ENGN2502: 3D Photography
Assignment 3: Merging Point Clouds

Instructor: Gabriel Taubin
Assignment written by: Fatih Calakli

5 April 2012

Introduction

The triangulation-based 3D scanning methods built in previous assignments are able to produce dense point clouds. However, only points on the front-facing side of the object can be reconstructed (*i.e.*, on the same side as the projector and camera). To produce a complete representation, multiple scans taken from various points of view must be integrated to produce a point cloud with sufficient sampling density over the whole visible surface of the object being scanned.

1 Computing Rigid Body Matching Transformations

The main challenge to merging multiple scans is that each scan is produced with respect to a different coordinate system. As a result, the rigid body transformation needed to register one scan with another must be estimated. Assuming that true point correspondences are already established, one can find the rigid body transformation using the closed form solution presented in [1].

What to turn in: Write a function named `align` that computes rigid body transformation to align two point clouds from N correspondences, $\{(p_1, q_1), ..., (p_N, q_N)\}$. Your function should take as input two points clouds and an array of correspondences, and output rigid body transformation (R, T) relating the source point cloud to the target point cloud. In order to test your function, take `pointCloud1` as your source point cloud. Apply small rigid body transformation to generate your target point cloud. In this case, you know exact point correspondences. Using all the correspondences, estimate the transformation and compare with the applied transformation.

2 The Iterative Closest Point (ICP) Algorithm

If the two points clouds are in approximate registration, one can use ICP which iteratively estimates the rigid body transformation between two point clouds. The algorithm comprises the following steps:

1. Select points from the first shape.
2. Associate points, by nearest neighbor with those in the second shape.
3. Estimate the closed-form matching transformation using the function `align`
4. Transform the points using the estimated parameters.
5. Repeat previous steps until the mean matching error is below a certain threshold.

What to turn in: This section doesn't require any implementation. Nevertheless, explain in words how you would merge if you had more than two partial scans covering a whole object. Assume that each point cloud is time-stamped, and the object moves slowly in front of the 3D scanner.

3 Manual Initialization

If the points clouds are not in approximate registration, ICP typically fails to find the correct transformation since the closest neighbors would not be approximate correspondences anymore. In this case a user input may be used to establish initial approximate correspondences[1]. This feature, enabling user to choose point correspondences, has been implemented in Meshlab[2]. Once the initial correspondences are established this way, ICP proceeds as usual.

What to turn in: Use Meshlab Alignment tool[3] to merge provided point clouds `pointCloud1`, `pointCloud2`, and `pointCloud3` to get a single point cloud. Save your project file as `alignment.aln`, and turn it in with your submission.

References

[1] K. S. Arun, T. S. Huang, and S. D. Blostein. Least-squares fitting of two 3-d point sets. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-9(5):698 –700, sept. 1987.

[2] A. Makadia, A.I. Patterson, and K. Daniilidis. Fully automatic registration of 3d point clouds. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 1297 – 1304, june 2006.

[1] Fully automatic alignment starting from arbitrary initial poses has been an active research area[2]

[2] `meshlab.sourceforge.net`

[3] Check `http://www.youtube.com/watch?v=4g9Hap4rX0k` for a tutorial.