seen from both (a) and (c) that GICP can converge at a local optima far from global optimum when the initial error is large.

increase the initial error. In the Ford dataset however, both GICP and MI-VARZ produce mixed results. This is primarily due to unstructured areas and multiple dynamic objects in the urban scene. In such cases, the heuristic based search in MI-VARZ leads to outliers (See Figure 4b and 4d for initial error $16°$) which are relatively farther from the correct rotation parameters than GICP outliers. Yet in most cases (Figure 4b) MI-VARZ has performance better than GICP. We also observe that MI-N doesn't perform as good as MI-VARZ and GICP. In general too, MI-N has less success rate as compared to MI-VARZ, this is because the number of points in a voxel do not convey any information about the local-structure unlike the case in variance of z.

D. Runtime analysis

In this section we compare the runtime of GICP with MI-N and MI-VARZ. We do not consider the 3D-NDT in this analysis as it fails to converge when the initial error is high. The average runtime of GICP, MI-N and MI-VARZ is shown in Table II. All the implementations were executed on a system powered by Intel Core i7-7700HQ CPU@ 2.80GHz $\times$ 8 and GeForce GTX 1050Ti. We used [35] to optimize the GICP by implementing the point-point search queries in GPU memory. The runtime of GICP after this optimization is also showed in Table II.

Another important challenge is the variability of runtime with change in the initial error. The runtime of GICP varies when we change the error in translation, whereas MI-N and MI-VARZ have runtime independent of the initial error.

In Table III, we show that all major steps in our proposed framework can be easily offloaded to GPU with minimal overhead. This is in contrast to other point-to-point alignment methods like GICP, which require building nearest neighbor structures (like K-D Tree) in GPU. The preprocessing step

Initial Error	GICP	GICP-GPU	MI-N	MI-VARZ
(m)	(s)	(s)	(s)	(s)
1	2.67	1.64	1.26	0.49
3	4.00	2.25	1.27	0.50
5	6.92	4.42	1.26	0.50
7	10.72	7.25	1.23	0.49
9	14.26	10.13	1.22	0.51

TABLE II: Runtime analysis of GICP (Original), GICP (GPU Search), MI-N and MI-VARZ. MI-N and MI-VARZ have negligible variability in runtime with respect to quality of the initial guess. However, the GICP runtime increases as the initial error is increased.

in GICP for nearest neighbor search takes most of the time in alignment process. Whereas in our MI based framework, the only bottleneck is GPU memory allocation (along with gpu context initialization).

MI Steps	Kernel	Total Time(ms)
GPU Allocate memory	-	189.00
Data CPU to GPU	-	0.5
Voxel Mapping	Map	5.87
Find overlap	Reduce	33.06
Feature histograms	Map	100.10
Entropy Calculation	Reduce	8.45

TABLE III: GPU profiling for a scan pair while executing MI-VARZ. Second column depicts the type of gpu kernel for the operation and third column is the total time (ms) spent in kernel execution for all iterations in the alignment process.

V. CONCLUSION

In this paper, we report a MI based scan alignment algorithm that maximizes the mutual information between the voxelized features of two partially overlapping scans by calculating a single-dimensional feature in a voxel. Our approach allows us to consider the no-feature voxels in our cost function with the intuition being: that both scans, when aligned, should have same number of unoccupied voxels in the overlapping region. The proposed method is tested and compared with a point-to-point and point-to-distribution method on two real-world datasets covering wide range of dynamic scenes: urban and rural. We see that our method performs relatively better even for large initial errors in transformation. Although, we implemented our method with data from a single sensor (lidar), it can easily be extended to fused-sensor data (lidar-camera). We show that our method has a fast GPU implementation which allows computation to be offloaded to GPU with minimal overhead.

REFERENCES

[1] P. J. Besl and N. D. McKay, "A method for registration of 3-d shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, pp. 239–256, Feb. 1992.

[2] M. Greenspan and M. Yurick, "Approximate k-d tree search for efficient icp," in *Fourth International Conference on 3-D Digital Imaging and Modeling, 2003. 3DIM 2003. Proceedings.*, pp. 442–448, Oct 2003.

[3] F. Lu and E. Milios, "Robot pose estimation in unknown environments by matching 2d range scans," *Journal of Intelligent and Robotic Systems*, vol. 18, no. 3, pp. 249–275, 1997.

[4] M. C. Bosse, *ATLAS: a framework for large scale automated mapping and localization.* PhD thesis, Massachusetts Institute of Technology. Dept. of Electrical Engineering and Computer Science, 2004.

[5] E. B. E. B. Olson, *Robust and efficient robotic mapping.* PhD thesis, Massachusetts Institute of Technology. Dept. of Electrical Engineering and Computer Science, 2008.

[6] A. Censi, "An icp variant using a point-to-line metric," in *IEEE International Conference on Robotics and Automation*, 2008.

[7] A. Segal, D. Haehnel, and S. Thrun, "Generalized-icp," in *Proceedings of Robotics: Science and Systems*, (Seattle, USA), June 2009.

[8] P. Biber and W. Straßer, "The normal distributions transform: a new approach to laser scan matching," in *IROS*, 2003.

[9] M. Magnusson, R. Elsrud, L. erik Skagerlund, and T. Duckett, "3d modelling for underground mining vehicles," in *In SimSafe 2005, Proceedings of the Conference on Modeling and Simulation for Public Safety*, 2005.

[10] J. W. Kim and B. H. Lee, "Robust and fast 3-d scan registration using normal distributions transform with supervoxel segmentation," *Robotica*, vol. 34, pp. 1630–1658, 007 2016.

[11] A. E. Johnson and S. B. Kang, "Registration and integration of textured 3-d data," in *Proceedings of the International Conference on Recent Advances in 3-D Digital Imaging and Modeling*, NRC '97, (Washington, DC, USA), pp. 234–, IEEE Computer Society, 1997.

[12] G. Godin, D. Laurendeau, and R. Bergevin, "A method for the registration of attributed range images," in *3DIM*, 2001.

[13] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vision*, vol. 60, pp. 91–110, Nov. 2004.

[14] G. Pandey, J. McBride, S. Savarese, and R. Eustice, *Visually bootstrapped generalized ICP*, pp. 2660–2667. 2011.

[15] R. P. Woods, J. C. Mazziotta, and S. R. Cherry, "MRI-PET registration with automated algorithm," *J Comput Assist Tomogr*, vol. 17, no. 4, pp. 536–546, 1993.

[16] D. L. G. Hill, D. J. Hawkes, N. A. Harrison, and C. F. Ruff, *A strategy for automated multimodality image registration incorporating anatomical knowledge and imager characteristics*, pp. 182–196. Berlin, Heidelberg: Springer Berlin Heidelberg, 1993.

[17] P. Viola and W. M. Wells III, "Alignment by maximization of mutual information," *International Journal of Computer Vision*, vol. 24, no. 2, pp. 137–154, 1997.

[18] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens, "Multimodality image registration by maximization of mutual information," *IEEE Trans Med Imaging*, vol. 16, pp. 187–198, Apr 1997.

[19] F. Maes, D. Vandermeulen, and P. Suetens, "Comparative evaluation of multiresolution optimization strategies for multimodality image registration by maximization of mutual information," *Med Image Anal*, vol. 3, pp. 373–386, Dec 1999.

[20] G. Pandey, J. R. McBride, S. Savarese, and R. M. Eustice, "Toward mutual information based automatic registration of 3d point clouds," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2698–2704, Oct 2012.

[21] J. P. W. Pluim, J. B. A. Maintz, and M. A. Viergever, "Mutual-information-based registration of medical images: a survey," *IEEE Transactions on Medical Imaging*, vol. 22, pp. 986–1004, Aug 2003.

[22] E. T. Whittaker and G. Robinson, "The newton raphson method," in *The Calculus of Observations: A Treatise on Numerical Mathematics*, vol. 44, pp. 84–87, 1967.

[23] D. W. Marquardt, "An algorithm for least-squares estimation of non-linear parameters," *Journal of the Society for Industrial and Applied Mathematics*, vol. 11, no. 2, pp. 431–441, 1963.

[24] K. LEVENBERG, "A method for the solution of certain non-linear problems in least squares," *Quarterly of Applied Mathematics*, vol. 2, no. 2, pp. 164–168, 1944.

[25] V. Torczon, "On the convergence of pattern search algorithms," *SIAM Journal on Optimization*, vol. 7, no. 1, pp. 1–25, 1997.

[26] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.

[27] J. A. Nelder and R. Mead, "A simplex method for function minimization," *The Computer Journal*, vol. 7, no. 4, p. 308, 1965.

[28] R. W. Wolcott and R. M. Eustice, "Fast lidar localization using multiresolution gaussian mixture maps," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pp. 2814–2821, IEEE, 2015.

[29] G. Pandey, J. R. Mcbride, and R. M. Eustice, "Ford campus vision and lidar data set," *Int. J. Rob. Res.*, vol. 30, pp. 1543–1552, Nov. 2011.

[30] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3354–3361, June 2012.

[31] A. Segal, "Generalized icp reference implementation."

[32] R. B. Rusu and S. Cousins, "3d is here: Point cloud library (pcl)," in *2011 IEEE International Conference on Robotics and Automation*, pp. 1–4, May 2011.

[33] M. Magnusson, A. Nuchter, C. Lorken, A. J. Lilienthal, and J. Hertzberg, "Evaluation of 3d registration reliability and speed - a comparison of icp and ndt," in *2009 IEEE International Conference on Robotics and Automation*, pp. 3907–3912, May 2009.

[34] E. Takeuchi and T. Tsubouchi, "A 3-d scan matching using improved 3-d normal distributions transform for mobile robotic mapping," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3068–3073, Oct 2006.

[35] M. Muja and D. G. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration," in *International Conference on Computer Vision Theory and Application VISSAPP'09)*, pp. 331–340, INSTICC Press, 2009.