Evaluating The Affine Grassmanian for First-Pass Histogram Place Recognition

Matthew Bradley University of Waterloo
John Zelek University of Waterloo
Email: {m7bradle,jzelek}@uwaterloo.ca

Abstract

Robotic applications like self-driving vehicles rely on Simultaneous Localization and Mapping (SLAM) for their position, which accumulates error over time. It is also requires initialization and periodically fails. This makes place recognition functionality vitally important, allowing recovery in these situations. LIDAR sensors have seen intense research for their immunity to lighting and generation of geometric data, while many recent description approaches have been based on graphs of semantically-relevant objects. This discretization into landmarks is less affected by viewpoint and occlusion, which can disrupt the distribution of single points and thus the effectiveness of global scan descriptors. One such method, Gosmatch, makes use of inter-object Euclidean distance in a series of histogram descriptors. While this approach works, it may be advantageous to incorporate more information into these descriptors. Affine Grassmannian distance, an approach combining relative position and orientation into a distance metric, is a promising approach to accomplish this. In this work we evaluate their suitability as a drop-in replacement for conventional Euclidean distances in the initial matching stage of Gosmatch's approach. As virtually all of Gosmatch's internal descriptors rely on distance histograms in some form, we believe this can provide an indication of the potential overall benefit affine Grassmanian distances offer.

1 Introduction

Navigational systems such as simultaneous localization and mapping (SLAM), form the core of many cutting-edge applications, including self-driving vehicles. These applications typically do not always have reliable access to GPS or other external means of positioning. Thus, loop closure is a critical subtask: they must be able to associate between subsequent visits to the same place so that drift which is accumulated in the intervening time can be eliminated. This also aids in recovery when the current position is not known or is lost due to disruption.

LIDAR sensors have received a great deal of research interest as they are immune to illumination problems that plague conventional vision systems, and also produce geometric information directly. While past LIDAR place recognition systems have relied upon point cloud statistical measures and local features, akin to camera-based bag-of-visual-word systems, recent work [1–3] has been oriented towards higher-level semantic and object-based place recognition. This work is predicated upon the idea that when humans memorize a place they do so based on the presence of high-level objects and semantics and their relationships, rather than by memorizing the minutiae of small local features and detail. [2]

One such method is Gosmatch [2], which makes use of a multi-stage pipeline for matching scans of different visits to the same place. An initial pass seeks to gather the top N most likely candidates through basic histogram descriptor matching, constructed from the physical distances between objects. This is then passed on to a system which attempts to use additional per-object distance histogram descriptors to match object between the likely candidates to determine a reranked top candidate before recovering relative transformation. The euclidean distances used to construct descriptors at all levels of Gosmatch only record distance to the center of an object however, which for some object types (ie. large planes) may not be accurately determined. It is expected be advantageous to be able to generate distance metrics which take into account relative orientation where possible.

For this purpose, the recently introduced affine Grassmannian manifold representation of 3D landmarks [4] provides a compelling mechanism for determining matching distances. It is a flexible and generic encapsulation for geometric objects of different dimension (points, lines, planes, etc) with a single method of computing position/rotation distance metrics between all of them. All objects are represented as geometric flats consisting of zero or more basis vectors. Points, lines, planes, and poses (0, 1, 2, and 3 basis vectors) are obvious examples. From these it is possible to construct representations for a variety of real world objects, with walls becoming planes and poles or tree trunks becoming vertical lines for example.

In this work we compare the relative performance of Euclidean distance as used by Gosmatch [2] against affine Grassmannian distances [4] for the purpose of generating first-pass distance histogram descriptors. As distances are used similarly in Gosmatch's cross-scan object association in its 2nd stage, the results are expected to also be applicable there. We utilize the same objects extracted from each scan and histogram description/matching techniques in both cases, ensuring a fair comparison.

2 Background Review

2.1 LIDAR Scan Matching For Place Rcognition

To eliminate drift in navigational systems using loop closure, or to otherwise recover one's position in a map, it is necessary to associate overlapping LIDAR scans captured at different times. Two general means of doing so are through descriptions of local keypoints and through global descriptors which look at the overall distribution of points around the LIDAR sensor.

Methods based on keypoints describe local regions in each scan so that their statistics can be compared between scans, for example [5] which proposed a voting system based on these small regions. Originally proposed for object recognition, this approach suffers from the highly variable density present in large-scale LIDAR scenes. Objects like trees with irregular shapes can also be problematic for extracting local features.[6]

More successful global descriptors capture the overall distribution of points in the scan as a whole. Scan Context [6] divides the local cylindrical region into rings of cells and describes points' maximum height in each one. DELIGHT [7] computes histograms on large sub-regions before performing a final spatial verification using keypoints between likely matches. LocNet [8] processes a polar representation of each whole LIDAR scan using a neural network.

However, these methods arguably do not capture how humans represent places, as regions containing interconnected landmarks. [2] Lidar place recogntion methods like SegMap[1], Gosmatch [2], and [3] match scans by segmenting point clouds into semantically-significant regions and constructing graphs between them. These representations encode not only the objects present but also the spatial relationships between them. Operating on semantically-significant regions, they are more robust to occlusions that can alter the observed distribution of individual points but may only remove a couple of high-level objects. [3]

In particular we consider Gosmatch here, which first attempts to select likely matches to the current scan using histograms of the distances between objects. The scans with histograms most similar to that of the current scene are passed on to a second stage which performs distance-histogram based matching to associate objects between scans. This allows the for refinement of the likely top candidate(s) and also enables alignment of the scans to be attempted.

2.2 Grassmannian Inter-Object Distance Metric

A recently developed use of affine Grassmannian manifolds [4] allows for the description of both the relative position and orientation of objects. With regular Grassmannian manifolds two objects expressed as a series of basis vectors can be compared based on the similarity of their orientations. [4] extends this concept such that relative position between objects in a LIDAR scan can also be considered, by augmenting the basis vectors describing each objects' orientation with an extra vector derived from their relative position. As Gosmatch [2] relies heavily on histograms of the euclidean distances between objects, here we examine affine Grassmannian distances as a possible substitute distance metric for improved descriptive power and performance.

3 Methods

In this work we compare the performance of Gosmatch's [2] first stage when using the original euclidean distances metric, and Grassmannian distances between objects to build histogram descriptors. To ensure a level playing field we use the same extracted objects in both cases, merely changing the distance metric as described below.

3.1 Data and Extraction of Objects

As input to the description system, we make use of a series of point cloud scans from the KITTI dataset [9]. These point clouds are processed to extract two kinds of objects in exactly the same way as [4], the initial demonstration of the affine Grassmannian. Not all scans are used, and instead are sampled at a spacing of 2 meters. They are taken from sequences containing loop closures, sequences 00, 02, 05, 08. What is yielded is a series of 2D planes and 1D lines, including object centroids and basis vectors (two vectors for planes, one for lines). Planes were extracted using the plane-finding algorithm provided by [4]. Lines are extracted from the SemanticKITTI dataset [10], derived from street poles.

3.2 Affine Grassmannian and Euclidean Distances

Euclidean distances as used in Gosmatch [2] are straightforward to collect, taking the distance between every pairing of reported object centroids in a scan. Affine Grassmanian distances however are more complex to determine.

For the affine Grassmannian the basis vectors of each object (and thus its orientation) are also collected, forming a Grassmannian matrix A which contains two columns for a plane and one column for a line/pole. The displacement from the origin of each object is also determined, yielding a vector b which represents the closest point on the object to the origin. This is the source of the affine Grassmannian's representation of an object's position in space.

When computing the matching distance between two objects, the distance between b vector points is used to augment one A matrix while the zero vector augments the other, adding an additional basis vector to the A matrices at comparison-time. There is some additional manipulation of the b vectors before matching occurs (to produce the analogous b0 vectors actually used for augmentation) with the details described by [4]. Multiplying the augmented A matrices and determining their principle angles using SVD allows for the final affine Grassmannian distance metric to be computed, incorporating both the objects' relative orientation and displacement from each other. This is repeated for every pairing of objects, as with the Euclidean distances.

3.3 Histogram Descriptors

Taking the distances between all planes with each other, all lines with each other, and all planes with all lines, three sets of distances are produced. These are used to construct three sub-histograms (60 bins), which are normalized and concatenated to form the final histogram description vector (180 bins) which is also normalized. This is the method employed by [2] to produce the scan descriptors for their first-stage matching. The difference here is that we perform this procedure with both the Euclidean distances (their technique) and affine Grassmanian distances. Both sets of resulting descriptors are independently tested for their matching performance. When matching, we take the L2 distance between every descriptor and every other, and rank the results. The top N matches for each scan among the other scans can then be taken, in this case the top N=100. In Gosmatch these would be passed on to inter-scan object association to further refine the choice of matching scans.

By way of example, if a scan contains planes A, B, and C and lines X, Y, and Z, then the three sub-histograms (which are concatenated to form its descriptor) each contain:

- Inter-object distances AB, BC, and AC. (between planes)
- Inter-object distances XY, YZ, and XZ. (between lines)
- Inter-object distances AX, AY, AZ, BX, BY, BZ, CX, CY, and CZ. (between planes and lines)

4 Preliminary Experimental Results

For each of the sequences considered, matching was performed as per the histogram-of-distance descriptors utilized by [2]. Three different measures of distance were used: Full affine Grassmannian distances, Euclidean distances between the b vectors used by the affine Grassmanian metric, and Euclidean distance between object centroids as in the original technique. Recall that the b vectors are the closest point on a given 2D plane or 2D line/pole that is closest to the origin. The rational given by [4] is that an infinite plane or line technically has no discernibly unique points, and so the closest points to the origin are taken and their displacement used. Measures are given for a few different bin sizes, with 60 histogram bins being found to be the best by [2] which is consistent with what is observed here.

When determining the ground truth for each frame, the set of all other scans in the sequence is taken which are located less than 10m away. Excluded from this are scans that occurred within 10 seconds, as they are likely from the same visit. A valid match for a given scan then, is any other scan which is a member of its set. It is possible when taking the top N scans which are most similar that multiple scans may be returned which can be found in the ground truth set for the scan under consideration. In the case of the full Gosmatch method [2], the returned list of scans from the initial histogram matching is later refined down to a final match.

The measures of performance given here are two-fold: The percentage of scans for which at least one correct ground truth match appeared in the top N (N=100) matching scans, and the average recall across all scans in the sequence. This average recall is the average of the percentage of ground truth matches for that scan which appear in the top N taken. If a scan has 20 other scans considered to be valid matches and 10 appear in the top N matching scans based on histogram matching, then that would be a recall of 50% for that scan in the sequence. The "top N matches" are merely the top N other scans which have histogram descriptors with minimum L2 distance (maximum similarity) to the histogram of the scan currently being considered. These are the output of the first stage of [2] which are provided for reranking via object-level association and geometric verification to determine the true final match result.

Observing the results, we can see the unexpected result that regular Euclidean distance between centers performs consistent best, despite the expectation that object centroids may not be reliably detected. The same Euclidean distance when applied to the points chosen for Grassmanian distance calculations performs similarly to these despite the Grassmanian taking into account orientation.

Table 1: For sequence 00, "at least one match found" and average recalls across three distance metrics.

00: At least one found	30 Bins	60 Bins	90 Bins
Grassmanian Distance	92.24%	91.72%	93.1%
Euclidean btw. b Vectors	91.9%	94.66%	94.14%
Euclidean btw. Centroids	**95.86%**	**97.76%**	**98.79%**
00: Average Recall	30 Bins	60 Bins	90 Bins
Grassmanian Distance	14.32%	14.45%	14.27%
Euclidean btw. b Vectors	12.9%	14.35%	14.52%
Euclidean btw. Centroids	**18.98%**	**21.87%**	**22.55%**

Table 2: For sequence 02, "at least one match found" and average recalls across three distance metrics.

02: At least one found	30 Bins	60 Bins	90 Bins
Grassmanian Distance	74.47%	70.87%	66.97%
Euclidean btw. b Vectors	78.98%	75.38%	75.98%
Euclidean btw. Centroids	**83.78%**	**83.18%**	**85.59%**
02: Average Recall	30 Bins	60 Bins	90 Bins
Grassmanian Distance	11.79%	10.47%	9.45%
Euclidean btw. b Vectors	12.4%	12.88%	13.2%
Euclidean btw. Centroids	**13.34%**	**13.73%**	**13.83%**

Table 3: For sequence 05, "at least one match found" and average recalls across three distance metrics.

05: At least one found	30 Bins	60 Bins	90 Bins
Grassmanian Distance	88.76%	89.6%	89.33%
Euclidean btw. b Vectors	89.33%	90.73%	89.6%
Euclidean btw. Centroids	**96.63%**	**96.07%**	**97.75%**
05: Average Recall	30 Bins	60 Bins	90 Bins
Grassmanian Distance	17.1%	17.26%	16.86%
Euclidean btw. b Vectors	17.43%	17.48%	17.75%
Euclidean btw. Centroids	**20.7%**	**21.96%**	**22.82%**

Table 4: For sequence 08, "at least one match found" and average recalls across three distance metrics.

08: At least one found	30 Bins	60 Bins	90 Bins
Grassmanian Distance	72.83%	72.08%	72.08%
Euclidean btw. b Vectors	79.62%	80.0%	78.11%
Euclidean btw. Centroids	**97.36%**	**97.74%**	**98.49%**
08: Average Recall	30 Bins	60 Bins	90 Bins
Grassmanian Distance	13.21%	12.07%	11.94%
Euclidean btw. b Vectors	11.97%	11.77%	10.97%
Euclidean btw. Centroids	**21.85%**	**24.34%**	**24.86%**

5 Discussion

5.1 Affine Grassmanian vs Euclidean Distances

Observing the results, the following is unexpected: regular Euclidean distances between centroids produce better matching than affine Grassmanian distances over all sequences. This suggests that the way that the position of each object is determined has a noticeable effect on the performance of the euclidean distance metric. Centroids from object detection seem to provide better performance than the b vectors chosen by the Affine Grassmanian, which assumes objects are infinite-extending planes or lines. A possible explanation is that these centroids are in fact fairly stable despite being drawn from clusters of points which may vary between observations. It is also possible that there is more noticeable variations in an object's b vector between scans, as the sensor origin moves and different points on the various objects become the closest to it.

When using the same points for position between the two methods (b vectors) we also observe that Euclidean distances and affine Grassmanian distances perform extremely similarly. This would suggest that in this application (histograms of distances) while the affine Grassmanian considers both orientation and position the relative position between objects has the stronger contribution. Alternatively, any contribution made by considering orientation is hard to notice when combined with possible poor performance from relative position when using b vectors.

In further experimentation, it would be instructive to modify the affine Grassmanian to make use of centroids instead of b vectors. This would essentially be to make the assumption that objects (planes, lines) are not of infinite extent and have a fixed position while still maintaining that they have orientations which can be compared. The result would be two-fold. It would first allow for a more complete comparison between euclidean distances and affine Grassmanian distances, when using the seemingly better method for determining position (centroids). It may also provde an improved platform upon which to evaluate the contribution of considering object orientation using the affine Grassmanian.

5.2 Overall Performance vs Gosmatch

While very little information is provided regarding the performance of Gosmatch [2]'s first stage of scan-level histogram descriptor matching, they do note in one instance that N=10, that they use 10 bins instead of the 100 used here to obtain measurable performance. The reason for this is believed to be the number of object categories considered. In the original Gosmatch paper [2], three categories of objects were used, and thus the concatenated histogram descriptor had six segments (three that were between a category and itself, and three containing distances between objects of different categories). Here we make use of the extracted objects used by [4] for evaluation of the affine Grassmannian distance metric, in which only two Grassmanian object categories are available and thus only three sub-histograms per descriptor are possible. This difference in the descriptiveness of histogram descriptors is believed to be responsible for the difference in performance. In future exploration it would be ideal to perform comparisons using datasets with more categories of extracted objects. This could be obtained by performing object extraction on the different categories of semantically labeled data in the SemanticKitti [10] LIDAR dataset, where many more classes are available.

6 Conclusions

We set out to evaluate affine Grassmanian distances as a viable enhancement to the histogram-based descriptors in [2], beginning with those used in its initial top-N ranking of match candidates. The belief was that introduction of rotational information into the distance metrics used would improve matching performance. Conversely it was observed to have had no or negative effect compared to euclidean distance. A likely contributing factor appears to be the affine Grassmanian's method of selecting points for each object's position in space. Alteration of the affine Grassmanian to use object centroids is expected to be a good alternative for future exploration. We also found indications of overall reduced performance compared to [2] but this is believed to be due to a reduction in available object categories and recommend investigating this using data with more object categories derived from SemanticKitti [10]. Overall, methods like [2] represent a promising approach to the problem of LIDAR place recognition, with robustness and viewpoint invariance through object-level landmarks, and we believe the application of the affine Grassmanian warrants further investigation. Work in this area is vital to the success of navigational systems based on LIDAR technology, and we hope to see systems like these continue to improve and support new applications.

Acknowledgments

We would like to thank the Canadian National Engineering Science and Research Council (NSERC) for supporting this research.

References

[1] R. Dube, A. Cramariuc, D. Dugas, H. Sommer, M. Dymczyk, J. Nieto, R. Siegwart, and C. Cadena, “Segmap: Segment-based mapping and localization using data-driven descriptors,” *The International Journal of Robotics Research*, vol. 39, no. 2-3, pp. 339–355, 2020.

[2] Y. Zhu, Y. Ma, L. Chen, C. Liu, M. Ye, and L. Li, “Gosmatch: Graph-of-semantics matching for detecting loop closures in 3d lidar data,” in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 5151–5157.

[3] X. Kong, X. Yang, G. Zhai, X. Zhao, X. Zeng, M. Wang, Y. Liu, W. Li, and F. Wen, “Semantic graph based place recognition for 3d point clouds,” in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 8216–8223.

[4] P. C. Lusk and J. P. How, “Global data association for slam with 3d grassmannian manifold objects,” *arXiv preprint arXiv:2205.08556*, 2022.

[5] M. Bosse and R. Zlot, “Place recognition using keypoint voting in large 3d lidar datasets,” in *2013 IEEE International Conference on Robotics and Automation*. IEEE, 2013, pp. 2677–2684.

[6] G. Kim and A. Kim, “Scan context: Egocentric spatial descriptor for place recognition within 3d point cloud map,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 4802–4809.

[7] K. P. Cop, P. V. Borges, and R. Dubé, “Delight: An efficient descriptor for global localisation using lidar intensities,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 3653–3660.

[8] H. Yin, L. Tang, X. Ding, Y. Wang, and R. Xiong, “Locnet: Global localization in 3d point clouds for mobile vehicles,” in *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2018, pp. 728–733.

[9] 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*. IEEE, 2012, pp. 3354–3361.

[10] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall, “SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences,” in *Proc. of the IEEE/CVF International Conf. on Computer Vision (ICCV)*, 2019.