Available online at www.sciencedirect.com

ScienceDirect

Procedia Engineering 100 (2015) 847 – 852

Procedia Engineering

www.elsevier.com/locate/procedia

25th DAAAM International Symposium on Intelligent Manufacturing and Automation, DAAAM 2014

Robot Assisted 3D Point Cloud Object Registration

Bojan Jerbić, Filip Šuligoj*, Marko Švaco, Bojan Šekoranja

University of Zagreb, Faculty of Mechanical Enigineering and Naval Arhitecture, Department of Robotics and Production System Automation, Ivana Lučića 5, 10000 Zageb, Croatia

Abstract

In this paper we describe a method for registration of 3D point clouds that represent objects of interest. A stereovision system is used to capture point clouds of a static environment, robot arm and an unknown object. By moving the robot arm in the environment the proposed system defines known occupied zones and is able to identify the robot arm. In order to identify a complete point cloud presentation of the robot gripper it is rotated in front of a stereovision camera and its geometry is captured from different angles. Iterative closest point algorithm is used to determine a rigid transformation between every new robot pose so the original point cloud can be appended with the transformed one. When the robot is holding a new object the registration procedure is repeated and known elements (environment, robot arm and gripper) are removed so that the object can be identified.

Peer-review under responsibility of DAAAM International Vienna

Keywords: Point cloud data; position estimation; object recognition; machine learning; stereovision

1. Introduction

The ability to identify and manipulate with objects in its environment is a crucial task for robots used in a wide variety of applications that assume unstructured working environment. In recent years depth cameras became a popular and accessible tool for capturing information about the robot environment. Range data can be acquired using a structured light camera [1], stereovision camera [2], LIDAR device [3] etc. Fore mentioned devices all generate discrete range measurements or point clouds from a fixed perspective. If a robot is working in an unstructured or dynamic environment the point cloud data can be used to detect objects and their position. When the position of a depth camera in a robot frame is known and the object pose is determined the robot can proceed to perform handling

* Corresponding author. Tel.: +385-98-1614645; fax: +385-1-6156-940.
E-mail address: filip.suligoj@fsb.hr

Peer-review under responsibility of DAAAM International Vienna
doi:10.1016/j.proeng.2015.01.440

operations. Benefits of using point cloud data in robotic applications are numerous so implementations span from scientific and household to industrial. Current advances in 3D data acquisition and processing for industrial applications and potential research opportunities are covered in [4]. Article on industrial application of point clouds [5] describes an automatic system that uses robot motion control and a laser scanner for the purpose of reverse engineering. In the emerging field of service robotics 3D point clouds are mostly used for creating maps of household environments [6] and for reliable object grasping and manipulation [7]. An efficient tool for 3D point cloud processing comes in a form of c++ library called Point cloud library (PCL) [8]. To solve a problem of object recognition and six degree of freedom pose estimation from a point cloud several methods are available and presented in [9]. Matching an object to a scene requires a known object template. Object template can be acquired from a CAD model and transformed to a point cloud or it can be registered from a 3D scanning system. Real time free-hand scanning system used for 3D model reconstruction is described in [10]. The system registers and incorporates point clouds of an object and removes scanning errors. There are two methods for solving the registration problem: Feature based registration and Iterative closest point algorithm (ICP). Feature based registration includes usage of keypoints and feature descriptors when estimating correspondence between two point clouds. Keypoints are interest points derived from a set of points that best describe the scene. Feature descriptors are computed from each keypoint. Popular feature histograms are Persistent point feature histogram [11], Fast point feature histogram [12], Viewpoint feature histogram [13] and Viewpoint oriented color-shape histogram [14]. Iterative closest point method iterates steps: search for correspondences, reject bad correspondences and estimate a transformation using the good correspondences.

Previous work of the authors described in [15] locates distinct markers in a 2D image and uses stereovision camera to extract 3-D coordinates from markers mounted on the robot tool and the object. This makes the setup very flexible because no predetermined robot position is needed for object tracking and pick and place applications. In this paper object and robot gripper registration is added so that previously developed system has no need for mounted markers. Such addition further enhances simplicity and autonomy of the system. Similar approach is used in [16] where autonomous robot system learns visual models of symmetric objects by rotating them 360 degrees and collecting 2D training images. Instead of using 2D images, in this paper we describe a registration of a dense 3D point cloud presenting the robot arm or any object of interest. Coordinated movement of the robot coupled with a registration procedure carried out by the stereovision camera enables creation of training models captured from different angles, concatenated and saved as point cloud templates. When template registration is completed stereovision camera can use point cloud of the scene taken from a single viewpoint to recognize the robot gripper and the object templates and estimate its pose.

2. System setup

Main components of the proposed system shown in Fig. 1. are: Fanuc LR Mate 200iC 5L robot arm, Bumblebee XB3 stereovision camera system and a personal computer (PC). Communication between the robot and the PC is handled using socket communication and developed client-server programs.

Multi-thread program in c++ uses the following software packages:

- Flycapture and Triclops from Point grey research company are used to manage point cloud capture from Bumblebee XB3 stereovision device
- Point cloud library (PCL) - a C++ library for 3D point cloud processing. Processing algorithms operate on 3D point cloud data for the purpose of: filtering, feature estimation, surface reconstruction, model fitting, segmentation and registration

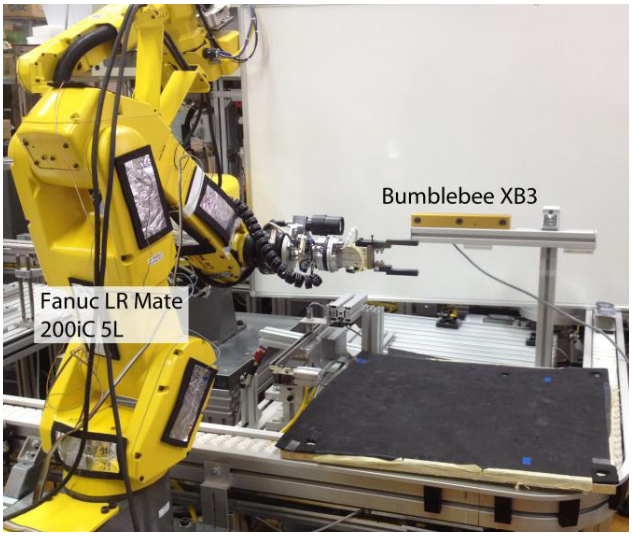

Fig. 1. System overview.

3. Model template registration

To capture a set of points that sufficiently covers 3D geometry of an object camera viewpoint or the object, it needs to be scanned from different viewpoints. The proposed approach uses a robot arm to rotate the object and capture point clouds from different perspectives. Transformation matrix combining two consecutive point clouds is obtained using the ICP algorithm. Two point clouds are concatenated and compared with the next one repeatedly until the complete object template geometry is acquired. Template registration is preformed separately for the robot gripper and the object of interest.

The following list describes application prerequisites:

- Object size and position must correspond with the available range of the stereovision camera
- No other objects can be present in designated area during the registration process
- Coordinate system alignment between the robot and the stereovision device can be used to eliminate wrong readings (see Table 1.)
- Objects used for registration must be rigid because the ICP algorithm doesn't work with deformable objects
- Gripper template must be captured without the robot joints as that would imply a deformable object
- Objects must have certain texture because of stereo correspondence necessary for determining 3D points from two camera images

3.1. Robot gripper registration

The procedure for the robot gripper registration is as follows:

1. The robot gripper is moved inside the registration area of the stereovision camera.
2. A confirmation signal is sent and a point cloud of the gripper is captured.
3. The robot rotates its last joint for a predefined angle α and sends a confirmation signal.
4. Consecutive point clouds are compared using the ICP algorithm. An iterative procedure repeats the transformation to minimize the distance between the points of two point clouds until matching criterion has been met. Matching criterion is adjusted depending on the data noise and the precision requirement. Fig. 2. shows two point clouds of the robot gripper rotated for 10 degrees. The result is a transformation matrix mapping every point P from cloud C_i to cloud C_{i+1}:

$$P_{i+1} = T \cdot P_i \quad (1)$$

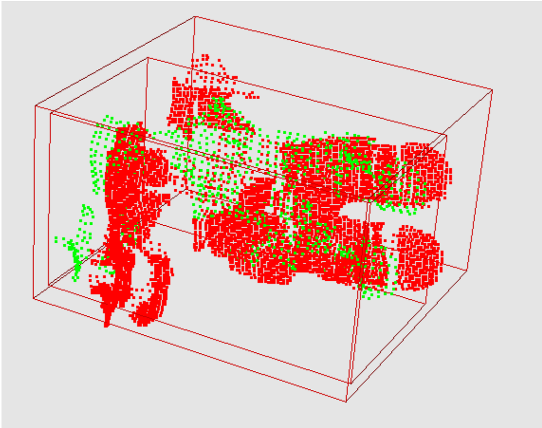

Fig. 2. Registering two point clouds (green and red).

5. The transformation matrix obtained with the ICP algorithm is compared and checked with the known rotation of the robot gripper. If the transformation estimation differs from expected values due to noisy data, the procedure is repeated or the rotation angle is reduced. Results from transformation evaluation using the ICP algorithm are showed in Table 1. The robot gripper was incrementally rotated for 10 degrees around an axis parallel with the x-axis of the stereovision camera and consecutive point clouds were matched. Values from transformation matrix were transformed to Euler angles and translation vector.

Table 1. Transformation evaluation using ICP.

Robot gripper rotation		Measured ICP transformation values					
α_1 (deg)	α_2 (deg)	Roll (deg)	Pitch (deg)	Yaw (deg)	X-axis (m)	Y-axis (m)	Z-axis (m)
0	10	9.088	0.073	3.621	-0.0156	0.0932	0.0277
10	20	9.605	-0.503	0.015	0.0037	0.1008	0.0304
20	30	10.123	1.132	0.062	-0.0134	0.1046	0.0312
30	40	10.211	2.144	-3.502	-0.0075	0.1018	0.0287
40	50	8.984	3.727	0.801	-0.0391	0.0891	0.0282
50	60	24.299	1.291	0.219	-0.0094	0.2401	0.1058
60	70	11.848	-0.335	5.321	-0.0222	0.1288	0.0415
70	80	5.102	1.411	2.916	-0.0237	0.0533	0.0119
80	90	8.642	1.268	0.037	-0.0111	0.0882	0.0253

6. If transformation estimation is satisfactory, point cloud C_{i+1} is concatenated with transformed C_i.

$$C_{i+1} = C_{i+1} + T \cdot C_i \quad (2)$$

7. The procedure from step 2. to 6. is repeated n-times with n being n=360/ α.
The result is a complete dense point cloud of the robot gripper.

3.2. Object registration

Object registration is user assisted since a human operator needs to place an object into the robot gripper. Object registration is conducted after the robot gripper registration procedure. The working procedure of object registration follows the same steps as the robot gripper registration with the exception of a differentiating step used after the point cloud capture. The differentiating step first matches a gripper template inside a scene using the ICP algorithm. A cuboid containing all the points of the gripper template is created. The scene is then filtered from all the points inside the cuboid. The remaining points are considered to belong to the object. Two cuboids enclosing the robot gripper and the object are shown in Fig. 3.

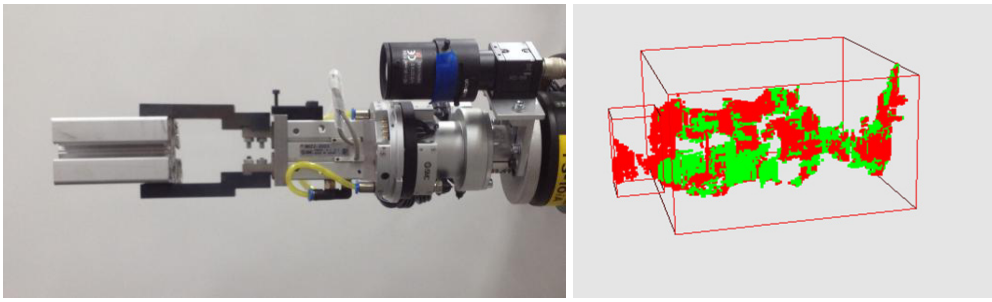

Fig. 3. Robot gripper and the object (left); Separated robot gripper from the held object in a point cloud (right).

Conclusion

In this paper an automatic point cloud registration of the robot gripper and the object of interest is demonstrated. The model templates are generated by using the robot arm with the stereovision camera to capture point clouds of the object from different viewpoints. The system is capable of self-evaluation when comparing known robot rotations to the ones estimated with the ICP. The differentiating step in the object registration procedure enables the system to differentiate object and gripper templates even when they are connected. The result is an automatic 3D object registration and recognition system.

Further work should include usage of voxel filters to remove noise from the data and feature descriptor registration to improve precision and time consumption of computations. A module for grasp planning will be implemented for autonomous grasping of localized objects. Implementation of algorithms for adapting to unstructured working environment will be based on a previously developed framework [17]. Grasp planning module will use the relative transformation of the gripper coordinate frame and the object coordinate frame to match the position learned when performing registration procedure.

Acknowledgements

Authors would like to acknowledge the support of Croatian Scientific Foundation through the research project ACRON - A new concept of Applied Cognitive RObotics in clinical Neuroscience.

References

[1] B. Freedman, A. Shpunt, M. Machline, and Y. Arieli, *Depth mapping using projected patterns*. Google Patents, 2012.

[2] S. M. Choi, E. G. Lim, J. I. Cho, and D. H. Hwang, *Stereo vision system and stereo vision processing method*. Google Patents, 2012.

[3] P. E. Bauhahn, B. S. Fritz, and B. C. Krafthefer, *Systems and methods for safe laser imaging, detection and ranging (lidar) operation*. Google Patents, 2009.

[4] Z. M. Bi and L. Wang, "Review: Advances in 3D Data Acquisition and Processing for Industrial Applications," *Robot Comput-Integr Manuf*, vol. 26, no. 5, pp. 403–413, Oct. 2010.

[5] S. Larsson and J. A. P. Kjellander, "Motion control and data capturing for laser scanning with an industrial robot," *Robot. Auton. Syst.*, vol. 54, no. 6, pp. 453 – 460, 2006.

[6] R. B. Rusu, Z. C. Marton, N. Blodow, M. Dolha, and M. Beetz, "Towards 3D Point cloud based object maps for household environments," *Robot. Auton. Syst.*, vol. 56, no. 11, pp. 927–941, Nov. 2008.

[7] M. Ciocarlie, K. Hsiao, E. G. Jones, S. Chitta, R. B. Rusu, and I. A. Şucan, "Towards reliable grasping and manipulation in household environments," in *Experimental Robotics*, 2014, pp. 241–252.

[8] R. B. Rusu and S. Cousins, "3d is here: Point cloud library (pcl)," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, 2011, pp. 1–4.

[9] A. Aldoma, Z.-C. Marton, F. Tombari, W. Wohlkinger, C. Potthast, B. Zeisl, R. Rusu, S. Gedikli, and M. Vincze, "Tutorial: Point Cloud Library: Three-Dimensional Object Recognition and 6 DOF Pose Estimation," *IEEE Robot. Autom. Mag.*, vol. 19, no. 3, pp. 80–91, Sep. 2012.

[10] T. Weise, T. Wismer, B. Leibe, and L. Van Gool, "In-hand scanning with online loop closure," in *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, 2009, pp. 1630–1637.

[11] R. B. Rusu, N. Blodow, Z. C. Marton, and M. Beetz, "Aligning point cloud views using persistent feature histograms," in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, 2008, pp. 3384–3391.

[12] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, 2009, pp. 3212–3217.

[13] R. B. Rusu, G. Bradski, R. Thibaux, and J. Hsu, "Fast 3d recognition and pose using the viewpoint feature histogram," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, 2010, pp. 2155–2162.

[14] W. Wang, L. Chen, D. Chen, S. Li, and K. Kuhnlenz, "Fast object recognition and 6d pose estimation using viewpoint oriented color-shape histogram," in *Multimedia and Expo (ICME), 2013 IEEE International Conference on*, 2013, pp. 1–6.

[15] F. Šuligoj, B. Šekoranja, M. Švaco, and B. Jerbić, "Object Tracking with a Multiagent Robot System and a Stereo Vision Camera," *Procedia Eng.*, vol. 69, pp. 968–973, 2014.

[16] W. H. Li and L. Kleeman, "Interactive learning of visually symmetric objects," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, 2009, pp. 4751–4756.

[17] M. Švaco, B. Šekoranja, and B. Jerbić, "Industrial Robotic System with Adaptive Control," *Procedia Comput. Sci.*, vol. 12, pp. 164–169, Jan. 2012.