WORK REPORT

27 Nov 2017 – 22 Dec 2017

Divya K Raman

Topics:

1. SIFT
2. Fast Feature Point Histogram
3. Motion Averaging

Resources studied:

1. SIFT
   http://aishack.in/tutorials/sift-scale-invariant-feature-transform-introduction/
2. FPFH
   http://pointclouds.org/documentation/tutorials/pfh_estimation.php
   http://pointclouds.org/documentation/tutorials/fpfh_estimation.php
   Paper: Rusu, R. B., Blodow, N., & Beetz, M. (2009, May). Fast point feature histograms (FPFH) for 3D registration. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on* (pp. 3212-3217). IEEE.
3. Motion Averaging
   Paper: Govindu, V. M. (2004, June). Lie-algebraic averaging for globally consistent motion estimation. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on* (Vol. 1, pp. I-I). IEEE.

Analysis:

1. Feature based 3D reconstruction

For each point cloud, fpfh features are generated using the fpfh_features function. For this, the cpp file fpfh_features.cpp needs to be converted to a mex file after which it can be used as a matlab function. This function takes in the point cloud and the normals at each point. Using this data, a 33 dimensional feature vector for each point is generated. point3D_matching.cpp is also converted to a mex file. This function takes in the fpfh features and the normals for 2 point clouds and returns the matched points. Umeyama's method returns the transformation that best aligns the 2 point clouds(the set matched points). The umeyama estimates the transformation between a set of matched points using an SVD based approach.

We then do a motion averaging to align all the scans(72 point clouds, ~5 degrees apart) of the elephant model obatined using a Kinect camera by placing the elephant on a turn table. Motion averaging using the spanning tree paths(using only Scanj to Scani transformations where j-i = 1) gives us the minimal solution. We also check how motion averaging works when more paths(more Scanj to Scani transformations) are added. One general observation is that the number of iterations needed for convergence increases as the number of paths increases. |j-i|<=1 gives us the solution corresponding to a spanning tree plus one loop closure, this reconstruction is far better in than the minimal solution reconstruction. We try out motion averaging by adding more paths such that |j-i|<=k. The final 3D reconstruction gets better with increasing k. Beyond a threshold, the quality of reconstruction again decreases. This is because as |j-i| gets larger, correct feature matches are hard to find. The optimal k for best reconstruction seems to be 6 in our experiment. In particular, reconstruction of the trunk and tail of the elephant is not too good. This is because we don't get good feature matches in this region owing to its sharp nature where even a 5 degree turn gives us quite a different view of the region. A solution to this is to obtain more scans of the trunk and tail region, preferably spaced 0.5-1 degree apart. The quality of the final reconstruction is also affected by the noise present in the dataset.

2. icp based 3D reconstruction

Here, transformations from Scanj to Scani are obtained using icp. The same experiment as above is repeated using both the stanford bunny(ideal dataset) and the elephant. The optimal value of k is 3 here, lower than that in the feature based approach. This is because icp doesn't work too well when the 2 point clouds between which we wish finding the transformation are far apart.

The 3D reconstruction of the bunny is far better than that of the elephant. This can be attributed to the fact that the bunny is an ideal dataset while the elephant is a noisy one.

An improvisation of the reconstruction of the elephant is obtained using the MAICP algorithm. The principle of this algorithm is to iteratively perform motion averaging in order to get the best 3D reconstruction possible. Tranformations from scanj to scani of the initial set of 72 scans are obtained using icp, and using these motion averaging is done to get an initial estimate of the absolute positions(orientation same as that of scan 1 in our case) of all the 72 scans. Using these absolute positions, transformations from scanj to scani are found on which a motion averaging step is again performed. This refines the absolute position of the 72 scans and gives us a better 3D reconstruction. This can be done repeatedly to get a good reconstruction. However, beyond three iterations, the quality of reconstruction doesn't improve much(reaches convergence). We will then have to look for more robust algorithms to improve our reconstruction further.

Fitting error:

It is hard to find an absolute ground truth to determine how good our reconstruction is. The fitting error can be used as a parameter to determine the quality of our 3D reconstruction.

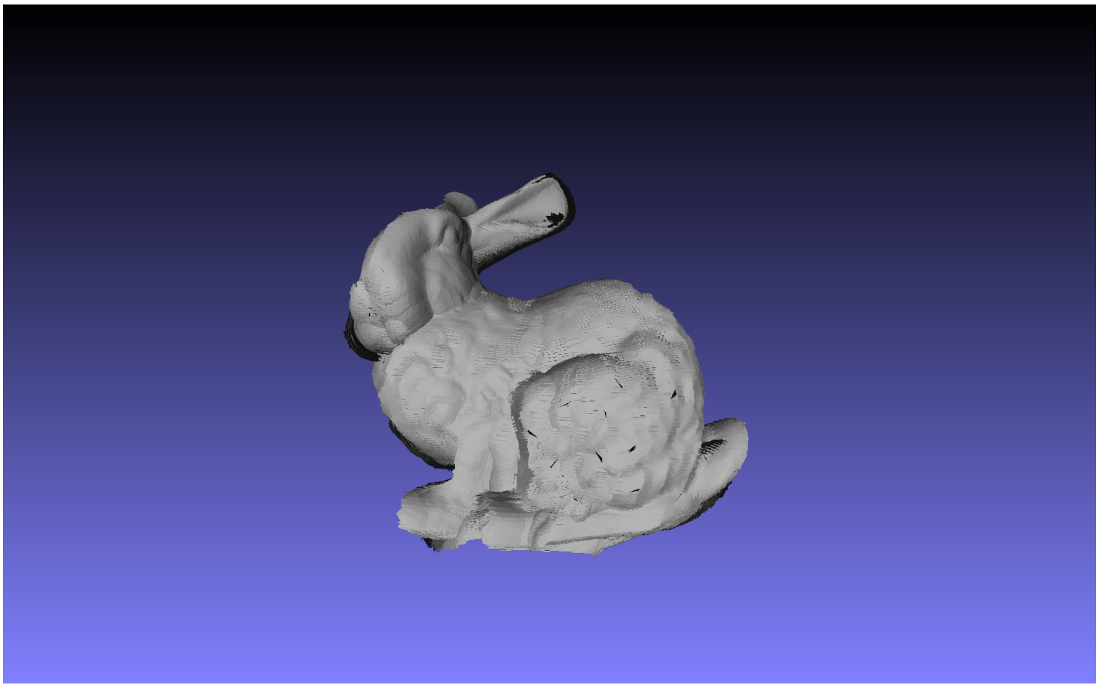
Mi gives the motion from 1st scan to ith scan(to align them). Let pi be the set of vertices of the ith scan. inv(Mi)*pi gives the absolute position of the ith scan(wrt 1st scan). Let S be the set of all scan pairs {i,j} whose transformation we used for motion averaging(i.e. the path/motion from scanj to scani or vice versa).The following pseudo code details the calculation of the fitting error.

```
fitting error = 0;
for i=1:72

        absi = inv(Mi)*pi

        for all j such that {i,j} is in S

                absj = inv(Mj)*pj;

                Ci = corresponding points between absi and absj present in
absi; Cj = corresponding points between absi and absj present in absj

                fitting error = fitting error+(frobenius norm(Ci-Cj))

        end

end
```

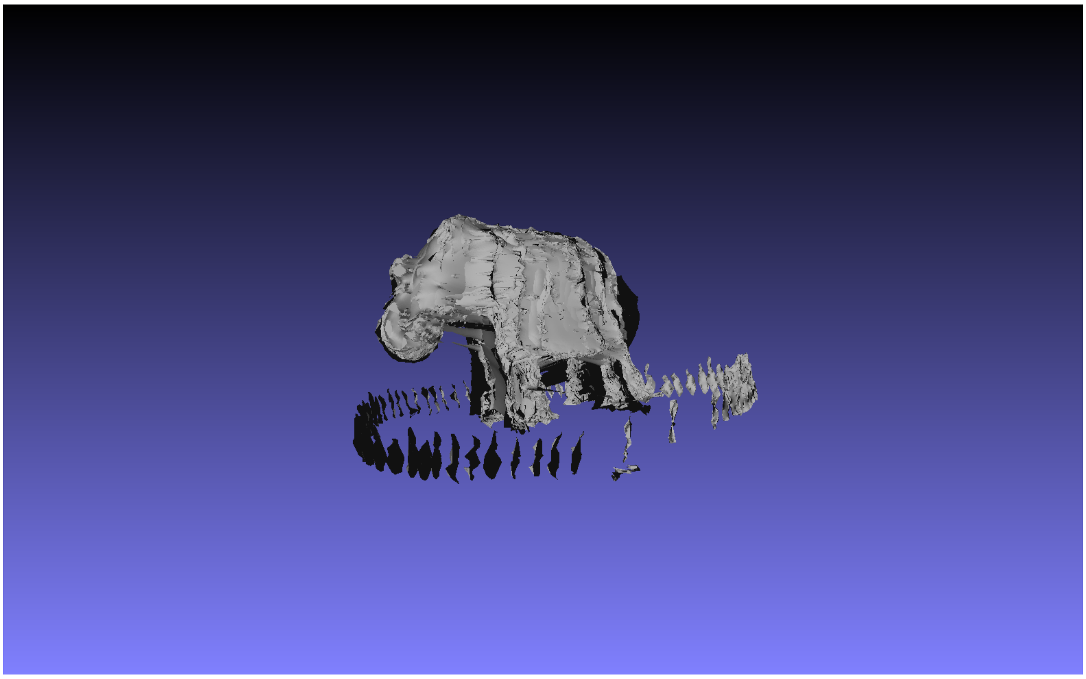
The fitting error obtained above is normalised using the diameter of the 3D object.

Results:

3D reconstruction of stanford bunny(ICP,followed by motion averaging)



3D reconstruction of the elephant(real dataset, MAICP)



Absolute motion expected results:

Translation from scani to global orientation = 0(we are dealing with turn table sequences)

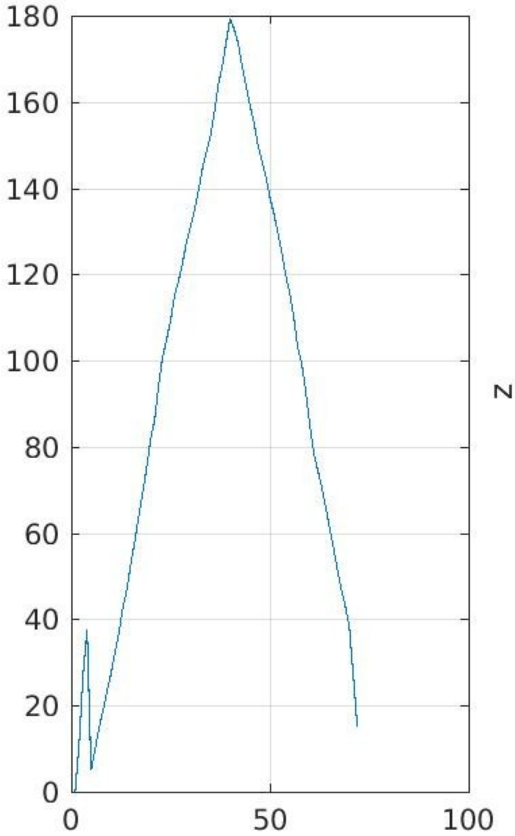
Rotation angle from scani to global orientation = (i-1)*5 degrees(we have 72 scans spanning over 360 degrees)

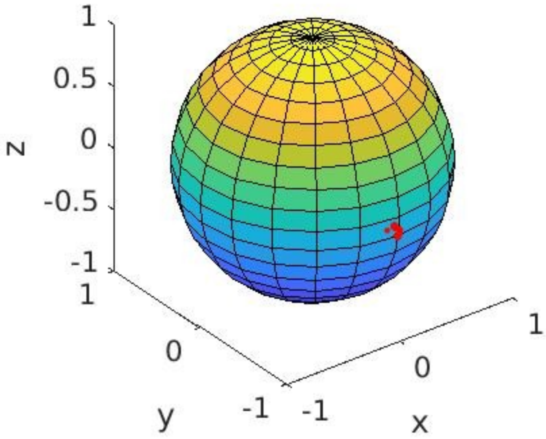
Rotation axis in all cases = y axis.

(global orientation = orientation of scan1)

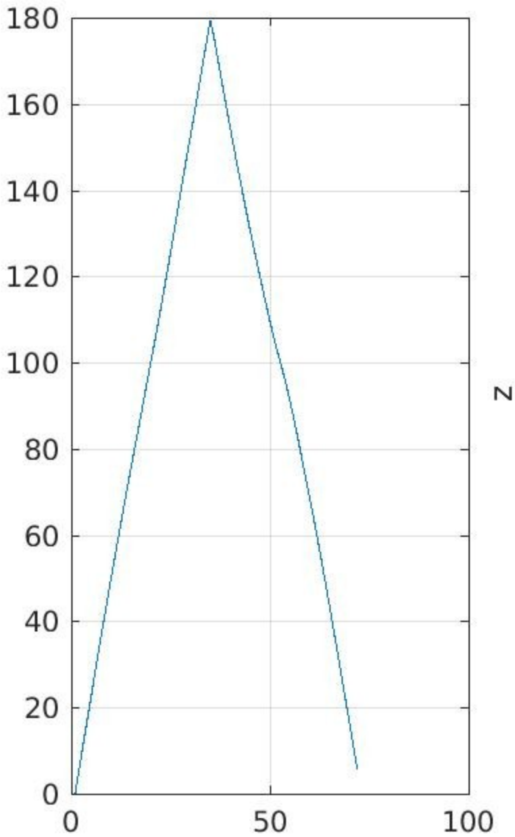
The actual results obtained are as follows:

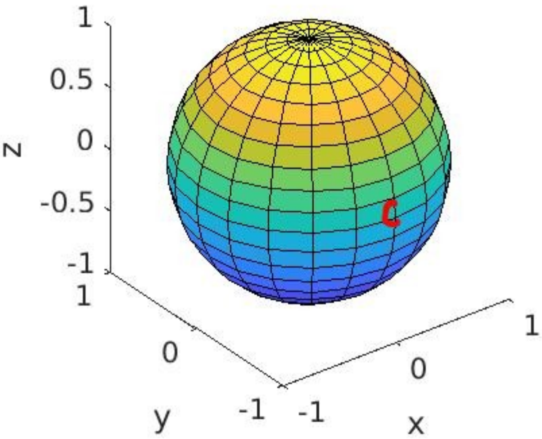
Elephant





Stanford bunny:





Stanford bunny being an ideal dataset gives us ideal results, while there are deviations in the results of 3D reconstruction of the elephant. This is mainly due to noise and the quality of the scans.

The diameter of the elephant is found to be 0.7.

Translation from absi to global orientation is non zero in case of the elephant, i.e. there does exist a translation error. Translation error is almost zero in case of the stanford bunny.

Acknowledgements

I wish to express my heartfelt gratitude to Prof. Venu Madhav Govindu for providing me with an opportunity to work in his lab and for his kind guidance and encouragement. I also wish to thank Uttaran for his immense help in my work here.