

A ROBUST ESTIMATION TECHNIQUE FOR 3D POINT CLOUD REGISTRATION

DHANYA S PANKAJ AND RAMA RAO NIDAMANURI✉

Indian Institute of Space Science and Technology, Valiamala, Thiruvananthapuram, Kerala, India

e-mail: dhanyaspankaj@gmail.com, ramarao.iit@gmail.com

(Received July 30, 2015; revised November 5, 2015; accepted December 4, 2015)

ABSTRACT

The 3D modeling pipeline involves registration of partially overlapping 3D scans of an object. The automatic pairwise coarse alignment of partially overlapping 3D images is generally performed using 3D feature matching. The transformation estimation from matched features generally requires robust estimation due to the presence of outliers. RANSAC is a method of choice in problems where model estimation is to be done from data samples containing outliers. The number of RANSAC iterations depends on the number of data points and inliers to the model. Convergence of RANSAC can be very slow in the case of large number of outliers. This paper presents a novel algorithm for the 3D registration task which provides more accurate results in lesser computational time compared to RANSAC. The proposed algorithm is also compared against the existing modifications of RANSAC for 3D pairwise registration. The results indicate that the proposed algorithm tends to obtain the best 3D transformation matrix in lesser time compared to the other algorithms.

Keywords: 3D registration, robust estimation, RANSAC.

INTRODUCTION

A majority of the 3D vision applications require a complete or registered 3D image of the object under consideration. Due to the scanning limitations of the 3D acquisition devices and self occlusion caused by the geometry of the objects, the complete 3D scan of an object is usually acquired as multiple overlapping partial scans. The registration of these overlapping partial 3D scans is an important stage of 3D modeling pipeline (Gomes *et al.*, 2014). The partial scans are acquired by moving the object in front of the scanning device, or by moving the scanner around the object or by using multiple scanning devices. The partial scan data are represented in local co-ordinate system of the current scanner position. The goal of 3D registration is to bring all of the partial scan data into a common co-ordinate system. The registration of the partial scans into a complete single 3D scan is usually accomplished in multiple stages. Two of the common stages are 'pair-wise registration' and 'multi-view registration' (Dorai *et al.*, 1996). The survey in Tam *et al.* (2013) gives an excellent account of the various 3D registration algorithms.

The pairwise registration stage involves registration of two partially overlapping scans at a time. It aims to find a 3D rigid body transformation that transforms one of the partial scans into the co-ordinate system of the other. The multi-view registration aims at finding a set of rigid body transformations that will seamlessly register all the partial scans into a common

co-ordinate system. The information from the pairwise registration stage is used for guiding the global multi-view alignment stage (Pulli, 1999). We deal with the pair-wise registration problem in this paper. The pair-wise registration problem is generally solved in two stages – coarse registration or initial alignment and fine registration (Campbell and Flynn, 2001). Coarse registration stage involves finding out a rough initial alignment between the scans. Once an initial alignment is obtained, fine registration stage refines the alignment to further improve the same. The commonly used algorithm for fine registration is iterative closest point (ICP) and its variants (Besl and McKay, 1992; Chen and Medioni, 1992; Rusinkiewicz and Levoy, 2001). Most of these ICP based algorithms require a good initial alignment to obtain a good fine registration. The initial alignment can be obtained using different techniques. A controlled scanning environment may be used to measure the transformations, but it is usually expensive and limits the possible views. Another commonly used technique is to align the scans by manually selecting the corresponding points. The popular and automatic algorithm to align the scans is by feature matching. A set of corresponding points in the two scans is obtained by matching local 3D features.

In order to align the scan pairs, corresponding points in the two point clouds should be determined. The point clouds are preprocessed to remove noise, surface normals are estimated (Klasing *et al.*, 2009) and then the keypoints are extracted. A classification of the 3D keypoints in literature can

be found in Salti *et al.* (2011). Some commonly used 3D keypoint detectors are local surface patches (Chen and Bhanu, 2007), shape index (Koenderink and van Doorn, 1992), keypoint detector presented in intrinsic shape descriptors (ISS) (Zhong, 2009), keypoint quality index (Mian *et al.*, 2010), 3D SIFT, 3D SURF (Knopp *et al.*, 2010) *etc.* A recent evaluation of the 3D Keypoint detection algorithms is presented in Filipe and Alexandre (2013). This work (Filipe and Alexandre, 2013) indicates that ISS (Zhong, 2009) is an efficient keypoint detector in terms of repeatability and efficiency. 3D feature descriptors, which are invariant to rigid transformations, are extracted at the keypoints, based on a local neighborhood around them. Structural indexing (Stein and Medioni, 1992), point signatures (Chua and Jarvis, 1997), spin image (Johnson and Hebert, 1999), shape indexes (Koenderink and van Doorn, 1992), 3D shape context (Frome *et al.*, 2004), ISS (Zhong, 2009), PFH (Rusu *et al.*, 2008), FPFH (Rusu *et al.*, 2009), SHOT (Tombari *et al.*, 2010) *etc.*, are some of the widely cited 3D descriptors available in literature.

Once the features are extracted at the keypoints, they are matched across the two point clouds to find the corresponding points. The features are matched by constructing feature k - d trees (Friedman *et al.*, 1977) of the two point clouds and then finding the nearest neighbor of a source feature point in the target k - d tree. The k - d tree search is made efficient with the use of approximation algorithms like FLANN (Muja and Lowe, 2009). Correspondence rejection algorithms can be employed to reject the wrong correspondences obtained by this algorithm.

Once the corresponding points are obtained, least square algorithms are available in literature to calculate the 3D transformation (Arun *et al.*, 1987). However, points may get wrongly matched because of partial overlap, noise in the scan, local nature of features, inaccuracies in the previous stages *etc.* Thus the percentage of correct matches (inliers) in a set of correspondences can be very low. The performance of least square estimators deteriorates as the number of inliers to the model goes down. Robust estimators like least median of squares (Rousseeuw, 1984), least trimmed squares (Rousseeuw and Leroy, 2005) *etc.*, which can tolerate outliers are developed by the statistical community. Most of these estimators have a breakdown point up to 50% inliers. The higher the breakdown point, the more robust the estimator is (Wang, 2004). However as Wang (2004) points out, most of these estimators are not practical in the case of many computer vision applications as many problems require a breakdown point of more than 50%. To deal with such problems, many robust estimators like

RANSAC (Fischler and Bolles, 1981), Hough transform (Hough, 1962) *etc.* are developed within the computer vision community. In the case of 3D registration, the percentage of inliers depends upon many factors as mentioned above and cannot be guaranteed to be greater than 50%. Hence the 3D transformation model estimation using feature matching usually employs one of the robust estimators.

The number of RANSAC iterations depends on the number of correspondences, the complexity of the model and the percentage of inliers (Chum and Matas, 2005). If the percentage of inliers is very low, the convergence of RANSAC is generally slow. To deal with these issues, many variants to RANSAC like MLESAC (Torr and Zisserman, 2000), progressive sample consensus (PROSAC) (Chum and Matas, 2005), locally optimized RANSAC (LoSAC) (Chum *et al.*, 2003), NAPSAC (Myatt *et al.*, 2002) *etc.* have been proposed by the computer vision community (Choi *et al.*, 2009). Many of these techniques are applied to 2D vision problems like epipolar geometry estimation, motion segmentation, homography estimation, object recognition, image retrieval *etc.* However, adaptations of these methods to the 3D registration problem are not available in literature to the best of our knowledge.

In this paper we propose a novel robust estimation algorithm which provides accurate results for 3D registration in less computational time. A fast and accurate estimation of 3D transformation is crucial in applications where real time acquisition of 3D data and modeling are done. A good initial alignment phase helps in avoiding a large number of closely spaced scans and also reduces the time spent in the fine alignment stage.

Section *Methods* discusses the 3D registration problem and the proposed approach. *Results* section explains the experiments conducted for evaluating the algorithm and the results obtained. Section *Discussion* provides a brief discussion of the results and factors influencing the algorithm. *Conclusion* section concludes the paper.

METHODS

In the proposed approach, the 3D transformation is estimated from minimal samples, similar to RANSAC. In the case of 3D feature matching, the distance between the features is available as a rough estimate of the quality of the match, even though not completely reliable. This information can be efficiently utilized for guiding the sampling. Utilizing additional information is more efficient than the random sampling strategy of RANSAC as pointed out by Chum and Matas

(2005). Different guided sampling strategies have been proposed in literature. NAPSAC (Myatt *et al.*, 2002) is based on the assumption that the probability of finding an inlier adjacent to another inlier is high. This is based on the assumption that inliers of a model tends to lie closely. However this assumption is problem dependent and generally does not hold in the case of 3D transformation estimation where all the selected points confined to a particular area may lead to a local transformation model. In fact, steps are usually taken to avoid the sample points lying close together and they should be distributed across the shape for more accurate alignment.

Another variant of RANSAC which uses a guided sampling process is PROSAC (Chum and Matas, 2005). It utilizes the extra information, like the quality of the data points, usually available in most of the computer vision tasks. Instead of sampling randomly from the data point pairs, samples are selected randomly from a subset of the correspondence set sorted by quality of correspondences. The subset is formed initially from the top-ranked correspondences, grows gradually and eventually degenerates into the random sampling set, as the iteration progresses. Thus the sampling method finds balance between the random sampling and the sampling based purely on the quality of correspondence matches. As more promising samples are considered early, this can lead to significant computation time improvements to the algorithm. However, if the quality score is not very indicative of the correctness of the match, then the accuracy of the estimate may suffer. Having a lot of self similar points in the model can also lead to sub-optimal solutions as the feature match quality of these wrong matches may be high. Since only a subset of the whole set of corresponding points is considered for sampling where all combinations of matches are not explored, and since in some cases the matching score may not very reliable, this sampling can lead to sub-optimal solutions.

In order to address the sub-optimal solution obtained by selective sampling of subsets of correspondences, we propose to perform a refinement of the calculated transformation model. Once the transformation model is calculated from the minimal sample, the number of inliers to the model is found out. The corresponding points which fall within a distance threshold, after the transformation is applied, are treated as inliers. The threshold selection depends upon the magnitude of noise in the data. Then the transformation model which gives the maximum number of inliers is calculated. An all-inlier sample may not lead to a transformation model which finds all the inliers (Chum *et al.*, 2003). The solution

obtained usually lie near to the optimal solution and can be refined locally. The refinement is performed when the the current best solution is found in the iteration. Multiple loops of ‘model estimation – inlier calculation’ is executed by gradually varying the threshold from high to the normal value. The process of varying the threshold helps in considering points lying closer to the present inliers initially for model estimation. The solution is then refined as the threshold is decreased gradually. This process of local refinement typically leads to an optimal solution compared to the current best solution. However since the process is repeated every time the current best solution is obtained, this can result in more computational time. The computational burden caused by the local optimization is compensated by the computational savings obtained by guided sampling.

PROPOSED ALGORITHM

The proposed algorithm for transformation estimation is given in Algorithm 1.

The set of N correspondences obtained from 3D feature matching are provided as input to the algorithm. The correspondences are then sorted according to the quality of the matched features. A set of n corresponding points with highest quality is denoted as U_n . A sample consists of the minimum number of corresponding points ($m = 3$) required to estimate the 3D transformation. The samples are drawn from the (growing) subset U_n of the total correspondences. Quality of a sample is the quality of the corresponding point pair in it with minimum quality. Let T_N be the number of samples (sample size m) drawn by standard RANSAC from the set of N correspondences. Let T_n be the average number of samples in the original set of T_N samples, having data points only from U_n and this is given by Eq. 1:

$$T_n = T_N \frac{\binom{n}{m}}{\binom{N}{m}},$$

$$T_{n+1} = \frac{n+1}{n+1-m} T_n, \quad (1)$$

$$T'_{n+1} = T'_n + \lceil T_{n+1} - T_n \rceil.$$

A recurrence relation for finding T_{n+1} is given by Eq. 1. As the values of T_n are not integer in general, T'_{n+1} is given by Eq. 1 where $T'_n = 1$. The growth function for the sample generation set U_n is formed from T'_n using the Eq. 2. The growth function should reflect the result of previous tests and this is indicated by k , the number of tests conducted so far, since RANSAC runs are typically characterized by a success followed by a number of failures. Samples are drawn

randomly from the subset U_n which grows in size from $n = m$ to T_N according to the growth function:

$$g(k) = \min \{n : T'_n \geq k\} . \quad (2)$$

Once the minimum sample is obtained as described, a 3D transformation model is computed from it using least square technique (Arun et al., 1987; Umeyama, 1991). Given a set of point pairs $\{x_i, y_i\}$, $i = 1 : N$, a rigid body transformation, which minimizes the accumulated distance between the corresponding points after applying the transformation, is calculated. This distance is given by Eq. 3 where R and t represents the 3D rotation matrix and translation vector respectively, x_i and y_i represents the 3D co-ordinate vectors of the point pair i :

$$E^2 = \frac{1}{N} \sum_{i=1}^N \|y_i - (Rx_i + t)\|^2 , \quad (3)$$

$$e^2 = \|y_i - (Rx_i + t)\|^2 . \quad (4)$$

The 3D transformation is applied to the source point in each of the corresponding point pairs and the distance between the target point and the transformed source point is computed for each of the point pairs. This distance for a point pair i is given by Eq. 4. If this error falls within a distance threshold, then the point is considered as an inlier to the model calculated. The threshold is calculated empirically and is based on the precision or inter-point distance of the point cloud. The set of all such inliers is formed. The sampling and model generation phase is repeated until a local best solution or the current best number of inliers is obtained.

Once a local best solution is obtained, a refinement of the solution is carried out. A hybrid iterative and inner RANSAC algorithm (Chum *et al.*, 2003) is implemented for model refinement. Once the current best solution is obtained, a sample is selected randomly from the set of inliers. This sample size can be greater than the minimal size as we are sampling from the set of inliers. In our experiments it is taken to be the maximum of 3 and half the number of inliers. Then the transformation model computed from this sample is further refined iteratively by varying the threshold for finding the inliers and estimating transformation from the inliers. This is repeated k_{LO} times and the best number of inliers is updated depending on the result.

The proposed solution takes care of the extra computational burden caused by local optimization steps by utilizing the time savings obtained by guided sampling. Also the sub-optimal solution attributed to the limited number of samples due to guided sampling

is taken care of by the local optimization where samples are generated from the set of current best inliers. Thus samples other than those from the sample generation set in guided sampling are also considered.

The stopping criterion of the loop is calculated as follows. Let m be the size of the minimum sample, and ϵ be the percentage of inliers. In k iterations we will be drawing k samples. If we are drawing k samples, then the probability of not finding an all-inlier sample in k samples is $(1 - \epsilon^m)^k$. k is selected such that this probability falls below η_0 . This is given by Eq. 5 where P_{I_n} represents the probability of obtaining an inlier which is approximated as the percentage of inliers, ϵ :

$$k \geq \log(\eta_0) / \log(1 - P_{I_n}) . \quad (5)$$

In the proposed algorithm, the samples are generated from a small subset of correspondences. Hence an additional check is also employed to ensure that the model obtained is not randomly endorsed by a set of outliers (Chum and Matas, 2005). Assuming binomial distribution for the cardinality of the set of random ‘inliers’, the probability of obtaining a random inlier set with size i is given by Eq. 6:

$$P_n^R(i) = \beta^{i-m} (1 - \beta)^{n-i+m} \binom{n-m}{i-m} . \quad (6)$$

For each sample generation set with size n , the minimum number of inliers I_n^{min} is calculated so that the probability of obtaining a random support of that size falls below a threshold, ψ . This is given by Eq. 7.

$$I_n^{min} = \min \{j : \sum_{i=j}^n P_n^R(i) < \psi\} . \quad (7)$$

Then, only a model which provides the number of inliers I_n greater than this minimum support size is selected as a solution, given by Eq. 8.

$$I_n > I_n^{min} . \quad (8)$$

The iterations are repeated until the two conditions given by Eqs. 5, 8 are satisfied.

RESULTS

The partially overlapping point clouds of various objects acquired for 3D modeling task were used for testing. Scans from two different datasets were considered for the study. Scans of three models from the Stanford dataset (Levoy, 2005), acquired using Cyberware 3030 MS scanner – Buddha, Bunny and Dragon models – were registered. The scans of Chef, Chicken and Parasaurolophus models, acquired

Algorithm 1 Proposed Algorithm

```
{%comment  $N$  : Number of correspondences
 $m$  : Minimum Sample Size( 3 for 3D registration )
 $\eta_0$  : Threshold for stopping criterion % comment }
```

```
 $n^* \leftarrow N, n \leftarrow m, k \leftarrow 0, I^* \leftarrow 0, k^* \leftarrow 1 T'_n \leftarrow 1,$ 
 $T'_N \leftarrow 200000,$ 
 $k < k^*$ 
```

1. Select model generation set

```
 $k \leftarrow k + 1$   $k = T'_n$  and  $n < n^*$ 
```

```
 $n \leftarrow n + 1$ 
```

```
where  $T_n, T_{n+1}$  and  $T'_{n+1}$  are given by Eq. 1
```

```
Sample selection of size  $m$   $T'_n < k$ 
```

```
Sample contains  $m - 1$  points at random from  $U_{n-1}$  at random and  $u_n$ 
```

```
Select  $m$  points from  $U_n$  at random
```

2. Model Generation

```
Generate model or estimate 3D transformation from the sample
```

3. Model Verification

```
Find inliers  $I_k$   $I_k > I^*$ 
```

```
A. Run Local Optimization Step
```

```
 $m_{min} \leftarrow \min(I_k/2, m)$ 
```

```
 $k_{LO} \leftarrow 10$ 
```

```
 $K \leftarrow 5$ 
```

```
a. Select  $m_{min}$  samples from the inliers  $I_k$ 
```

```
b. Generate model from the samples
```

c. Iterative

```
Find inliers with distance  $< K\theta$ 
```

```
Generate model from these inliers using linear algorithm
```

```
Reduce threshold and repeat step Iterative until threshold =  $\theta$ 
```

```
d. Repeat steps a to c  $k_{LO}$  times and store the best model and best  $I^*$ 
```

```
B. Select termination length  $n^*$  and  $k^*$  such that equations
```

```
 $k_{n^*} \geq \log(\eta_0) / \log(1 - \epsilon_{n^*}^m)$ 
```

```
and
```

```
 $I_n^* \geq I_{n^*}^{min}$ 
```

```
are satisfied.
```

```
C.  $I^* \leftarrow I_k$ 
```

```
D.  $k \leftarrow k + 1$ 
```

by Minolta scanner from the University of Western Australia dataset (Mian *et al.*, 2006) were also tested. Actual transformations are available for the Stanford models and hence this information was used for evaluating the performance of the algorithms. In the case of the UWA dataset models, the actual transformations are not available and hence the distance between the corresponding point pairs after registration was considered as an error measure for evaluating the performance of the algorithms. The .ply files were converted to .pcd files before processing. The .pcd file contains the 3D co-ordinates information. We have not considered intensity or color information for this study.

In order to evaluate the performance of the proposed algorithm, 3D pairwise registration of partially overlapping point clouds was implemented using the three existing algorithms RANSAC, the variants of Ransac – PROSAC (Chum and Matas, 2005) and LoSAC (Chum *et al.*, 2003) – in addition to the proposed algorithm.

One of the 3D images (source) was registered to the other (target) by estimating the 3D rigid body transformation. The 3D image pairs were down-sampled using voxel-grid filtering and processed using statistical filtering. The normals at data points were computed using PCA and ISS keypoints were extracted. 3D shape context descriptors were extracted

at the keypoints in both the point clouds. The correspondences of the keypoints in the source point cloud were found in the target point cloud by matching the features. The features were matched by using the nearest neighbor search in feature $k-d$ trees. The best correspondences were found using the compared algorithms where the 3D transformation estimation was obtained by SVD based least squares method (Arun *et al.*, 1987; Umeyama, 1991). A minimum distance constraint was applied while sampling the points which ensure that the sampled points are distributed across the point cloud and are not crowded together and leads to better transformation estimation.

The point clouds were processed and the set of corresponding point pairs were obtained along with the distance between the corresponding feature vectors. These corresponding points were given as input to all the four algorithms and the resulting transformation results were compared. In the case of Stanford models, the actual transformation in terms of 3D translation units and the 3D rotation represented as unit quaternions, with respect to the common co-ordinate system is available. This information was used to calculate the transformation between the pairs of point clouds considered for pairwise registration. After registration, the obtained 3D transformation matrix was converted into x, y, z translation units and the rotation unit quaternion. The norm of the distance

between the translation units and the dot product between the rotation unit quaternions were computed. Lesser the norm of the translation difference, better the computation of 3D translation. The rotations are compared by taking the dot product of the estimated and actual quaternions. Since the quaternions are normalized, the dot product of value of 1 indicates that the 3D rotations are approximately equal. The dot product value indicates how close the estimated rotation is to the actual rotation. Higher the dot product value, better the estimated rotation. The time taken by the algorithms in the main loop on an Intel I3 3.10 GHz processor was also computed and is reported in milliseconds.

For the UWA datasets, the actual transformations are not available. Hence for evaluation, instead of translational distance norm and dot product between quaternions, an error measure called transformation validation score was used. This score is a measure of the distance between the corresponding points of the pair of scans after transformation is applied. To account for the points with no actual corresponding points, a distance threshold was applied so that the points greater than a distance apart are filtered from the calculation. The score is normalized with respect to the number of correspondences found. The lower the transformation validation score, the better is the alignment and the registered clouds were additionally verified manually for ensuring this. The number of inliers to the computed transformation model was also computed. The number of hypotheses or models evaluated was also computed and this is an indication of the number of samples tested before finding the final transformation. The experiments were repeated 50 times and the average values are reported. The algorithm was implemented in C++ with the help of PCL library (Rusu and Cousins, 2011).

The partially overlapping scans of the Stanford model Buddha were registered and the various evaluation metrics were compared across the different algorithms. The x -axis indicates the different scan pairs in all the figures. The dot product of the actual rotation unit quaternion and the one obtained from the registration was computed and plotted for different set of scan-pairs, for the different algorithms and is given in Fig. 1. From Fig. 1, we can see that the proposed algorithm and LoSAC obtain consistently good rotation estimates across the different scan pairs. PROSAC incurs more rotation errors and the RANSAC results are not consistently good. The result shows that the proposed algorithm obtains good and consistent results for rotation estimation.

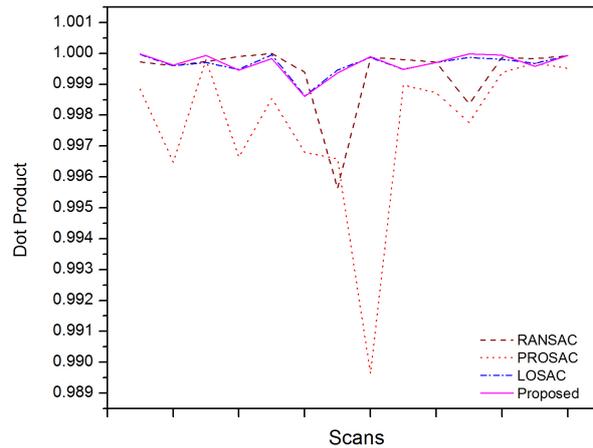


Fig. 1. Buddha dataset – dot product between actual and estimated rotation quaternions.

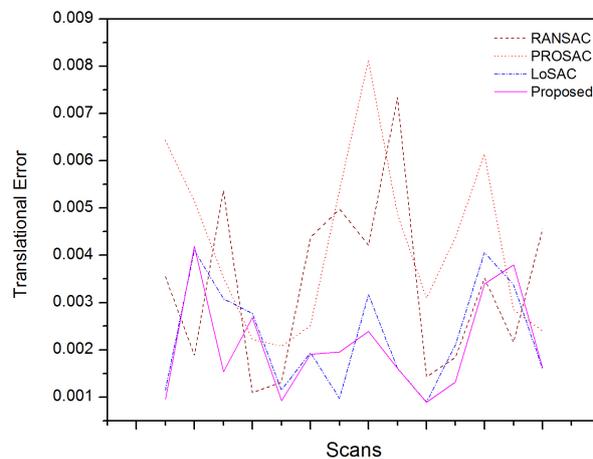


Fig. 2. Buddha dataset – error in translation estimate.

The error in computation of 3D translation for the same set of scans is plotted in Fig. 2. Fig. 2 indicates that the proposed algorithm achieves comparatively lower error in the estimation of 3D translation. RANSAC and PROSAC results are much worse. LoSAC and the proposed algorithm achieves comparable results.

The time taken in the main loop is also plotted for different algorithms in Fig. 3. From Fig. 3, it can be clearly seen that the time taken by the proposed algorithm is much lower compared to LoSAC or RANSAC for all the scans. Only PROSAC runs faster than the proposed algorithm and it should be noted that the rotational and translational accuracy of the PROSAC is much less compared to the proposed algorithm.

Another set of experiments were conducted using the overlapping scans of the Stanford Bunny dataset and the results are given in Figs. 4–6. The percentage of inliers in the Bunny dataset is much less as compared to the Buddha dataset. However, the trend in the results is similar to that of the Buddha dataset. The proposed algorithm attains good 3D transformation estimates compared to the other algorithms in lesser time. In this dataset, the time taken by the proposed algorithm is even lesser than that of PROSAC. This may be attributed to the lower percentage of inliers in this dataset. When the percentage of inliers is very low, the proposed algorithm even outperforms PROSAC in execution time. It shows better results than RANSAC and PROSAC in terms of accuracy.

The same set of experiments was also conducted using the scans of the Dragon dataset. The results are given in Figs. 7–9. The inlier percentage of correspondences in this dataset is also low for the selected feature. The results are similar to that of the Bunny dataset. Here also the proposed algorithm outperforms the PROSAC and RANSAC algorithms in terms of accuracy of the estimates. It is comparable to LoSAC in the case of accuracy but performs better in terms of execution time. The execution time of the proposed algorithm is also comparable to that of PROSAC and even outperforms it in some instances.

Scans of 3D models from the UWA Dataset are also used for evaluating the proposed algorithm. True transformations are not available and hence the transformation validation score is reported. The transformation validation score indicates the average error per corresponding pair. The unit of this score is in accordance with the unit of the real world coordinates of the point cloud used. Fig. 10 shows the transformation validation scores of the different algorithms on the Chef dataset scan pairs. Since this is not mentioned in the dataset, we have not indicated it in our results. From the figure, we can see that the proposed algorithm obtains the minimum error compared to other algorithms. The time taken by the proposed algorithm is much less than LoSAC and RANSAC and slightly more than PROSAC, in most cases, as in Fig. 11. For one instance for the scan pair 5, RANSAC is faster than the proposed algorithm, but the estimation error for RANSAC is higher.

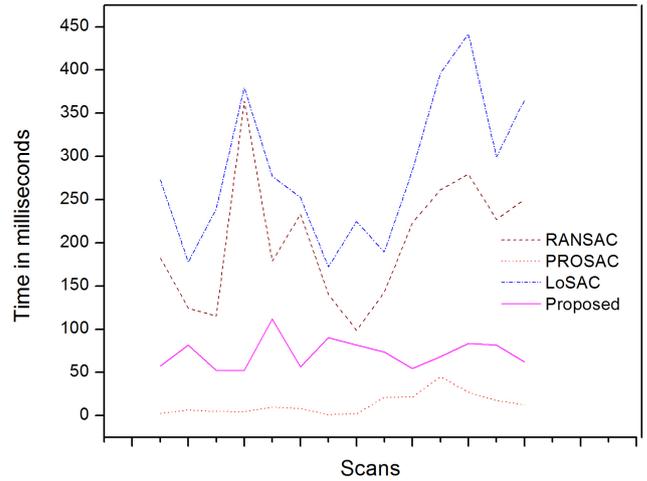


Fig. 3. *Buddha dataset – time taken for RANSAC iterations in milliseconds.*

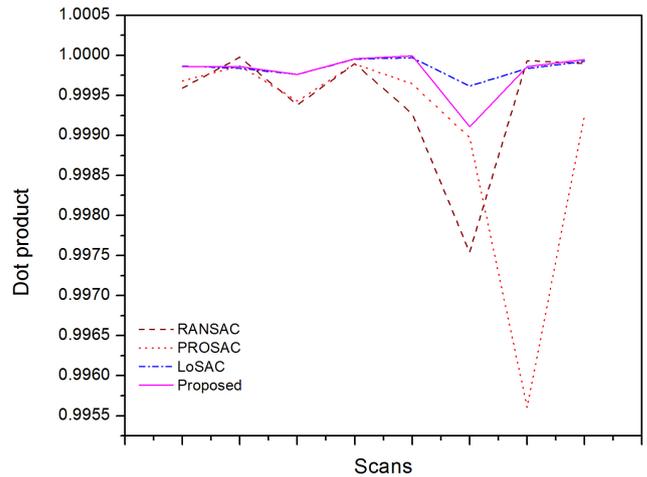


Fig. 4. *Bunny dataset – dot product between actual and computed rotation quaternions.*

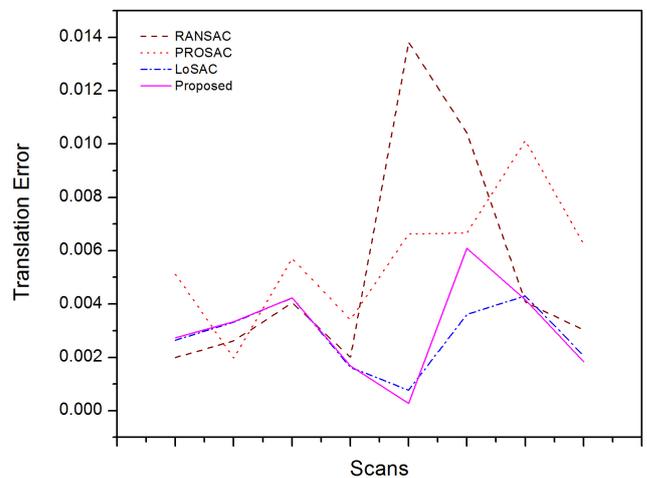


Fig. 5. *Bunny dataset – error in translation estimate.*

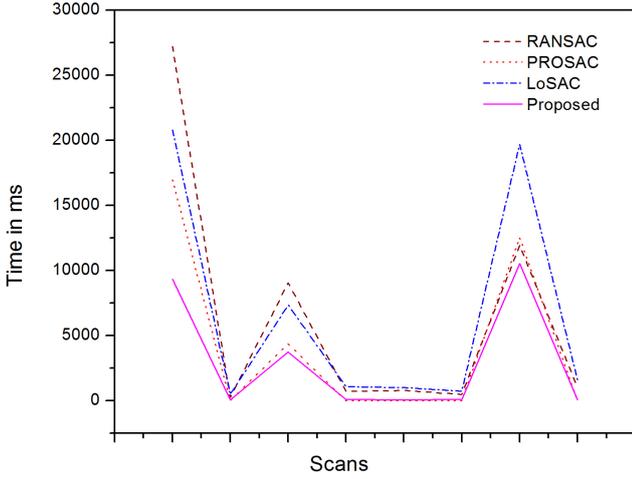


Fig. 6. Bunny dataset – time taken for RANSAC iterations in milliseconds.

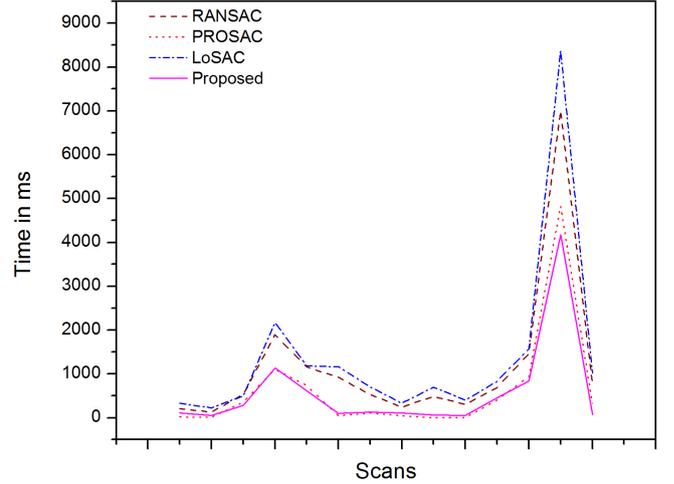


Fig. 9. Dragon dataset – time taken for RANSAC iterations in milliseconds.

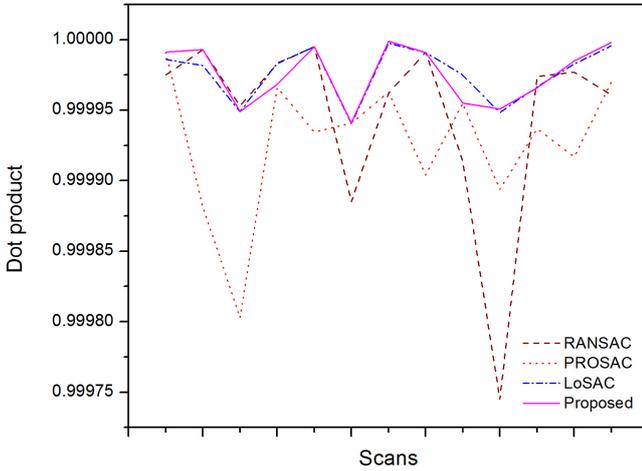


Fig. 7. Dragon dataset – dot product between actual and computed rotation quaternions.

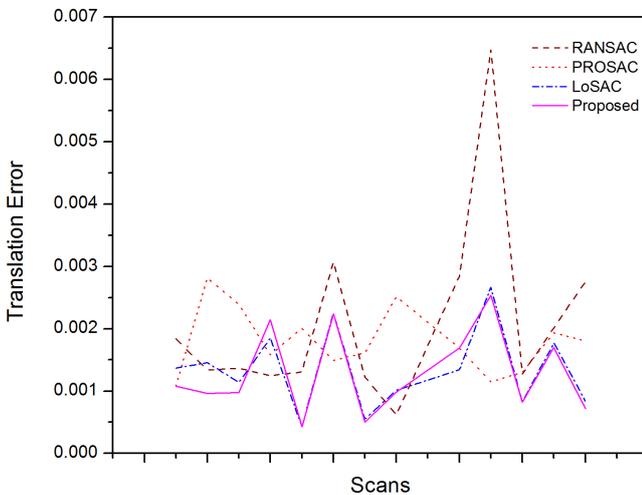


Fig. 8. Dragon dataset – error in translation estimate.

The number of inliers found out by the proposed algorithm is the highest and is comparable to that of LoSAC, as given by Fig. 12. The number of model hypotheses tested by the algorithms is also reported in Fig. 13 and the results indicate that this is lowest for the proposed algorithm. The models tested inside the local optimization stage are not included. However, the numbers of models tested by LoSAC, without including the inner models are also much higher than that the proposed algorithm.

Similar experiments were conducted on the scan pairs from two other models of the UWA dataset – Chicken and Parasaurolophus models. The results obtained are given in Figs. 14–21. The results on these datasets also support the reasoning in the previous cases. There is a case in the Chicken dataset where the accuracy of the proposed approach is lower than that of other algorithms and similar to that of PROSAC. We attribute this to the lower percentage of inliers for the scan pair and the small subset of correspondences selected by guided sampling. In other cases, the proposed algorithm performs consistently well.

The results of registration of some of the scan pairs are shown in Figs. 22–23. The results obtained by RANSAC and the proposed algorithm are shown for comparison. The figures show that the registration obtained by the proposed algorithm is very accurate compared to RANSAC and its variants. This helps in reducing the time taken by the fine registration stage to achieve convergence. Accurate results help the subsequent stages to converge to a precise solution.

The unregistered point clouds and the clouds after initial registration by the proposed method are shown in Figs. 24–29.

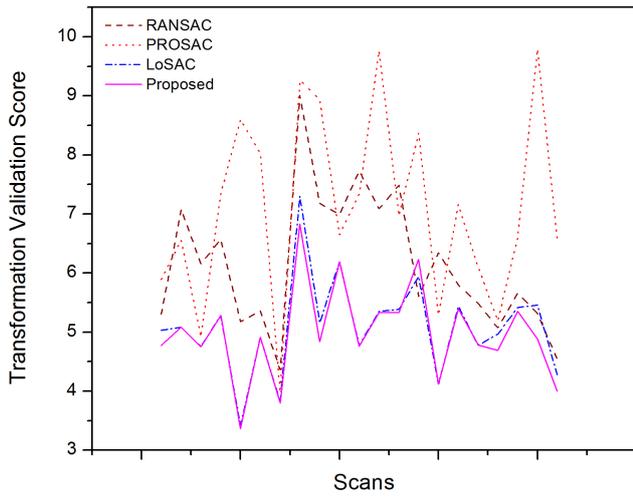


Fig. 10. *Chef dataset – transformation validation score.*

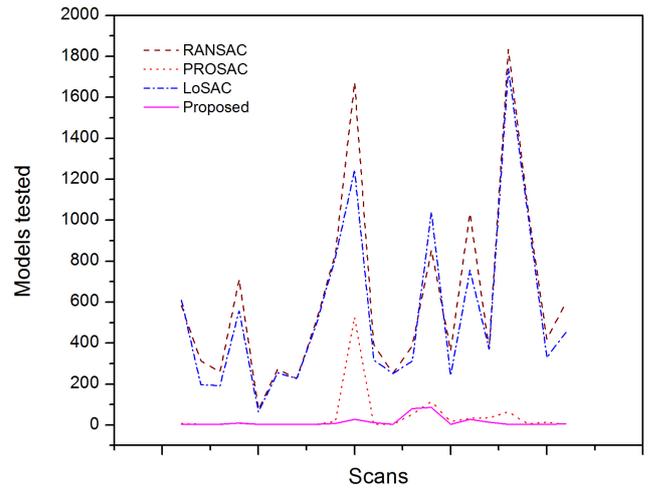


Fig. 13. *Chef dataset – number of models tested.*

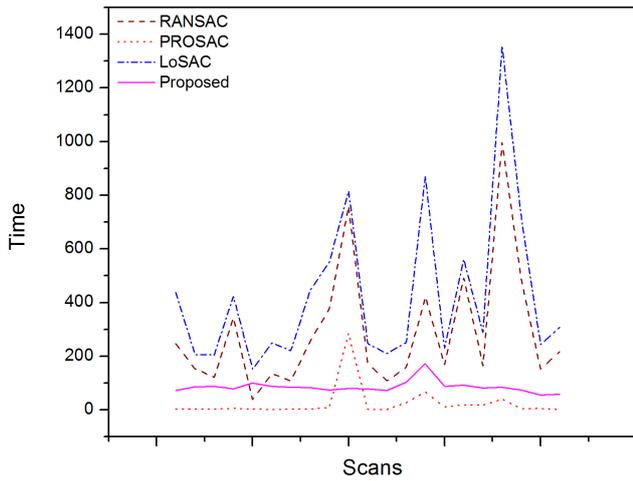


Fig. 11. *Chef dataset – time taken for RANSAC iterations in milliseconds.*

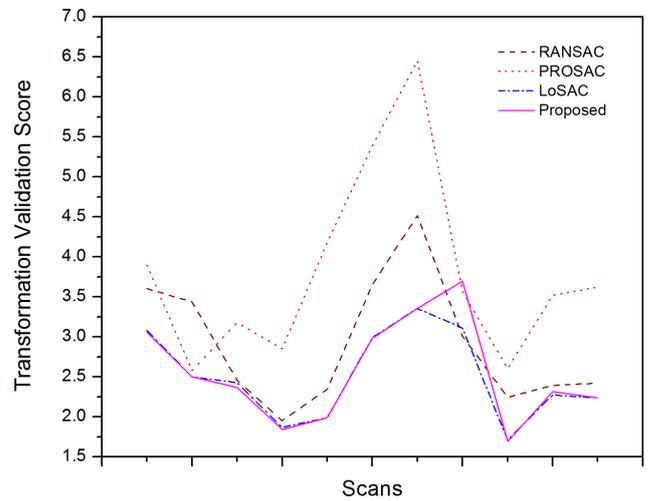


Fig. 14. *Chicken dataset – transformation validation score.*

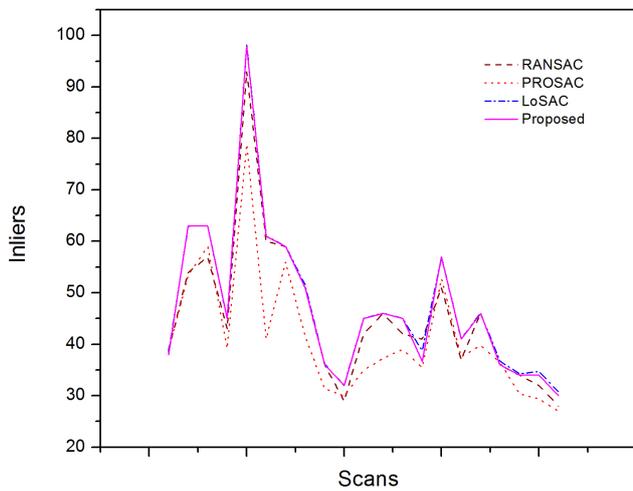


Fig. 12. *Chef dataset – number of inliers.*

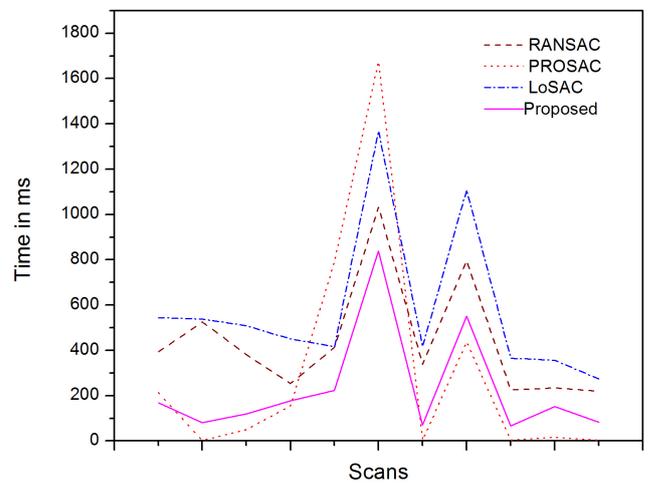


Fig. 15. *Chicken dataset – time taken in milliseconds.*

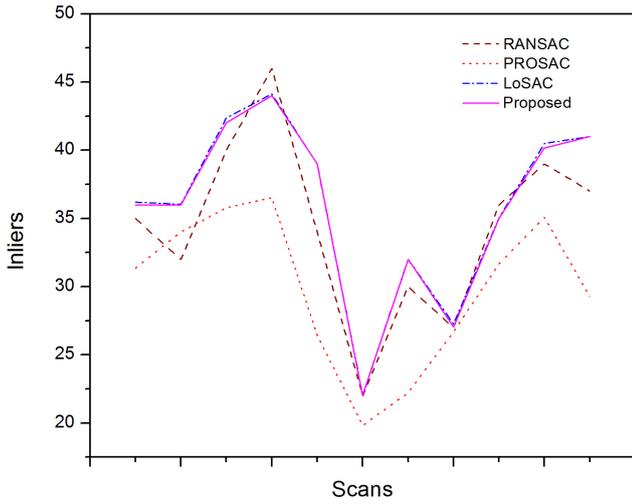


Fig. 16. *Chicken dataset – number of inliers.*

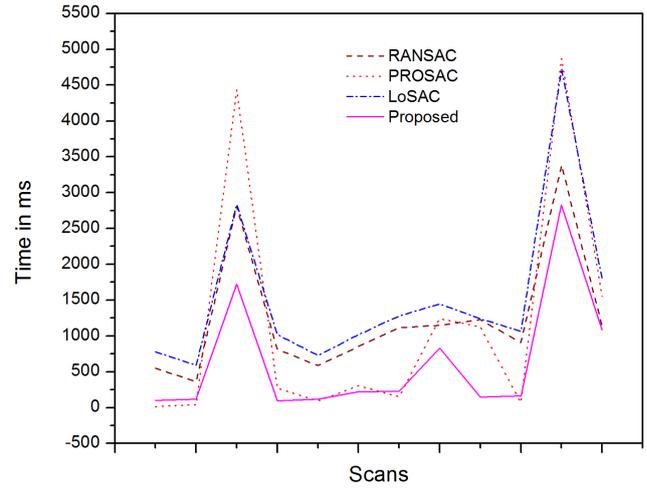


Fig. 19. *Parasaurorolophus dataset – time taken in milliseconds.*

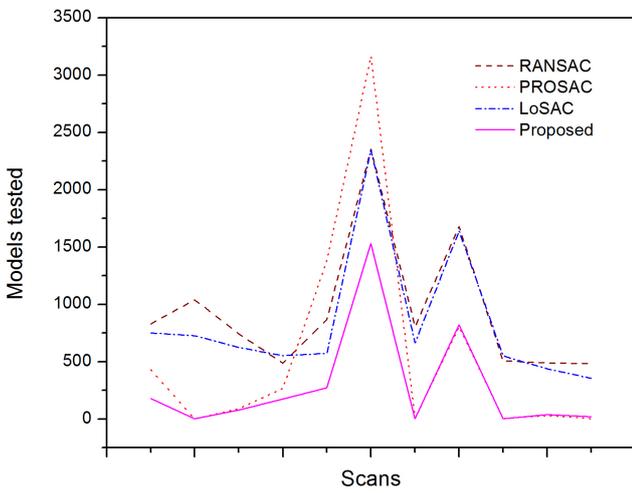


Fig. 17. *Chicken dataset – number of models tested.*

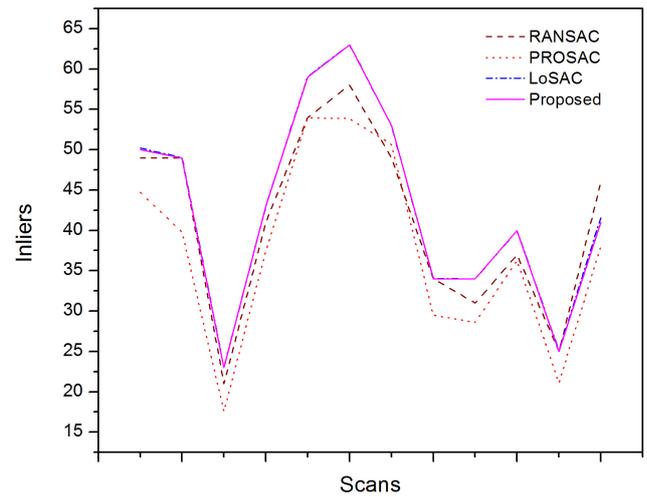


Fig. 20. *Parasaurorolophus dataset – number of inliers.*

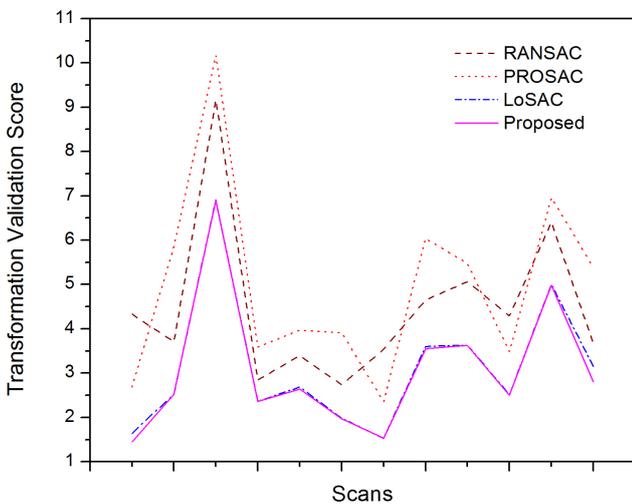


Fig. 18. *Parasaurorolophus dataset – transformation validation score.*

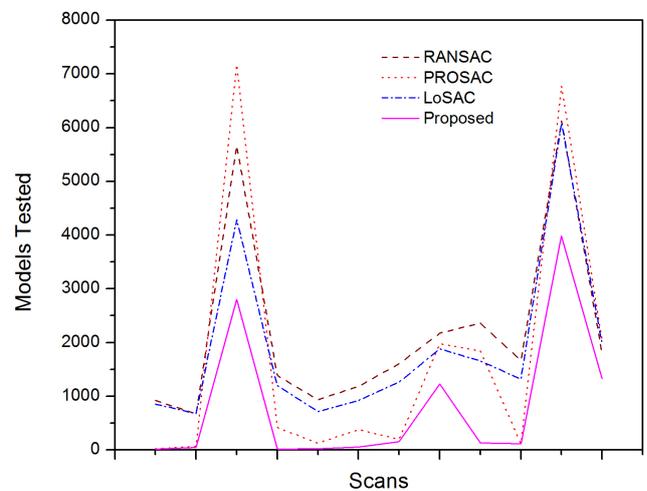


Fig. 21. *Parasaurorolophus dataset – number of models tested.*

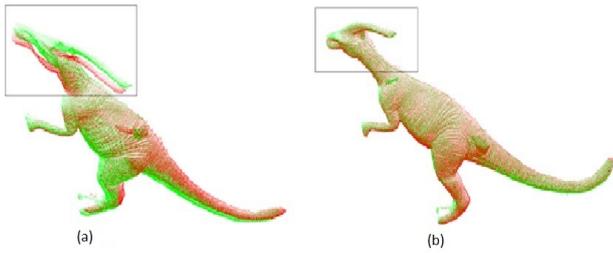


Fig. 22. Registration results of *Parasaurolophus* model scan pair by RANSAC (a) and proposed method (b)

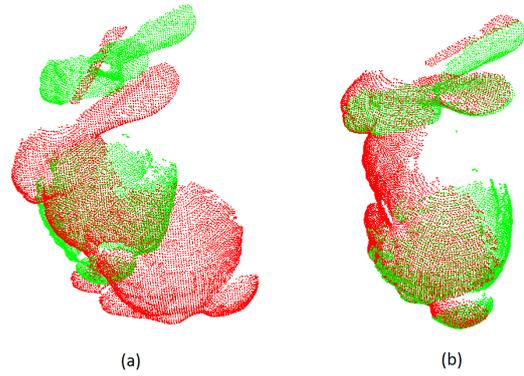


Fig. 25. Bunny dataset scan pair before registration (a) and after initial registration (b).

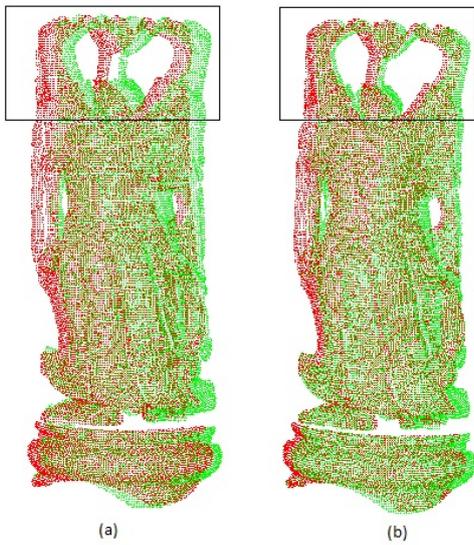


Fig. 23. Registration results of *Buddha* model scan pair by RANSAC (a) and proposed method (b).

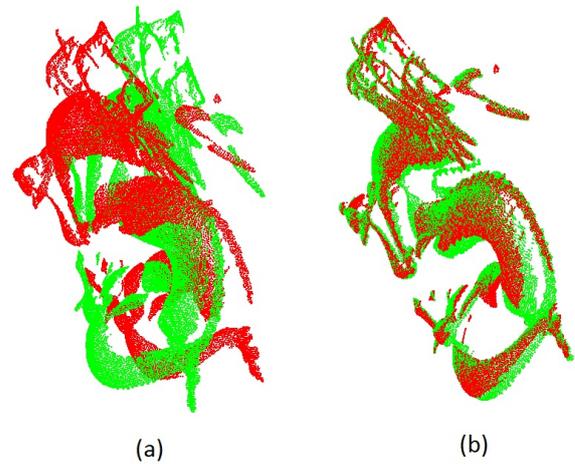


Fig. 26. Dragon dataset scan pair before registration (a) and after initial registration (b).

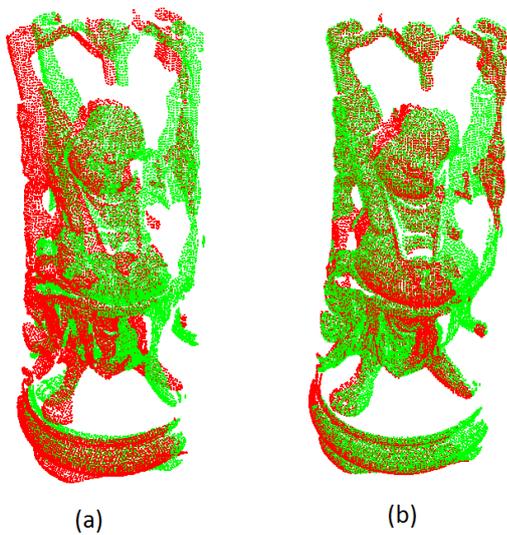


Fig. 24. Buddha dataset scan pair before registration (a) and after initial registration (b).

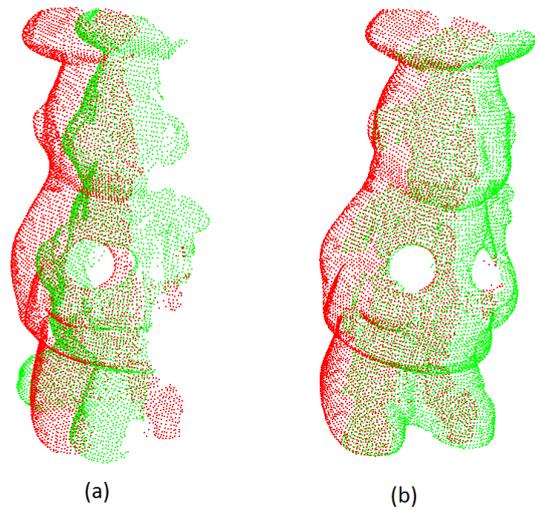


Fig. 27. Chef dataset scan pair before registration (a) and after initial registration (b).

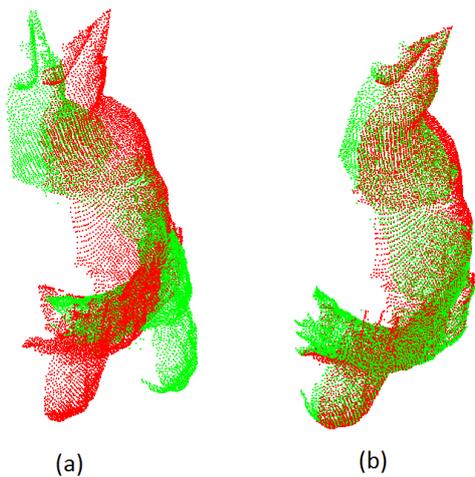


Fig. 28. Chicken dataset scan pair before registration (a) and after initial registration (b).

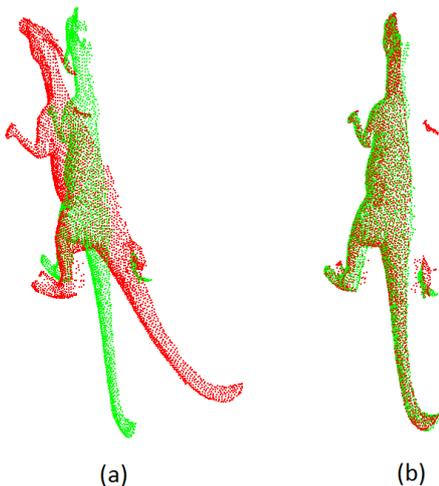


Fig. 29. Parasaurolophus dataset scan pair before registration (a) and after initial registration (b).

DISCUSSION

In the proposed approach, the guided sampling effectively utilizes the correspondence match scores from the 3D feature matching. Considering the probably best correspondence pairs in early iterations helps in reducing the number of models tested and thus results in less computational time. This is evident from the various results discussed. The algorithms which utilizes guided sampling, like PROSAC (Chum and Matas, 2005) and the proposed algorithm, achieves the best running time. However the results obtained by PROSAC are less accurate compared to the other algorithms. This can be attributed to the fact that many possibly good correspondence pairs may not

be considered by PROSAC for model generation due to the early termination. This drawback is rectified with the help of a local optimization step in the proposed algorithm. In the local optimization step, the model generation samples are formed from the set of inliers. This sample is also a non-minimal sample, which helps in including many possibly good correspondences for computing the model. Thus the non-minimal samples formed from inliers results in the generation of good models. In addition to this, since varying thresholds are used for computing the model in various iterations, a local refinement of the computed transformation is carried out. This results in a more accurate estimation of the transformation model. This can be seen from the results obtained. The proposed algorithm achieves the accuracy comparable to that of LoSAC (Chum *et al.*, 2003). LoSAC, however is computationally more intensive which can be seen from the running time which is more than the other algorithms considered. Random sampling results in a large number of models to be tested and this may lead to many local optimization steps. This extra computational burden is compensated by the guided sampling stage in the proposed method. The most likely correspondences evaluated early in the loop leads to local optimization of good models initially. This helps in reducing the number of models tested and hence the running time. The stopping criterion ensures that a good model (i.e formed from inliers) is obtained by the algorithm and this is not a model supported by random outliers. From the above results, we can see that the proposed algorithm performs the 3D registration task with good accuracy as well as computational time.

The proposed method uses the correspondences found by 3D feature matching. The influence of the feature selection process is in deciding the ratio of inliers to outliers in the corresponding pairs. So the influence of feature selection process can be considered equivalent to the influence of the percentage of inliers. The correctness of 3D feature matching depends upon the geometric distinctiveness of the point cloud surface. If the point cloud has less geometrical features, the ratio of inliers to outliers will be less in the found correspondences. A very low inlier percentage will lead to more computational time requirements by the proposed algorithm. In such cases, other 3D features which make use of the colour or texture information may be used. Also in the case of point cloud pairs where the inlier percentage is very high, the extra computational overhead by the local optimization may be made optional by including additional verification in the algorithm.

The proposed algorithm aims at the coarse registration of point clouds and hence can match

partially overlapping point clouds at arbitrary locations. The convergence funnel using the Stanford bunny model point cloud was examined. The model point cloud was rotated and translated and then registered with the original point cloud. Translations (radially) along the x-z plane up to a distance of 6 times the height of the bunny and rotations about y axis with 30 degree increments up to 360 degrees were considered. The method achieved global convergence in all cases. Hence the method can be considered to be not limited by specific rotation or translation limits but only by the distinctiveness of the features selected for the input point clouds.

CONCLUSION

The pair-wise initial alignment of partially overlapping 3D point clouds generally employs 3D feature matching. On account of the possibility of high percentage of outliers in 3D correspondences, robust estimation techniques are often used. The paper presents a novel approach for dealing with the outliers in the set of correspondences for 3D transformation estimation. The proposed algorithm has been implemented for the 3D registration of six different models from two datasets. The performance of the algorithm has been compared with the popular algorithm RANSAC and its variants and the results indicate that the proposed algorithm outperforms them in relative domains of time or accuracy. Since fine alignment stage like ICP requires good initial alignment, the results of the proposed algorithm leads to faster and accurate ICP convergence compared to other methods. The proposed method thus finds a balance between accuracy in the estimate and time of execution.

ACKNOWLEDGEMENT

The authors would like to thank the Stanford University and the University of Western Australia for providing the datasets freely available online. The authors acknowledge the comments and suggestions of the anonymous reviewers which helped improve our submission. The work of the first author is supported by a Doctoral Fellowship provided by the Indian Institute of Space Science and Technology, Department of Space, Government of India.

REFERENCES

- Arun K S, Huang T S, Blostein S D (1987). Least-squares fitting of two 3-D point sets. *IEEE T Pattern Anal* 5:698–700.
- Besl P J, McKay ND (1992). Method for registration of 3-D shapes. *Proc SPIE* 1611:586–606.
- Campbell R J, Flynn P J (2001). A survey of free-form object representation and recognition techniques. *Comput Vis Image Und* 2:166–210.
- Chen H, Bhanu B (2007). 3D free-form object recognition in range images using local surface patches. *Pattern Recogn Lett* 28(10):1252–62.
- Chen Y, Medioni G (1992). Object modelling by registration of multiple range images. *Image Vision Comput* 10(3):145–55.
- Choi S, Kim T, Yu W (2009). Performance evaluation of RANSAC family. *Proc Brit Mach Vision Conf (BMVC '09)* 81.1–12
- Chua C S, Jarvis R (1997). Point signatures: A new representation for 3D object recognition. *Int J Comput Vision* 25(1):63–85.
- Chum O, Matas J (2005). Matching with PROSAC – progressive sample consensus. *P IEEE Comput Soc Conf Comput Vision Pattern Recogn (CVPR 2005)* 1:220–6.
- Chum O, Matas J, Kittler J (2003). Locally optimized ransac. In: Michaelis B, Krell G, eds. *Pattern Recogn. Lect Not Comput Sci* 2781:236–43.
- Dorai C, Wang G, Jain AK, Mercer C (1996). From images to models: Automatic 3D object model construction from multiple views. *P 13th IEEE Conf Pattern Recogn* 1:770–4.
- Filipe S, Alexandre LA (2013). A comparative evaluation of 3D keypoint detectors. *Proc 9th Conf Telecomm Conftele* 145–8.
- Fischler MA, Bolles RC (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun ACM* 24(6):381–95.
- Friedman JH, Bentley JL, Finkel RA (1977). An algorithm for finding best matches in logarithmic expected time. *ACM T Math Software* 3(3):209–26.
- Frome A, Huber D, Kolluri R, Bülow T, Malik J (2004). Recognizing objects in range data using regional point descriptors. In: Pajdla T, Matas J, eds. *P ECCV 2004. Lect Not Comput Sci* 3023:224–37.
- Gomes L, Pereira Bellon OR, Silva L (2014). 3D reconstruction methods for digital preservation of cultural heritage: A survey. *Pattern Recogn Lett* 50:3–14.
- Hough PV (1962). Method and means for recognizing complex patterns. *US Patent* 3,069,654.
- Johnson AE, Hebert M (1999). Using spin images for efficient object recognition in cluttered 3D scenes. *T Pattern Anal* 21(5):433–49.

- Klasing K, Althoff D, Wollherr D, Buss M (2009). Comparison of surface normal estimation methods for range sensing applications. P IEEE Int Conf Robot Autom (ICRA '09) 3206–11.
- Knopp J, Prasad M, Willems G, Timofte R, Van Gool L (2010). Hough transform and 3D surf for robust three dimensional classification. In: Daniilidis K, Maragos P, Paragias N, eds. P ECCV 2010. Lect Not Comput Sci 6316:589–602.
- Koenderink JJ, van Doorn AJ (1992). Surface shape and curvature scales. *Image Vision Comput* 10(8):557–64.
- Mian A, Bennamoun M, Owens R (2010). On the repeatability and quality of keypoints for local feature-based 3D object retrieval from cluttered scenes. *Int J Comput Vision* 89(2-3):348–61.
- Mian A, Bennamoun M, Owens R (2006). Three-dimensional model-based object recognition and segmentation in cluttered scenes. *IEEE T Pattern Anal* 28(10):1584–601.
- Muja M, Lowe DG (2009). FLANN, fast library for approximate nearest neighbors. P Int Conf Comput Vis Theory (VISAPP'09).
- Myatt DR, Torr PHS, Nasuto SJ, Bishop JM, Craddock R (2002). NAPSAC: High noise, high dimensional robust estimation-its in the bag. Proc 13th Brit Mach Vision Conf (BMVC 2002) 2:458–67.
- Pullik K (1999). Multiview registration for large data sets. P IEEE 2nd Int Conf 3-D Digital Imaging Model 160–8.
- Rousseeuw PJ (1984). Least median of squares regression. *J Am Stat Assoc* 79(388):871–80.
- Rousseeuw PJ, Leroy AM (2005). Robust regression and outlier detection, vol. 589. John Wiley & Sons.
- Rusinkiewicz S, Levoy M (2001). Efficient variants of the ICP algorithm. P IEEE 3rd Int Conf 3-D Digital Imaging Model 145–52.
- Rusu RB, Blodow N, Marton ZC, Beetz M (2008). Aligning point cloud views using persistent feature histograms. P IEEE/RSJ Int Conf Intell Robot Syst (IROS 2008) 3384–91.
- Rusu RB, Blodow N, Beetz M (2009). Fast point feature histograms (FPFH) for 3D registration. P IEEE Int Conference on Robotics and Automation 2009, 3212–17.
- Rusu RB, Cousins S (2011). 3D is here: Point cloud library (PCL). P IEEE Int Conf Robot Autom (ICRA) 1–4.
- Salti S, Tombari F, Di Stefano L (2011). A performance evaluation of 3D keypoint detectors. P IEEE Int Conf 3D Imaging Model Proces Visual Transmiss (3DIMPVT) 236–43.
- Stein F, Medioni G (1992). Structural indexing: Efficient 3-D object recognition. *IEEE T Pattern Anal* 14(2):125–45.
- Tam GK, Cheng ZQ, Lai YK, Langbein FC, Liu Y, Marshall D, Martin RR, Sun XF, Rosin PL (2013). Registration of 3D point clouds and meshes: A survey from rigid to non-rigid. *IEEE T Vis Comput Gr* 19(7):1199–217.
- Tombari F, Salti S, Di Stefano L (2010). Unique signatures of histograms for local surface description. In: Daniilidis K, Maragos P, Paragias N, eds. P ECCV 2010. Lect Not Comput Sci 6313:356–69.
- Torr PHS, Zisserman A (2000). MLESAC: A new robust estimator with application to estimating image geometry. *Comput Vis Image Und* 78(1):138–56.
- Levoy M. (2005) The Stanford 3D Scanning Repository. <http://www-graphics.stanford.edu/data/3dscanrep>
- Umeyama S (1991). Least-squares estimation of transformation parameters between two point patterns. *IEEE T Pattern Anal* 4:376–80.
- Wang H (2004). Robust statistics for computer vision: model fitting, image segmentation and visual motion analysis. PhD Thesis. Monash University.
- Zhong Y (2009). Intrinsic shape signatures: A shape descriptor for 3D object recognition. P IEEE 12th Int Conf Comput Vision Worksh (ICCV Workshops) 689–96.