3D Global Registration Using ICP & Branch-and-Bound Search

Rohith Krishnan Pillai
rohithkp@andrew.cmu.edu
Robotics Institute
Carnegie Mellon University

Oscar Dadfar
odadfar@andrew.cmu.edu
Computer Science Department
Carnegie Mellon University

Zihao Yang
zihaoyan@andrew.cmu.edu
Computer Science Department
Carnegie Mellon University

Abstract

Point cloud registration is a staple for many different applications in robotics and graphics communities. Iterative Closest Point (ICP), which is the standard method for this task suffers from local minimas and hence is only useful given a good initial alignment. We focus on the task of global registration of 2 point clouds, without the use of a priori information with regards to alignment, using a Branch and Bound (BnB) search over the euclidean transformation space with ICP sub-routines. We explore the idea that a BnB method can help in overcoming the local minimas that ICP would succumb to by comparing multiple methods using this framework.

The Iterative Point Cloud (ICP) Method is a ubiquitous algorithm for aligning two rigid-body point clouds subject to varying translations and orientations [1] . Widely used applications of the ICP algorithm include device localization in robotics and geometric reconstruction from surface scans. The ICP algorithm works by iteratively estimating the transformation and rotation matrix between point clouds based on a root mean squared distance between point correspondences. This affine transformation matrix is then applied to the source point cloud, bringing it closer to the target point cloud, and the process is repeated for a fixed number of iterations. What makes ICP robust is that it does not require a set of correspondences mapping source to target points, but rather computes them as the closest target point to each source point at each iteration.

However, ICP is susceptible to local minima. Because it operates on an iterative estimate and apply transformations scheme, point clouds can get stuck in a region where any small transformations would make alignment worse. Such effects occur when ICP computes the wrong correspondences and overfits points to a certain target region, prohibiting other points in the source point cloud to reach their target. Varying the initial orientation of the source point cloud can lead to ICP computing different transformations.This means that there is potential for a more optimal performing ICP algorithm that utilizes some outer-layer search routine in order to find the best initial conditions to perform ICP on and to gradually move away from local minimas towards a global minima.

1 Related Works

One of the most influential papers was Go-ICP [2], where researchers converted ICP from a local to global optimization search by using octrees to discretize the point cloud into 8 cube regions before optimizing over each region. They use a Branch-and-Bound search strategy to carry the best resulting transformation from each cubic region into the next iteration. Using Branch and Bound helps the algorithm reach a global optimum, although different partitioning strategies than cubic regions may prove to be more advantageous.

A robust point set registration method is introduced by [3], by modelling the discrete point clouds as continuous density functions using gaussian mixture models, where each point in the set is an uniformly weighted gaussian component with the spatial location analogous to a mean vector and common spherical covariance. The registration proceeds by iteratively reducing the L2 distance between the two GMMs using a quasi-newton algorithm, to update transformation parameter estimates. The use of GMMs allows for continuous optimizations, although it suffers in performance with highly differing sampling rates.

Other researchers in [4] formalized the relations between correspondence- based and some information-theoretic point cloud registration algorithms, and based on that algorithm, the authors proposed a new robust point set registration algorithm, and suggested a new point cloud Kernel Density Estimation approach which relies on maximizing the resulting distribution's entropy w.r.t. the kernel weights. The algorithm proposed in this paper is more on the theoretical side.

The authors of [5] investigated the rigid registration of large sets of points. They formulated this problem as a general Maximum-Likelihood (ML) estimation of the transformation and the matches, and showed that, in the specific case of a Gaussian noise, it corresponds to the ICP algorithm with the Mahalanobis distance. Finally, they obtained a slightly more complex criterion that can be efficiently solved using Expectation-Maximization (EM) principles. The EM principles proposed in this paper have great potential to be tuned or extended.

2 Methods

Our research attempts to survey multiple branch-and-bound techniques in order to help find a global optimum in ICP alignment. Our objective is not to directly complete or improve on the previously listed papers, but to experiment with alternative search strategies for finding a global alignment of two point clouds that could potentially have improved temporal or spatial performance.

We propose that finding different parameters to branch-and-bound on and changing the related policy update function can each have their own impact on accuracy and run-time performance. We denote the global objective function for two point-clouds A and B as the summation of the L2 norms between correspondences in A to correspondences in B. We try to minimize this global objective.

2.1 Index ICP BnB

Our first attempt uses a deterministic iterative ICP solver[1] (Index BnB ICP) that solves for translations by taking the difference of the centroids, and uses SVD on the dot product of the matrices in order to solve for rotation. The point clouds must contain the same number of points, so we trim the number of points from the point cloud with more data. For the larger point cloud, we use a mask denoting which points in the point cloud are used to compute the ICP transformation. After computing each transformation, we compute the nearest-neighbor distances for each point, and remove k points from the mask with the largest distance while adding in k points not in the mask with the smallest distance. We branch and bound on the number of points k replaced in each iteration, in order to move towards a global minimum.

2.2 Convex Hull ICP BnB

Our second attempt migrates over to an Open3D library based ICP solver[2] and uses the convex hull of the meshes as an initial alignment. This is then passed to the ICP BnB method from the first method, while at each iteration of the algorithm the convex hulls are used for the ICP alignment rather than the raw points. By doing so, we hope to reduce the weighting around densely packed regions of the point clouds and only consider the general shape from the convex hull when trying to align the point clouds.

2.3 PCA initialized - ICP Index BnB

Along the same reasoning on better initialization for the ICP, our third attempt uses principle component analysis (PCA) of the 2 point clouds to get the rough alignment. In practice, we need to consider the 4 different combinations for un-oriented PCA axes and find the rotation matrix with the smallest trace. While this is common, we implement in addition a corrected selection of the rotation matrix using the minimum source to target distance. These rough aligned point clouds are then passed

[1] https://github.com/ClayFlannigan/icp
[2] http://www.open3d.org/docs/release/

into our Index BnB based algorithm using an Open3D library based ICP solver for the estimate of the optimal alignment. However, it has to be noted that PCA comes with the significant disadvantage of not being able to find a good rough alignment when the point clouds are close to being symmetrical, and is very sensitive to noise unlike SVD decomposition. Figure 3 in the appendix shows the results for PCA rough alignment technique, and how our distance correction method improves the overall rough alignment even for difficult initial configurations. This method unlike ICP Index BnB also removes the assumption of the same number of points in both the source and target point clouds by always calculating the transformation from the less dense to the more dense point cloud and then providing the transformation or it's inverse, as appropriate.

2.4 PCA initialized - Randomized Weighted ICP BnB

Taking inspiration from the randomized weighted majority algorithm for average regret in Game Theory, the PCA initialized - ICP Index BnB method was modified to use some of these ideas. While the Index BnB algorithm chooses deterministically which points will be replaced in each iteration of the local BnB branch, this method uses a randomized weighted strategy instead to sample the points from the source point cloud. In particular, all the of the source point cloud weights are initialized to 1, and normalized to get a uniform probability of being selected at each iteration of the BnB. Then at each iteration of the ICP BnB, we randomly sample m points from source point clouds using the normalized weights as probabilities. These m points are used in the ICP, and all the points with a distance greater than δ are then penalized with their corresponding weights being reduced by an ϵ factor. Here, m, δ, ϵ are all hyperparameters, where we use $m = 0.7 * N_{src}, \delta = 0.5, \epsilon = 0.01$. Then the weights are again normalized before being saved into the entry added to the frontier of the BnB search.

2.5 Initial Rotation ICP BnB

Our fifth attempt uses branch and bound on the initial model rotation in order to combat ICP's susceptibility to local minima. By evaluating the model at various evenly-distributed initial rotations, we compute one iteration of ICP and store the resulting objective for each initial rotation in a heapdict, and continue evaluating the best-performing objective until the resulting objective is less than a given threshold ($t = 5.0$) and terminate, or the difference between the current and previous-step objectives are less than a given threshold ($\epsilon = 5.0$), to which we disregard this branch and attempt the next best initial rotation.

2.6 Inertia Sphere Method

Our last proposed method inspired from a simple physics observation: centre of mass and principal axis of moment of inertia are inherent features to localize an object. The principal axis of moment are equivalent to principal axis in principle component analysis (PCA), only with a difference in length. It is intuitive and effective to construct coordinate systems with centre of mass and principal axis of moment of inertia and calculate the transformation matrix between source and target coordinate systems. However, in practice, due to the noise and removal of certain points from the target point cloud, PCA method is unstable. This calls for a more robust version of PCA, or inertia matrix analysis.

Our proposed method is to calculate the moment of inertia along many axis, where the center of mass and a uniformly sampled point are on it, so that we can create a "Inertia Sphere"(See Figure 1). We then calculate the short and long axis of this quasi-ellipsoid. We first pick the point p that is on the shortest axis of Inertia Sphere, then n nearest neighbours are chosen, and we calculate the average of their moment of inertia as the average moment of inertia of point p $I_p = \frac{1}{n}\Sigma_{i=1}^{n} I_i$.

We compute the average moment of inertia of points $q \in Q$ that has a moment of inertia $I_q < I_p$. We pick the point r that has the least average moment of inertia as the point on the short axis, and follow the same strategy for picking the long axis. The final step is to calculate the transformation matrix between the source coordinate system established by the short and long axis along with the cross product of these two axis and the target coordinate system using ICP.

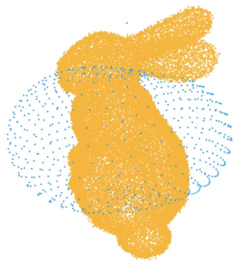

Figure 1: Inertia Sphere
The yellow point cloud is the source object, the blue quasi-ellipsoid is the Inertia Sphere. Each point on the Inertia Sphere is uniformly sampled with respect to solid angle, the distance between a point and the origin point (the centre of mass) is the normalized moment of inertia along the line of this point and the origin point.

Variance (σ)	0.00005	0.0005	0.005	0.05	0.1
Removal (p)	0.05	0.1	0.25	0.5	0.75
Method A.	**Baseline ICP (SVD)**				
Iters	1	1	1	1	1
Time(s)	1.24	0.78	0.92	1.46	1.27
Objective	3201.64	7351.40	4677.54	3476.76	2557.48
Method B.	**Baseline ICP (Open3D)**				
Iters	1	1	1	1	1
Time(s)	0.73	0.64	0.82	1.02	0.94
Objective	3289.82	4040.81	3698.09	2839.17	2143.46
Method C.	**Index ICP BnB**				
Iters	12	16	14	19	18
Time(s)	19.82	24.37	23.63	27.80	25.34
Objective	1526.38	2834.19	2394.32	2947.39	2329.03
Method D.	**Convex Hull ICP BnB**				
Iters	2	5	8	11	9
Time(s)	4.13	10.18	18.95	23.87	17.82
Objective	133.52	3561.97	3803.52	4322.26	15245.60
Method E.	**PCA Initialized - ICP Index BnB**				
Iters	20	20	20	20	20
Time(s)	13.16	12.44	14.03	17.18	8.51
Objective	3426.06	4293.58	3569.40	2486.55	1361.02
Method F.	**PCA Initialized - Randomized Weighted ICP BnB**				
Iters	20	20	20	20	20
Time(s)	33.45	32.82	27.87	19.76	10.3
Objective	3868.91	3659.86	3084.20	2109.18	**1123.76**
Method G.	**Initial Rotation ICP BnB**				
Branch	27	27	125	125	343
Iters	2	7	3	5	21
Time(s)	5.61	19.424	17.72	13.83	30.82
Objective	**481.77**	**68.99**	**309.51**	**740.69**	1137.65
Method H.	**Inertia Sphere ICP**				
Iters	1	1	1	1	1
Time(s)	1.51	1.38	1.29	1.15	1.01
Objective	723.99	236.81	502.80	747.20	1129.18

Table 1: Results for global registration methods, with the lowest objective for each column bolded.

3 Results & Discussion

3.1 Evaluation Protocol

For each algorithm, we compute a generated noisy variant of the Stanford bunny point cloud (approx. 35000 points) by adding zero-centered Gaussian noise with variance σ to each point while also removing points with probability p in order to test on sparse, less-stable point clouds. We then introduce a random rotational offset to the generated point-cloud and use our algorithm to align it. The objective that we attempt to minimize is the sum of the smallest L2 norms per point from the generated point cloud to base point cloud after the computed transformation T is applied.

3.2 Methods Discussion

Table 1A and 1B show the baseline SVD and Open3D ICP algorithms on one iteration. Since we are removing more points as the removal parameter p increases, then we will have less distances to sum over, giving us the perception that we were better able to lower the objective when it was due to a reduction of points.

Table 1C. shows the results of the Index Branch-and-Bound algorithm on the SVD-based ICP, including the total iterations until convergence and execution time. By using an adaptive step size when figuring out how many indices to swap, which was based on the change in the objective from the previous iteration to the current, we were able to further decrease the time taken to converge. Although we were able to reduce the objective lower than one iteration of ICP, changing the indices still caused the point clouds to reach a tighter local minima than before since we're still using the same ICP solver without changing the initial orientation scheme.

The Convex-Hull Branch-and-Bound method in Table 1D was able to converge on the initial generated model when $\sigma = 0.00005$ and $p = 0.05$, but succumbed to local alignment in future cases. This is due to the fact that a noisier generated model with fewer points will create a more misaligned convex hull that will be more difficult to align with the original data. Thus, for noisy data, the Convex Hull method is non-optimal, and shows some of the worse results of all the methods.

The PCA Initialized - ICP Index BnB method in Table 1E fares much better than most methods on sparse data, but has an objective score close to that of plain ICP for more dense point clouds. This can be attributed to the fact that the assumptions of the PCA being similar between the source and target point-clouds have not been met, and this leads to an rough alignment that does not necessarily improve the global alignment results.

While some of the limitations with the PCA based rough alignment moves on to the results of the Randomized Weighted ICP BnB method in Table 1F, it relatively outperforms the Baseline methods as well as the PCA Initialized - ICP Index BnB method due to the randomized sampling helping it overcome some of the local optimas. Although similar to results in Table 1E, the method does best with sparse data, these results are however not as good with more dense data.

In Table 1G, the Initial Rotation Branch-and-Bound method in Open3D features a Branch row that denotes how many different initial rotations we computed in order to arrive at the optimal solution. This is a cubed value since we test the same number of rotation intervals on each Euclidean axis. There is an evident trade-off between the number of initial branches and the number of iterations, where more time spent branching will lead to less time spent iterating trying to find the optimal branch. While the performance of Table 1G is worse with noisier data, Table 1H's Inertia Sphere ICP method is able to return similar results in a fixed time regardless of the noise of the mesh only after operating for one iteration. We provide visual comparisons of these two global solutions in Figure 2, in the Appendix section on the last page.

4 Conclusion

Global registration of meshes and point clouds are very important in many applications such as 3D reconstruction, but current methods are only able to handle relatively small displacements and initial transformations. In this paper we explored various methods that try to tackle global registration focused on the use of Branch-and-Bound search based techniques to escape local optimas. There quite a few improvements that can be made to the global registration using BnB that were not explored such as performance increases from parallelizing the search and more optimal objective functions that would weight outliers in noisy data less. The results show that BnB based approaches are a promising solution to global registration that could benefit from more research.

References

[1] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.

[2] Jiaolong Yang, Hongdong Li, Dylan Campbell, and Yunde Jia. Go-icp: A globally optimal solution to 3d icp point-set registration. *IEEE transactions on pattern analysis and machine intelligence*, 38(11):2241–2254, 2015.

[3] Bing Jian and Baba C Vemuri. Robust point set registration using gaussian mixture models. *IEEE transactions on pattern analysis and machine intelligence*, 33(8):1633–1645, 2010.

[4] Jeroen Hermans, Dirk Smeets, Dirk Vandermeulen, and Paul Suetens. Robust point set registration using em-icp with information-theoretically optimal outlier handling. In *CVPR 2011*, pages 2465–2472. IEEE, 2011.

[5] Sébastien Granger and Xavier Pennec. Multi-scale em-icp: A fast and robust approach for surface registration. In *European Conference on Computer Vision*, pages 418–432. Springer, 2002.

Appendix

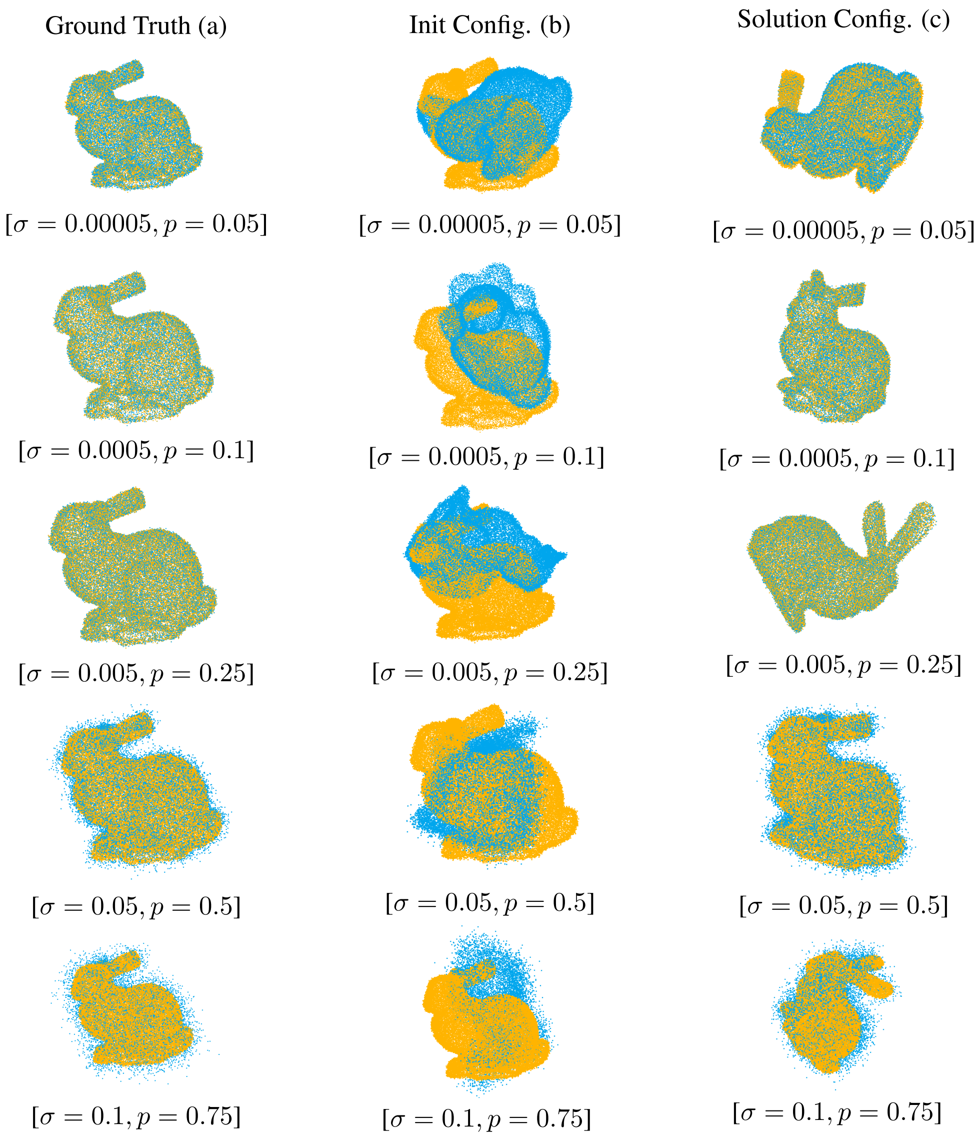

Figure 2: Results of the Initial Rotations BnB Method and Inertia Sphere Method in finding global optima. For noisier data, the former method requires more initial rotations to evaluate.

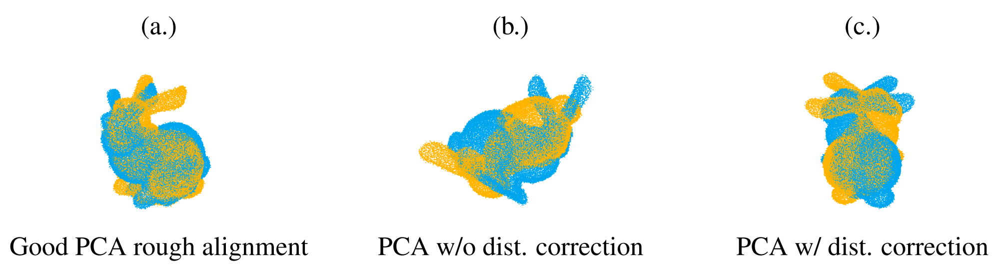

Figure 3: The PCA alignment examples with the best case alignment shown in (a.) not perfect due to noise, and for a more difficult initialization the PCA rough alignment without distance correction (b.) and the improved distance corrected PCA rough alignment with (c.)