Incremental Structured ICP Algorithm

Haokun Geng, Johnny Chien, Radu Nicolescu, and Reinhard Klette

The *.enpeda..* Project, Tamaki Campus
The University of Auckland, New Zealand

Abstract. Variants of the ICP algorithm are widely used in many vision-based applications, such as visual odometry, structure from motion, 3D reconstruction, or object segmentation. Establishing correct correspondences between two sets of 3D points, generated by stereo vision, needs to take those uncertainties into account. We propose a novel variant of the traditional ICP algorithm for solving the mentioned alignment problem. Our method, named *incremental structured iterative closest point*, aims at improved registrations of 3D points calculated for real world data. Evaluations are carried out by measuring the distances between two inputs, in both local and global perspectives, and experimental results are visually presented in this paper.

1 Introduction

The iterative closest point (ICP) algorithm is commonly used to determine a rigid transformation between two sets of points in Euclidean space by minimising distances or gaps between both sets. Generally speaking, the ICP algorithm takes two point clouds as its input, one point cloud is defined as being the *target*, which is kept at a fixed position in the world coordinate system. The other point cloud is defined as being the *source*. By minimising Euclidean distances (summarised in an error measure) between target and transformed source, the ICP algorithm iteratively aligns *source* to *target*. It should finally come close to an overall best transform (which can be decomposed into a rotation and a translation) of the source "towards" the target.

In this paper, we focus on solving the pairwise registration problem for two point clouds in 3D Euclidean space. Each point cloud represents a set of 3D surface points detected by using stereo vision when processing a sequence of stereo frames [9]. See Fig. 1 for two examples of input stereo pairs.

The intended application is to reconstruct roadsides from a mobile platform [16]. Both point clouds represent data obtained from Frame t and Frame $(t+1)$. The pairwise registration problem for such two frames of a stereo video sequences is considered as being a linear optimization problem. We assume that the environment is static, and only the mobile platform (with attached stereo cameras) is moving. In conclusion we can assume that there is always a rigid transform for aligning both point clouds.

We base our work on the common *point-to-point ICP approach*, because this methodology is appropriate for unorganised point data sets, and this variant can be easily improved and converted into the point-to-plane variant by using additional input information.

Fig. 1. Two subsequent stereo pairs (showing left image of each pair only) of a recorded stereo sequence with their corresponding point clouds.

This paper presents a novel variant of the common ICP method which we call the *incremental structured ICP*. Our proposed method uses a structured process to filter outliers by incrementally enlarging the coverage of the input point clouds, from one iteration step to the next step. Compared to other ICP methods, our method is able to reduce the impact of outliers and of missing data, as usual for disparity maps. Our method accurately establishes correspondences between the two input point clouds, with or without any pre-defined heuristics, such as defined by assigned weights to correspondences, or by a guess of an initial transform.

2 Literature Review

The original ICP algorithm has been firstly introduced by Besl and KcKay in [1] in 1992. A concept similar to ICP was also independently developed by Chen and Medioni in [3]. These authors used a different approach based on 3D surface registration, this work is usually recognised as point-to-plane variant of ICP. Here, the additional input of surface normals aims at an increase in robustness of the ICP algorithm, without reducing speed or increasing algorithmic simplicity.

Since the ICP algorithm proved to be an effective way of solving the registration problem, many variants of ICP have been proposed in the sequel. The problem of merging data sets from multi views is addressed in many publications. Johnson et al. [8] implemented a multi-view merging approach with not only the 3D information, but also with extra color properties. Pulli [11] also suggested the use of angles between normals for solving the pairwise registration problem of large data sets. Rusinkiewicz et al. [12] classified and summarised six typical steps that could improve the classic ICP, and those are (1) subset selection, (2) subset matching, (3) correspondences re-weighting, (4) outlier rejection, (5) error metric assignment, and (6) error metric minimising. Feldmar and Ayache [5] introduced a new ICP extension by using additional curvature properties and extended Kalman filters to improve the accuracy of the registration results in the correspondence re-weighting step.

Most of the variants of ICP seek to either improve their performance or to gain accuracy. But the most difficult task of ICP is how to reduce the impact of outliers present in the input datasets. The authors of [4] used a least-trimmed squares (LTS) approach to increase the robustness of their method. RANSAC-based methods (i.e., [6] are also a proven solution to reject outliers from the input. Sharp et al. in [15] suggested that the use of Euclidean invariant features can significantly reduce the chance of an ICP algorithm being trapped in a local optimal situation. Another attempt of a new ICP method is implemented by Segal et al. [14]; these authors combined the common ICP and its point-to-plane derivative into a probabilistic framework, in order to gain robustness without sacrificing speed or algorithmic simplicity.

3 Incremental Structured ICP

Our proposed method is called *incremental structured ICP* (IS-ICP). It applies a structured filtering procedure to reject outliers by incrementally enlarging the coverage of the input data sets. Similarly to the common ICP, our method takes two 3D scans (point clouds) as its input. We calculate an initial transformation $\mathbf{T}_{guess}$ as an optional input. The output is a refined transform constructed as a least-squares solution. It is common practice that the required alignment transform between both input point clouds is usually relatively small; thus it can be assumed that the input data define a linear optimization problem.

For consistency, we assume that both input point clouds $P_{tgt} = \{x_0, \ldots, x_n\}$ and $P_{src} = \{y_0, \ldots, y_n\}$ are of equal cardinality, with n being a known number of 3D points in the following explanations and calculations.

Initial Guess Transformation. The initial guess transformation of ICP is usually defined (in homogeneous coordinates) as a 4×4 identity matrix by default. In our proposal, we decide to produce a calculated initial transform as an optional input for the proposed IS-ICP algorithm. First of all, we need to find the 3D *centroid* (C_{tgt} and C_{src} of the two input cloud. Based on that, we form the covariance matrix (as described in [1]) of each point cloud as follows:

$$\mathbf{M} = \sum_{i=1}^{n} [(y_i - C_{src})^\top \cdot (x_i - C_{tgt})] \tag{1}$$

where $\mathbf{M}$ is a 3×3 matrix in our case; it can be decomposed using a singular value decomposition $(\mathbf{U}, \mathbf{S}, \mathbf{V})$. The rotation matrix is calculated as follows:

$$\mathbf{R} = \mathbf{V}\mathbf{U}^\top = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \tag{2}$$

The translation vector $\mathbf{t} = C_{tgt} - \mathbf{R} \cdot C_{src} = [t_x, t_y, t_z]^\top$ is obtained based on the rotation matrix. Finally, the initial transformation $\mathbf{T}_{guess}$, which maps P_{src} into the region occupied by P_{tgt}, can be composed as follows:

$$\mathbf{T}_{guess} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

This step is considered to be the initial step of our proposed IS-ICP method. The initial transformation should contribute positively to the final result when the majority of the input point clouds are inliers.

The Derived IS-ICP. The incremental structured ICP is derived from the original point-to-point ICP algorithm. It improves the original algorithm in two phases: (1) in the establishment of correspondences, and (2) in outlier rejection.

In the original ICP algorithm, once a correspondence is established, its weight value or its relation becomes unchangeable. Our proposed method tries to maintain the relations and weights of any established correspondences, in order to avoid infinite local optimization loops. We decide to use simple constant weights (just 0 or 1) to handle the weight assignments. The authors of [7] also suggested an effective method to assign weights.

In the rejection phase, as discussed in [11], there is at least 10% outliers rejected according to a pre-defined error metric. This metric is usually the point-to-point Euclidean distance between the two input point clouds, and it can be evaluated by the least square (error) minimising function of ICP. The authors of [10] also suggested that the used rejection threshold could be a positive real times the standard deviation of the Euclidean distances (e.g., threshold $2.5 \cdot \sigma$).

Random Subset Selection. Our method randomly picks a subset of points P'_{src} from the source point cloud. The size the subset must be greater than, or equal to 3, but not exceed 50% of the total number of the source point cloud. Then, we use a KD-tree to find the corresponding closest points in the target point cloud. By performing the error minimising function, a transformation matrix $\mathbf{T}'$ of the given subset should be produced at this stage.

If the difference between the subtransform matrix and the initial transform (as described in Eqs. (3)) is below a pre-defined threshold, we conclude that the subset is valid for further processing, otherwise the algorithm needs to randomly generate another subset again until it complies with this condition. Unselected points are put into a candidate pool for future considerations.

Incremental Recruitment and Rejection. After an initial subset is confirmed, the next step is the subset incremental rejection step. By selecting more points each time into the subset, the coverage of the input point clouds is incrementally enlarged in every iteration step. The algorithm will keep finding the optimal transform till all the points in the source point cloud are tested.

The algorithm considers the new subset transformation matrix only if the changes in rotation and translation are getting smaller (compared to the previous pair of subsets), otherwise the newly selected points will be rejected. The incremental structured ICP algorithm follows the rules as below:

$$|d|_{\triangle} = ||\mathbf{R}_0 \cdot \mathbf{t}_0 - \mathbf{R}_1 \cdot \mathbf{t}_1||^2 \tag{4}$$

where $\mathbf{R}_0$ and $\mathbf{R}_1$ are the rotation matrices, and $\mathbf{t}_0$ and $\mathbf{t}_1$ are the translation vectors. They are the components of the sub transformations, as mentioned in Equ (3). Value $|d|_{\triangle}$ represents the difference of Euclidean distances, which reflects the change in rotation and translation. Thus, the resulting transformation can be described as follows:

$$\mathbf{T}'_{n+1} = \begin{cases} \mathbf{T}'_{k-1} \cdot \mathbf{T}'_k & \text{if } |d|_{\triangle(k-1)} \geq |d|_{\triangle(k)} \\ \mathbf{T}_{k-1} & \text{if } |d|_{\triangle(k-1)} < |d|_{\triangle(k)} \end{cases} \tag{5}$$

where k is the index of the randomly selected subsets.

So, the overall transformation resulting from randomly selected subsets is as follows:

$$\mathbf{T}' = \mathbf{T}'_0 \cdot \mathbf{T}'_1 \cdot \mathbf{T}'_2 ... \cdot \mathbf{T}'_n \tag{6}$$

where n is the maximum number of loops for subset selections. Our algorithm does not reweight any correspondences or reject outliers in the initialising phase. The result $\mathbf{T}'$ is the best optimal transformation from the local subsets to align the source point cloud to the target point cloud.

From Local to Global Optimisation. Because our proposed method is only taking 3D coordinate information as its input, so it is most likely to be trapped by a local optimal (minimal) situation. To solve this problem, we need to provide a testing function to correct the error with the goal of global optimisation.

(1) Translation problem. When the input data has simple geometric layouts, the local optimal problem can be assumed to be a purely translational problem, thus it can be solved by using the centroid information:

$$\mathbf{t} = C_{tgt} - \mathbf{I}_3 \cdot C_{src} \tag{7}$$

This correction function is added after the stop criteria. If the Euclidean distance between the two data sets is getting smaller, it means that the correction should be kept, and another iteration of ICP needs to be performed.

(2) Rotation problem. When the geometric information is rich and complex, it could introduce a local optimal situation as the source point cloud is aligned based on a reflection-type rotation matrix. An example is shown in Fig. **??**, (a) and (b). Usually the numerics of the rotation matrix could be correct, but the rotation matrix wrongly transforms the source cloud along $x-$, $y-$, or z-axis for 180 degrees. Similarly to the solution of the translation problem, the rotation needs to be about the $x-$, $y-$, and $z-$axis. A correctly adjusted rotation should decrease the Euclidean distance between the two input point clouds.

4 Evaluation

We compare our proposed method with two other variants of ICP, the common ICP [1] and the reweighted ICP [2]. For experimental purposes, these two algorithms are implemented based on the basic "point-to-point" concept, and they take 3D point information as the only input, no additional data is provided for matching (i.e., RGB data, or normal values).

For the simulated bunny dataset (created and provided by [13]), all of the three algorithms are able to give correct matching results in most of the cases. In order to test the accuracy and robustness of the three methods, we decided to give an initial alignment (it could also be used as the ground truth) to the source cloud, with more than sixty degrees of rotation, and more than one meter of translation in all directions along the $x-$, $y-$, and $z-$axis.

Figure 2, top, shows the outputs of the three algorithms for the simulated data. The standard ICP gives a close matching, however it is trapped in a local optimal situation. The output of the reweighted ICP indicates its lack of robustness when there is a relatively large difference in rotation between the input

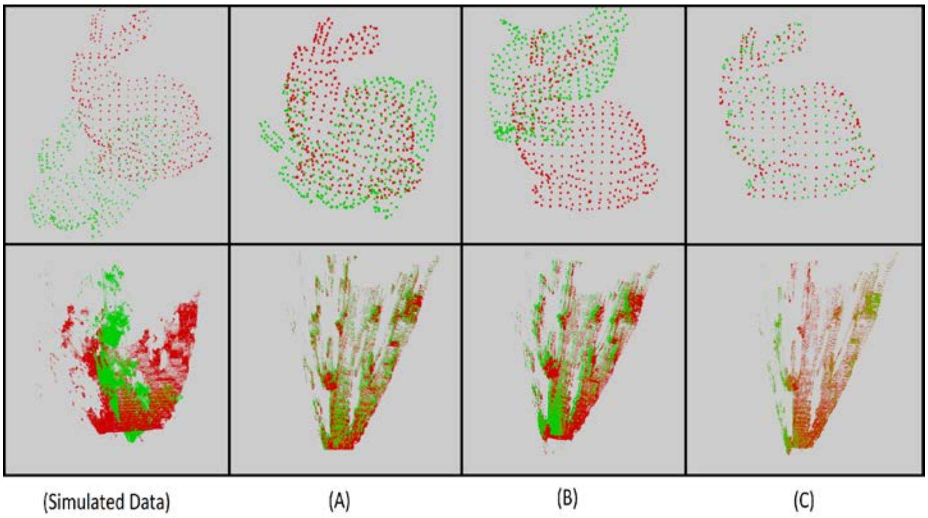

Fig. 2. *Top row:* Simulated data: A bunny with an initial alignment. *Bottom row:* Real world data with an initial alignment. Red shows the target point cloud, and green the source point cloud. *(A):* Standard ICP. *(B):* Reweighted-ICP. *(C):* IS-ICP.

target and source point cloud. This is because the weighted value of any given correspondence is unchangeable once the weight is decided by its re-weighting furcation.

Figure 2, bottom, shows the final (stable) alignments for the real world data (see Fig. 1) of the three methods. The outputs clearly show that the incremental structured ICP method has the best matching results among the three for a difficult initial alignment. The standard ICP method can generally produce good results, but it can not avoid to be trapped in a local optimal situation. Reweighted ICP sometimes assigns wrong values to the correspondences, so that the unchangeable weights can lead to a wrong matching result.

5 Conclusions

We proposed an ICP algorithm, focusing on improving the matching accuracy by incrementally enlarging the coverage of the input data. Existing ICP variants can be trapped in a local optimal situation (as shown in Fig. 2, top), so we took both a local and global optimisation approach into account. Our method keeps the similar basic structure of the standard "point-to-point" ICP, so the simplicity and performance of the method has been maintained in this way.

We compared our method with some other variants of ICP; the experiments show that our incremental structured ICP method brings improvements in both robustness and accuracy. When the input data sets are simple, our proposed method has a reduced chance to be trapped by a local optimal situation. When the input data sets are complex with outliers and missing data, our proposed method typically improves the matching results.

References

1. Besl, P., McKay, N. D.: A Method for Registration of 3-D Shapes. In: IEEE Trans. Pattern Analysis Machine Intelligence, vol. 14, pp. 239–256 (1992)
2. Chartrand, R., Yin, W.: Iteratively Reweighted Algorithm for Compressive Sensing. In: Proc. *ICASSP*, pp. 3869 – 3872 (2008)
3. Chen, Y., Medioni, G.: Object Modeling by Registration of Multiple Range Images. In: Proc. *IEEE Int. Conf. Robotics Automation*, vol. 3, pp. 2724–2729, (1991)
4. Chetverikov, D., Svirko, D., Stepanov, D., Krsek, P.: The Trimmed Iterative Closest Point Algorithm. In Proc. *ICPR*, vol. 3, pp. 545–548 (2002)
5. Feldmar, J., Ayache, N.: Rigid and Affine Registration of Smooth Surfaces Using Differential Properties. In: Proc. *ECCV*, LNCS, vol. 801, pp. 396–406 (1994)
6. Fontanelli, D., Ricciato, L., Soatto, S.: A Fast RANSAC-based Registration Algorithm for Accurate Localization in Unknown Environments Using LIDAR Measurements. In: Proc. *CASE*, pp. 597–602 (2007)
7. Godin, G., Rioux, M., Baribeau, R.: Three-dimensional Registration Using Range and Intensity Information. In: Proc. *Videometrics III*, SPIE, vol. 2350, pp. 279 (1994)
8. Johnson, A., Kang, S. B.: Registration and Integration of Textured 3-D Data. *Image Vision Computing*, vol. 17, pp. 135–147 (1999)
9. Klette, R.: *Concise Computer Vision: An Introduction into Theory and Algorithms*. Springer, London (2014)
10. Masuda, T., Sakaue, K., Yokoya, N.: Registration and Integration of Multiple Range Images for 3-D Model Construction. In: Proc. *ICPR*, vol. 1, pp. 879 (1996)
11. Pulli, K.: Multiview Registration for Large Data Sets. In: Proc. *Int. Conf. 3-D Digital Imaging Modeling*, pp. 160–168 (1999)
12. Rusinkiewicz, S., Levoy, M.: Efficient Variants of the ICP Algorithm. In: Proc. *3DIM*, pp. 145–152 (2001)
13. Rusu, R.B., Cousins, S.: 3D is Here: Point Cloud Library (PCL). In: Proc. *ICRA* pp. 1 – 4 (2011)
14. Segal, A., Haehnel, D., Thrun, S.: Generalized-ICP. In: Proc. *Robotics Science Systems* (2009)
15. Sharp, G., Lee, S.-W., Wehe, D.: ICP Registration Using Invariant Features. In: IEEE Trans. Pattern Analysis Machine Intelligence, vol. 24, pp. 90–102 (2002)
16. Zeng, Y., Klette, R.: Multi-run 3D Streetside Reconstruction From a Vehicle. In: Proc. *CAIP*, LNCS, vol. 8047, pp. 580–588 (2013)