Segmentation and Pose Estimation of Planar Metallic Objects

Haider Ali and Nadia Figueroa

Institute of Robotics and Mechatronics (RM)

German Aerospace Center

Oberpfaffenhofen-Wessling, Germany

Email: haider.ali@dlr.de, nadia.figueroa@dlr.de

Abstract—**The problem of estimating the pose of metallic objects with shiny surfaces is studied. A new application has been developed using state-of-the-art 3D object segmentation (euclidean clustering) and pose estimation (ICP) methods. We analyze the planar surfaces of the metallic objects in 3D laser scanner data. First we segment these planar objects using euclidean clustering based on surface normals. Thereafter to estimate the pose of these segmented objects we compute Fast Point Feature Histograms (FPFH) descriptors. Finally we use an ICP algorithm that computes the rigid transformation with Singular Value Decomposition(SVD). Two different round of experiments are conducted:- one for the clustering and the other one for the pose estimation. We present the experimental results and analysis along with the possible application scenario and future work.**

Keywords-**Euclidean Clustering; FPFH descriptors; ICP; 3D segmentation; Planar Objects;**

I. INTRODUCTION

The pose estimation has gained a lot of attention in computer vision community in recent years. Although there is still a gap to be filled by bringing the computer vision online/offline recognition systems for the robotics community. We focus on a simple planar object pose estimation method at the cost of acceptable recognition rate. The high recognition is not required in this typical pile of object application. The objects which will not be recognized in the first round will be treated in the next data scan.

In this work, we investigate the object recognition and pose identification of metallic objects for bin picking applications. We studied a two step procedure:- (i) segmentation of the planer objects using euclidean clustering based on the angular deviations between geometric surface normals, and (ii) the pose estimation of extracted clusters using Fast Point Feature Histograms (FPFH) descriptors with the representation of a rigid transformation.

II. RELATED WORK

A Generalized Hough Transform based 3d object detection method has been proposed by N. Guil et. al [3]. They calculate tilt and pan transformations between the image and template to detect planar objects. Another Pose estimation for planar structures has been proposed by Gilles Simon et. al [4] by computing homo-graphic projection matrix in the image. They compute camera pose using homo graphic information between consecutive images. The method is composed of single plane and multi plan registration schemes and has been tested in indoor as well as outdoor environments.Carmen de Trazegnies et. al [5] has proposed a contour based 3d planar object recognition method. A linear combination of planar images in a sequence of views has been introduced in the Fourier domain using Hidden Markov Models. A 3d planar object pose estimation method using eigenspace analysis and range data analysis has been investigated by Ovidiu Ghita et. al [7]. The approach is based on a rejection criteria by computing surface normal in the scene and relevant object model. Another normal estimation of 3D planer objects pose using Bayesian inference method has been proposed by Changhai Xu et. al [8]. The method is robust to find an optimal solution for low-textured and high-textured planer objects in a noisy data. Mondragon, I.F. et. al [9] has proposed a real time 3d UAVs pose estimation using projective transformation of planer objects. This method has been tested to track homographies based planar objects in aerial data.

We address the problem of pose estimation by computing the rigid motion from an identified object with an unknown pose to an object model with a known pose. When using 3D

scanners, this rigid motion is obtained by registering the 3D point clouds. 3D registration is classified into two problems, coarse and fine registration [6]. Their difference is based on whether an initial guess of the rigid motion is available. If this initial guess is available, a fine registration is sufficient. However, in most cases this information is not available. Coarse registration techniques generate an initial guess of the motion between two point clouds. This guess is estimated by matching correspondences. The correspondence matching can use *local descriptors* which represent the surrounding surface of a single point (Point Signatures [18], Spin Images [15]) or *global descriptors*, which are representations of the surface of the complete point cloud (Principal Component Analysis [16], Algebraic Surface Models [17]).

Since the introduction of point signatures and spin images, new local feature descriptor alternatives have been developed. These alternatives differ on their computation techniques, either as a signature/histogram or on the description of relationships between neighboring points, based on their surface normals (Point Feature Histograms [12], Signature of Histograms of Orientations [19]).

The matching and rigid motion estimation step is generally implemented with RANSAC-based algorithms, as introduced by Chen et al.[20] and Feldmar [21]. Chen et al. sample point triplets between point clouds to find the best euclidean motion. They demonstrate that 3 points is the minimum required to estimate the rigid motion, if no additional information on the points is available. Feldmar uses a single sampled point, however he considers the surface normal and principal curvature of the point. These techniques have been extended to using local features. Radu et al. [11] introduce a Sample Consensus Initial Alignment algorithm (SAC-IA), that samples point triplets based on their Point Feature Histogram correspondence.

Fine registration methods use an initial estimation to converge to a more accurate solution. The most widely used fine registration method to date is the Iterative Closest Point (ICP). The point-to-point ICP algorithm was first described by Besl and McKay [13]. Zhang [14] adds a robust outlier rejection in the matching correspondences stage. Chen and Medioni [22] created the point-to-plane variant, which considers the locally planar surfaces.

We use the Fast Point Feature Histogram (FPFH) de-

scriptor with the SAC-IA to find our initial guess [12]. For fine tuning the initial guess, we use an ICP algorithm that computes the rigid transformation with Singular Value Decomposition(SVD). These methods have been implemented within the framework of the Open Source Point Cloud Library (PCL) [23].

III. DATA ACQUISITION

We have collected the data using the hand held laser scanner ModelMaker D100 [3]. This can make scans with stripes of width 50,100 and 200 mm and provides better scans with glossy and highly changing contrast surfaces. The data has been collected with a minimum sample distance (0.4mm) and the maximum sample distance (1.0mm) as shown in the (Fig.1), (Fig.2).

Figure 1: Laser Scanner ModelMaker D100: Scanning Scenario

IV. SURFACE NORMALS ESTIMATION

The surface normals are generated using surface normal estimation from the mesh based on the sensor viewpoint as presented in the (Fig.3). A triangulation mesh is composed of set of vertices $PV = [p_i : 1 \leq i \leq n_P]$ and triangles $T = [t_j : 1 \leq j \leq n_T]$. The surface of each triangle are computed. The surface normals of each vertex point $n = (n_x, n_y, n_z)$ are estimated by weighting the surface normals of the neighboring triangles [1],[2].
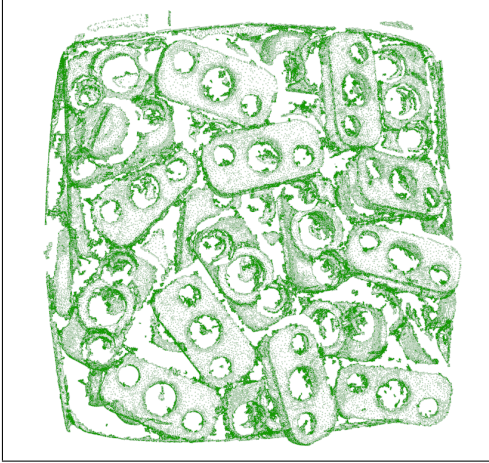
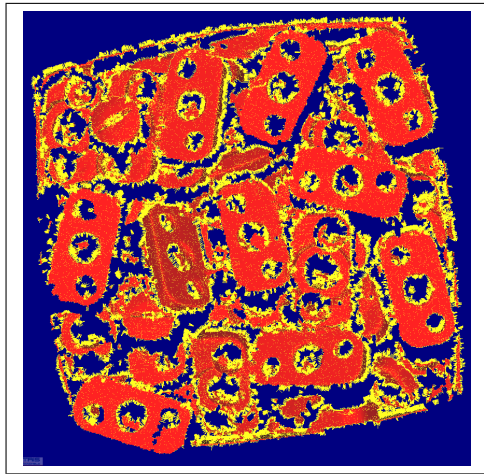Figure 2: An example scene with a un-ordered pile of objects: Closer View

Figure 3: A pile of objects with surface normals

V. EUCLIDEAN CLUSTERING

We have a scene containing a pile P of unorganized metallic objects placed on top of each other inside a box. The pile consists of the same metallic objects, but with different orientations. These objects have a completely planar surface on one side and a semi-planar surface on the other.

The objective is to find and segment the objects whose planar surface is visible to the sensor. This is done by extracting n-clusters C from pile P that represent the planar objects. A simple Euclidean clustering consists on extracting clusters of data that lie within a certain search radius. We extend this technique to only cluster data that represents a planar surface. We do this by analyzing the angle deviations between the surface normals of the clustered points within

a search radius.

The estimated surface normal directions n_x, n_y, n_z represent the surface normal vector n. We compute the angle θ between two normals n_i, n_j as follows:

$$\theta_{ij} = \arccos(n_i \cdot n_j) \qquad (1)$$

This angle is computed for every point-pair within a Euclidean cluster. We set an angle threshold ε. ε is the maximum permissible angle between surface normals that can belong to a plane. If $\theta_{ij} > \varepsilon$ for points p_i, p_j within a euclidean cluster, p_j is rejected from the cluster. This analysis is applied to each point neighbor during the Euclidean cluster creation. For fast nearest neighbor search, we use a kd-tree representation of the point cloud [11]. This was implemented under the Open Source Point Cloud Library Framework [23].

A. The Clustering Algorithm

The algorithmic steps are the following:-

1) We acquire a point cloud of pile P containing 3D coordinates and n_x, n_y, n_z directions of each points surface normal.

2) We create a kd-tree representation of the point cloud of the pile P based on the 3D coordinates.

3) We create an empty list of clusters C and a list of points that need to be processed L.

4) For every point p_i in P, we perform the following steps:

   - Add p_i to the list of points that need to be processed L.
   - For every point p_i in L, we perform the following steps:
     - We search for the nearest neighbors NN^i of p_i within a sphere of radius $r < d_{tl}$
     - For every neighbor nn_k^i, we check if the point exists in L, if it exists we skip to the next neighbor nn_{k+1}^i, otherwise we apply the following check.
     - We estimate θ for p_i and its valid neighbor nn_k^i, if $\theta < \varepsilon$ we add it to L, otherwise we skip to the next neighbor
   - When all points of L have been processed, we add it to the list of clusters C, and reset it to empty.

5) This algorithm terminates when all points in P have been processed and belong to one cluster.

B. Experimental Results

We set the following parameters for the decomposition of a region of space into clusters based on the euclidean distance between points, and the normal angular deviation.

- d_{tl}: Is the maximum spatial cluster tolerance. It is computed as a measure in the L2 Euclidean space. We have analyzed the points distribution of the object model generated by the laser scanner and found that the minimum sample distance is 0.4mm. Considering this, we limit the cluster growing to one or two nearest neighbors so we use $d_{tl} = 1mm$.
- ε: This is the maximum allowed angular difference between normals in radians for cluster/region growing. This is the most important parameter for the specific clustering of planar objects. Initially, we have computed the angle difference between normals of a planar model. We take this into account and identified 0.3 rad as a maximum tolerance ε for discarding points that do not lie on a plane.
- *MinClusterSize*: Not all extracted clusters are used for identification. We limit the size of the clusters by setting a minimum number of points that a cluster may contain. We have chosen 1000 points as a minimum size of the cluster, this value is the total number of points contained in the planar object model.

In (Fig.4) all the extracted clusters are shown which are below the above specified threshold parameters in the euclidean clustering process.

VI. CLUSTER REGISTRATION

We estimate the pose of a specific object cluster C_i by registering it to a model M. We use a two step 3D Registration procedure:-

1) Coarse Registration: We match correspondences between C_i and M based on local features. Using these correspondences we estimate a rough rigid transformation between them.
2) Fine Tuning: The rough rigid transformation is used an initial guess for a fine registration method like Iterative Closest Point (ICP) [13] [14].

Figure 4: An example of extracted clusters

For the Coarse Registration step we compute Fast Point Feature Histograms (FPFH) [12] descriptors to find the correspondences between the model M and the cluster C_i.

FPFH Descriptors represent the geometry of a point's neighborhood, by computing histograms of relationships between point-pairs. The relationships are the relative difference between a point-pair's 3D coordinates x, y, z and surface normals $n = (n_x, n_y, n_z)$. These relative differences are computed as a triplet of angles $< \alpha, \phi, \theta >$ between a pair of normals n_i, n_j within a fixed coordinate frame [12].

To find correspondences in the FPFH search space and estimate a rigid transformation, we use the Sample Consensus Initial Alignment (SAC-IA) method proposed by Rusu *et al* [11] [12]. In this method, correspondence candidates are sampled and ranked by computing an error metric. The rigid transformation that generated the lowest error metric is used as the initial alignment.

The fine tuning method used is the Standard ICP point-to-point method [14]. The ICP algorithm can be described in two steps:

1) Searches for correspondences in the 3D space using a nearest neighbor approach.
2) Estimates a rigid transformation that minimizes the distance between the corresponding points.

The method iterates over these two steps to find the transformation that yields to a minimum distance or threshold.

A. The Pose Estimation Algorithm

As an initial offline step the FPFH descriptors of M are computed. The algorithmic steps for pose estimation are the

following:-

1) We compute the FPFH descriptors of cluster C_i.

2) We estimate a coarse rigid transformation, using the SAC-IA algorithm:

   a) Select n (3) random points from cluster C_i, whose pairwise distances are greater than minimum threshold d_{min} (2cm)

   b) A set of m (10) corresponding points from the model M, whose histograms are similar to the cluster point are chosen. For each cluster point a match is randomly selected from this set of similar points.

   c) A rigid transformation is computed between the sampled points from C_i and the corresponding points from M. An error metric that describes the quality of the transformation is computed.

   We iterate over this step with a user-defined number of iterations. The rigid transformation yielding the lowest error metric is chosen.

3) We use the rigid transformation computed in the last step as an initial guess to ICP.

4) The final rigid transformation is obtained.

We follow these steps for every cluster C_i extracted from P.

B. Experimental Results

As an initial offline step we have generated a 3D model of the metallic object M using the Focus Handheld Software provided with the laser scanner (Fig.5). Each extracted cluster is registered to this model to generate a rigid transformation matrix.
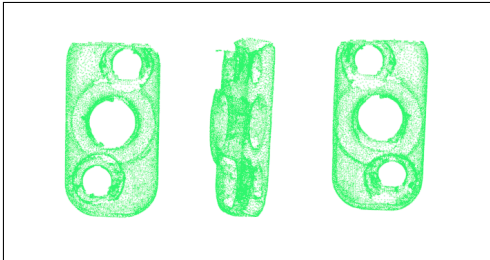

Figure 5: An example of metallic object 3d Model

We have identified one negative (Fig.7) and ten positive aligned clusters (Fig.6). The local FPFH descriptors of planar surfaces like the extracted clusters are very similar. The similarity between the descriptors may lead to align

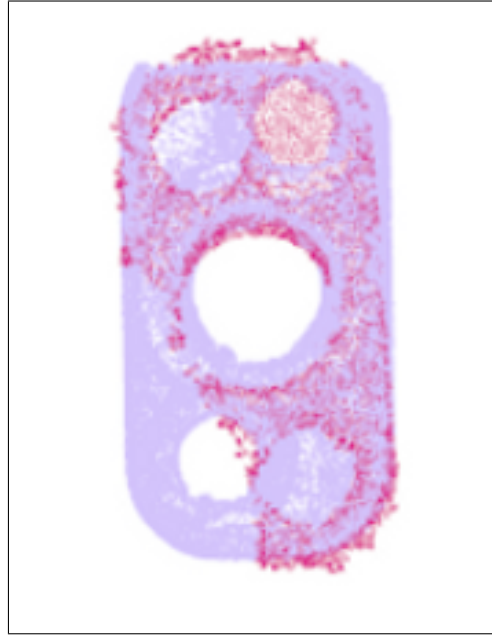Figure 6: Positive Aligned Clusters to the Model

Figure 7: Negative Aligned Clusters to the Model

the planar surfaces in opposite direction. This behavior was identified with occluded clusters and clusters with non-uniformly distributed points. The computed features from non-uniformly distributed points in the clusters may get into a local minima while aligning to the model. We have used Point Cloud Library (PCL) implementations of algorithms for this task. [1] [2].

VII. APPLICATION

We aimed the 3d metallic object recognition task for a bin picking application scenario. We plan to use KUKA Lightweight Robot (LWR 4) which is developed for a payload of 7 Kg and itself has a mass of 15 Kg [10]. It has a small 2 finger gripper which can pick the estimated metallic objects from a pile based on their pose (Fig.8). We

focus to scan the object as an offline activity and bring the segmentation and pose estimation task on-line. Additionally a ranking module is under-development for computing the rank of good aligned clusters based on the fitness score of the registration.
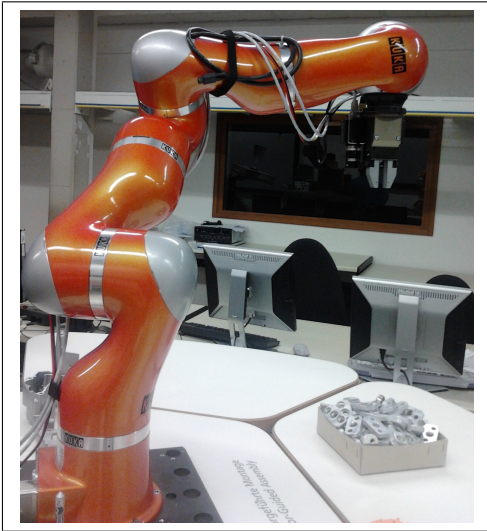

Figure 8: An example of the possible future application scenario

VIII. Conclusion

We have presented a simple method for planar object segmentation and pose estimation with quite promising results. As presented in the experimental results, most of the clusers are well aligned to the model. There exists very few flipped and negatively aligned clusters due to the non-uniform distribution of points. For these specific metallic objects laser scanner has been used, because it can deal with the reflectance of the metallic surfaces. This method is extend-able to other planar objects obtained by stereo or kinect sensors. We plan to work on real life non-metallic objects obtained by Kinect while they provide comparable planar surfaces. In the future work, we will use the estimated pose for a bin picking application.

Acknowledgment

References

[1] Gabriel Taubin, *Estimating The Tensor Of Curvature Of A Surface From A Polyhedral Approximation*, Proceedings of the Fifth International Conference on Computer Vision (ICCV), IEEE Computer Society, Washington, DC, USA, 1995.

[2] Adrian Hilton and John Illingworth, *Geometric fusion for a hand-held 3D sensor*, International Journal of Machine Vision and Applications, Vol(12), Num(1), PP(44-51), Springer Berlin / Heidelberg, Germany, 2000.

[3] Guil, N and Cózar, J. R. and Zapata, E. L., *Planar 3D Object Detection by Using the Generalized Hough Transform*, Proceedings of the 10th International Conference on Image Analysis and Processing, ICIAP '99, IEEE Computer Society, Washington, DC, USA, 1999.

[4] Simon, G.; Berger, M.-O., *Pose estimation for planar structures*, Proceedings of IEEE International Conference of Computer Graphics and Applications Vol(22), Num(6), PP(43-53), 2002.

[5] Carmen de Trazegnies and Cristina Urdiales and Antonio Bandera and Francisco Sandoval Hernández, *3D object recognition based on curvature information of planar views*, International Journal of Pattern Recognition, Vol(36), Num(11), PP(2571-2584), 2003.

[6] Joaquim Salvi and Carles Matabosch and David Fofi and Josep Forest, *A review of recent range image registration methods with accuracy evaluation*, International Journal of Image and Vision Computing, Vol(25), Num(5), PP(578-596), 2007.

[7] Ghita, Ovidiu and Whelan, Paul F. and Vernon, David and Mallon, John, *Pose estimation for objects with planar surfaces using eigenimage and range data analysis*, International Journal of Machine Vision Applications, Vol(18), Num(6), PP(355-365), 2007.

[8] Changhai Xu and Benjamin Kuipers and Aniket Murarka, *3D pose estimation for planes*, ICCV Workshop on 3D Representation for Recognition (3dRR-09), 2009.

[9] Iván Fernando Mondragón and Pascual Campoy and Carol Martínez and Miguel A. Olivares-Méndez, *3D pose estimation based on planar object tracking for UAVs control*, Proceedings of IEEE International Conference on Robotics and Automation (ICRA), PP(35-41), 2010.

[10] Rainer Bischoff and Johannes Kurth and Gnter Schreiber and Ralf Koeppe and Alin Albu-Schffer and Alexander Beyer and Oliver Eiberger and Sami Haddadin and Andreas Stemmer and Gerhard Grunwald and Gerhard Hirzinger, *The KUKA-DLR Lightweight Robot arm a new reference platform for robotics research and manufacturing*, International Symposium on Robotics (ISR2010), Munich, Germany, 2010.

[11] Radu Bogdan Rusu, *Semantic 3D Object Maps for Everyday Manipulation in Human Living Environments*, PhD Thesis, Computer Science Department, Technische Universitt Muenchen, Germany, October 2009.

[12] Radu Bogdan Rusu and Nico Blodow and Michael Beetz, *Fast Point Feature Histograms (FPFH) for 3D Registration*, Proceedings of IEEE International Conference on Robotics and Automation (ICRA), Kobe, Japan, May 2009.

[13] P.J. Besl and N.D McKay, *A Method for Registration of 3-D Shapes*, The IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), Vol(14), 1992.

[14] Zhengyou Zhang, *Iterative Point Matching for Registration of Free-Form Curves*, International Journal of Computer Vision, Vol(13), Num(2), PP(119-152), 1994.

[15] A. E. Johnson and M. Hebert. *Using spin images for efcient object recognition in cluttered 3d scenes*, The IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), Vol(21), Num(5), PP(433-449), 1999.

[16] D. Chung, Y.D.S. Lee, *Registration of multiple-range views using the reverse-calibration technique*, The IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), Vol(31), Num(4), PP(457-464), 1998.

[17] J. Tarel, H. Civi, D. Cooper, *Pose estimation of free-form 3d objects without point matching using algebraic surface models*, Proceedings of IEEE Workshop on Model-Based 3D, 1998.

[18] *C.J.R. Chua*, Point signatures: a new representation for 3d object recognition, International Journal of Computer Vision, Vol(25), Num(1), PP(63-65), 1997.

[19] *F. Tombari and S. Salti and L. Di Stefano*, Unique Signatures of Histograms for Local Surface Description, European Conference on Computer Vision (ECCV 2010), Crete, Greece, 2010.

[20] C.-S. Chen, Y.-P. Hung, J.-B. Cheng, *A fast automatic method for registration of partially overlapping range images*, International Conference on Computer Vision (ICCV), Bombay, India, 1998.

[21] J. Feldmar, N. Ayache, *Rigid, affine and locally affine registration of free-form surfaces*, Technical Report of INRIA, Sophia Antipolis, France, March, 1994.

[22] Y.Chen and G.Medioni, *Object modeling by registration of multiple range images*, IEEE International Conference on Robotics and Automation (ICRA), Sacramento, USA, 1991.

[23] Radu Bogdan Rusu and Steve Cousins, *3D is here: Point Cloud Library (PCL)*, Proceedings of IEEE International Conference on Robotics and Automation (ICRA), Shanghai,China, 2011.