Project 7: Semantic Mapping

Swati Singhal
Dept. of Computer Science
University of Maryland
College Park, Maryland 20740

Rohan Chandra
Dept. of Computer Science
University of Maryland
College Park, Maryland 20740

***Abstract*—Implementations of ICP, 3D Object Extraction, 3D Object Segmentation and Semantic Mapping**

I. 3D Reconstruction

This consists of two steps. The first step is *background elimination* where we remove all extraneous planes from the image, including the tabletop, walls, ground and other miscellaneous items. This is required for objection segmentation later on where we don't want outliers interfering with our ICP algorithm.

The second step is local *ICP or Iterative Closest Point*. This is a frame-by-frame pointcloud matching algorithm that minimizes the reprojection error between point correspondences in succesive frames.

A. Background Elimination

We employed the RANSAC algorithm presented in Nitin's research paper. We randomly select 3 points and fit a plane and identify inlier points on that plane. Accordingly we choose all non-plane points. We perform this RANSAC step twice to remove multiple planes.

We also apply a novel post processing step to clean sensor noise. This involves two steps. First we compute the distance to each point from the mean of the RANSAC filtered point cloud. Anything below a factor of the size of the model is retained as object candidates. In many cases, we still retain part of the ground(in the case of multiple objects) as well as the upper part of the wall(in the case of single objects). For single objects, we make use of the simple fact that the wall is almost invariable above an approximate 'z' value. We thus refine our points by removing all points above $z > 1000$. In the case of multiple objects, we retain wall plus the ground. We perform a box cropping step where we crop out the desired scene objects within set x-y axis limits.

B. Iterative Closest Point

The local ICP is non-convex as it includes unitary (rotational matrices) in the objective function and the set of rotation matrices are non-convex. This means that a global optimum cannot be ensured and a local optimum is only desirable in frames that do not change by rotation of more than $\alpha > 20°$. For this reason, we use the first 16 frames for our experiments and preliminary results which have been presented below.

The algorithm for ICP is given as follows:

Algorithm 1 ICP

Require: Input Pointclouds p and q
1. p,q $\longleftarrow$ ComputeCorrespondences(p, q)
2. p,q $\longleftarrow$ RemoveOutliers(p, q)
3. n $\longleftarrow$ ComputeNormals(q)
4. x $\longleftarrow$ A^+b where A^+ is the pseudo-inverse of A
5. Reconstruct R and T from x

C. Results

We obtain the following results for 3D reconstruction of single and multiple objects.

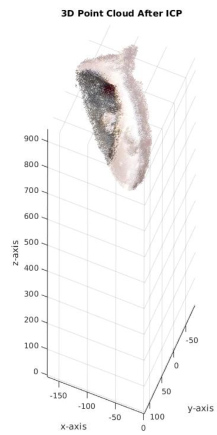

Fig. 1. Single 1: Iron

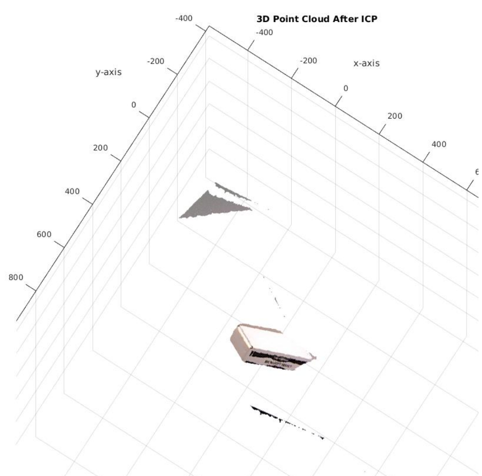

Fig. 2. Single 2: Box

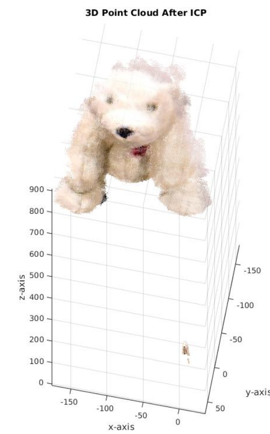

Fig. 3. Single 3: Teddy

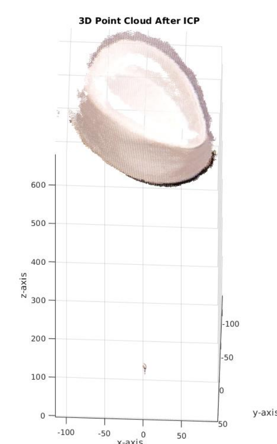

Fig. 4. Single 4: Hat

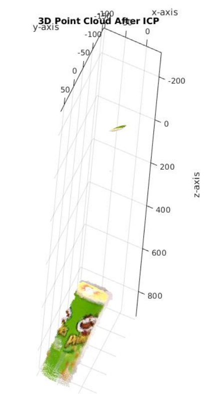

Fig. 5. Single 5: Pringle

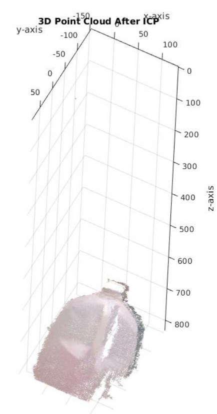

Fig. 6. Single 6: Jug

The following video show the reconstruction process over 50 frames.

Fig. 7. Video for 3D single object reconstruction

Results for multiple scene reconstruction are as follows:

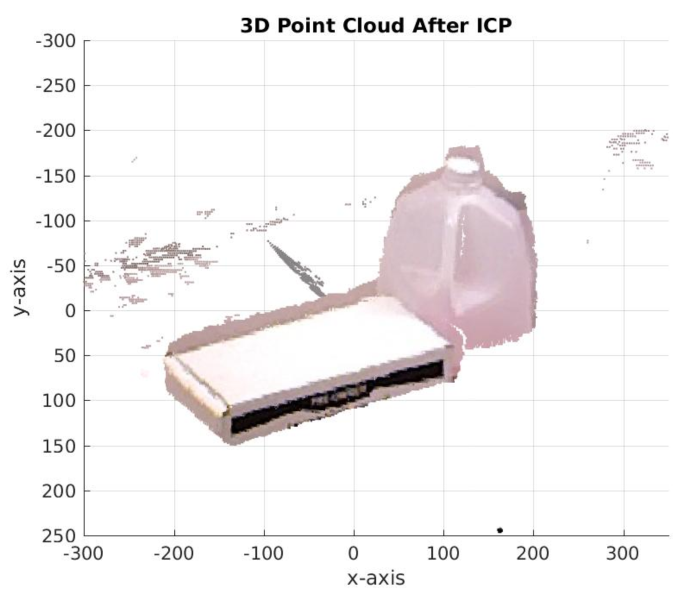

Fig. 8. Multiple Scene 1

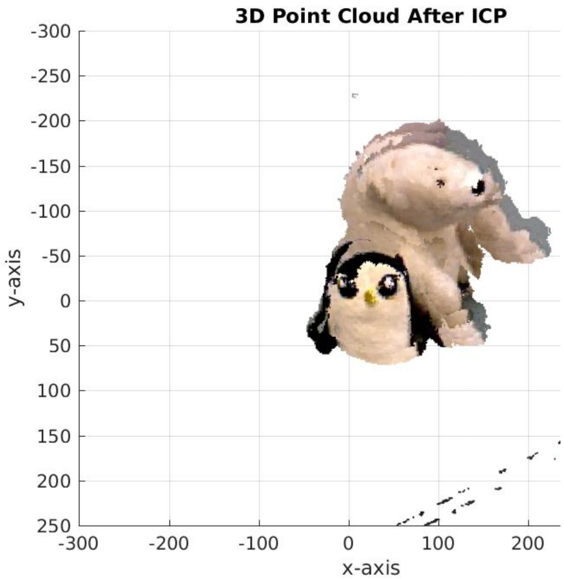

Fig. 9. Multiple Scene 2

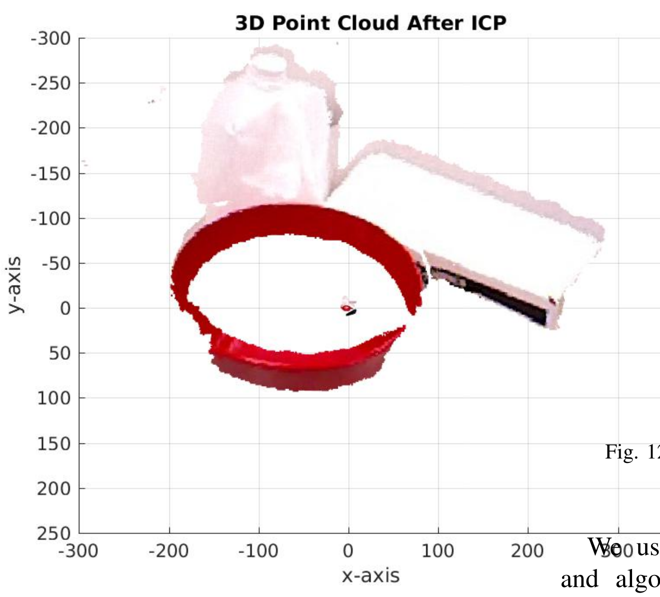

Fig. 10. Multiple Scene 3

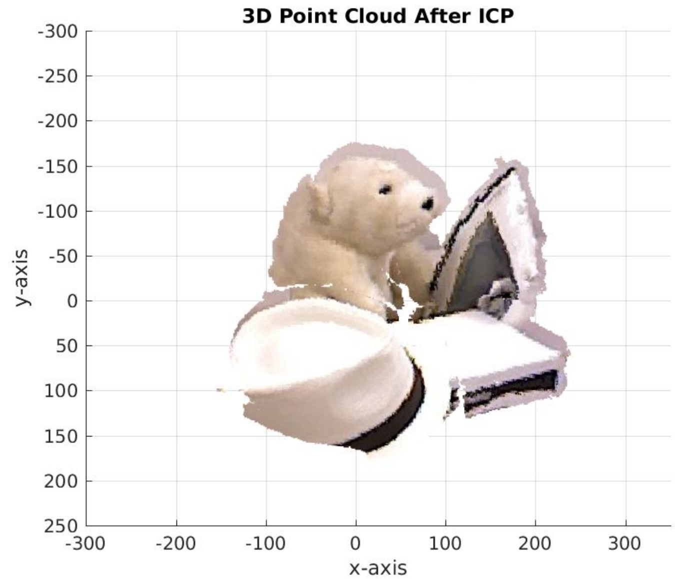

Fig. 11. Multiple Scene 4

The following video show the reconstruction process over 16 frames.

Fig. 12. Video for 3D multiple object reconstruction

II. SEGMENTATION

We used two approaches, namely, algorthm 3 and algorithm 4. These can be summarized as follows:

Algorithm 2 Approach 1: Segmentation

Require: Single and multi object point clouds
  for multiple object scene **do**
    for single object scene **do**
      Do ICP and find matches
    end for
  end for

We get the following result from approach 1:

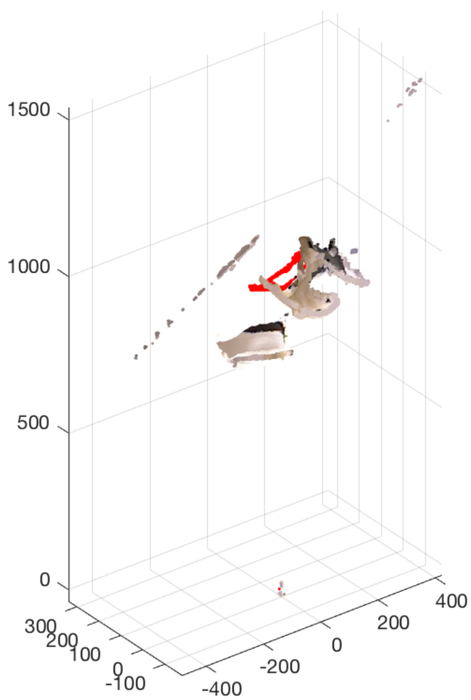

Fig. 13. Failiure of local ICP

Algorithm 3 Approach 2: Segmentation

Require: Single and multi object point clouds, R_o, T_o
  Define few possible rotation matrices and an initial translation matrix
  for R **do**
    find an optimal translation Matrix using non-linear least square
  end for
  p,q $\longleftarrow$ ComputeCorrespondences(p, q)
  p,q $\longleftarrow$ RemoveOutliers(p, q)
  n $\longleftarrow$ ComputeNormals(q)
  x $\longleftarrow$ $A^{+}b$ where A^{+} is the pseudo-inverse of A

  Reconstruct R and T from x

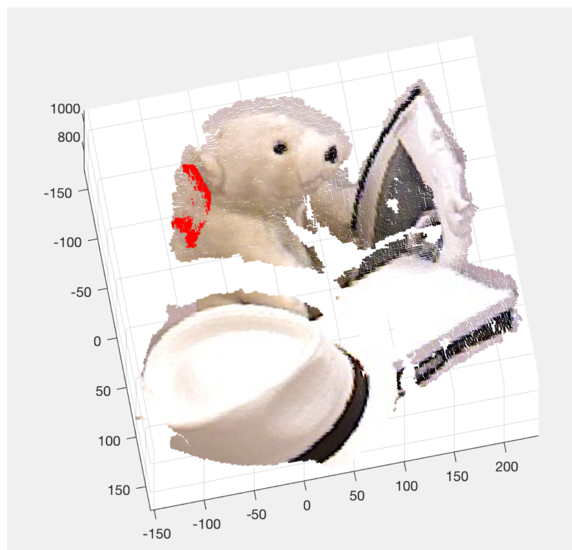

Fig. 14. success of guided ICP

Although the result is still far from perfect, we can see we have successfully managed to identify a small part of the teddy bear.

III. Semantic Mapping

Semantic Mapping are maps or webs of words. The purpose of creating a map is to visually display the meaning-based connections between a word or phrase and a set of related words or concepts. Semantic maps help students, especially struggling students and those with disabilities, to identify, understand, and recall the meaning of words they read in the text.

In computer vision, integrating this concept with images leads to a very exciting intersection in Natural Language Processing. In this project, we are required to 'connect' images with meaningful words or phrases that would be similar to how a human would normally think of the scene.

There were two approaches that we could think of to complete this part. One wa to train deep networks. Using a dictionary of pre-compile dwords and phrases and an image dataset, one can set up a simple caption based semantic system. Unfortunately our team does not posses the required skillset to train and engineer workable deep neural models yet!

The other approach somewhat less glamourous is to build detection models based look-up table based on visual cues like shape and color. Using these basic models, one can already answer a bunch of various meaningful questions such as:

- Which is the most red colored object in the scene?
- Which is the most rectangular object ?

and so on.

In our project we have managed to build several such models including a shape detector and a color detector. Our models are robust so as to deal with different aignments of the object and to occlusion. We have so far managed to detect cuboids and red colored objects and the results can be shown below:

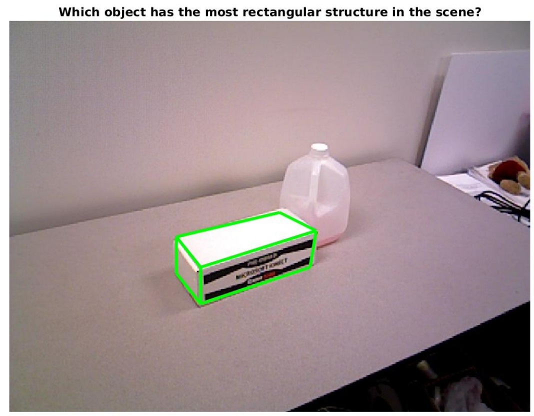

Fig. 15. Output from Shape detector

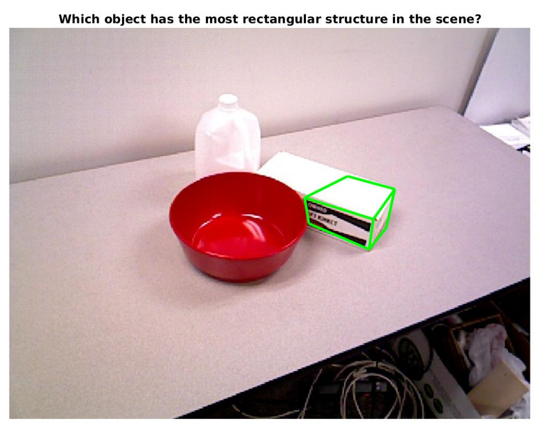

Fig. 16. Output from shape detector with occlusion

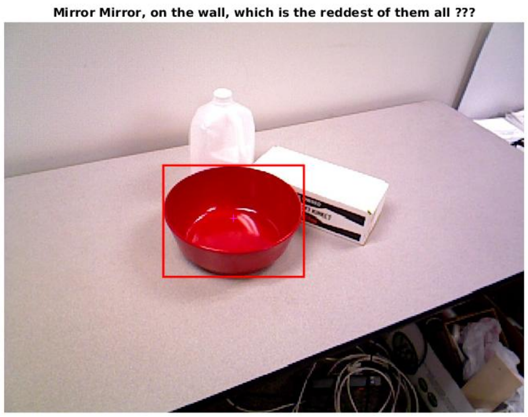

Fig. 17. output from color detector