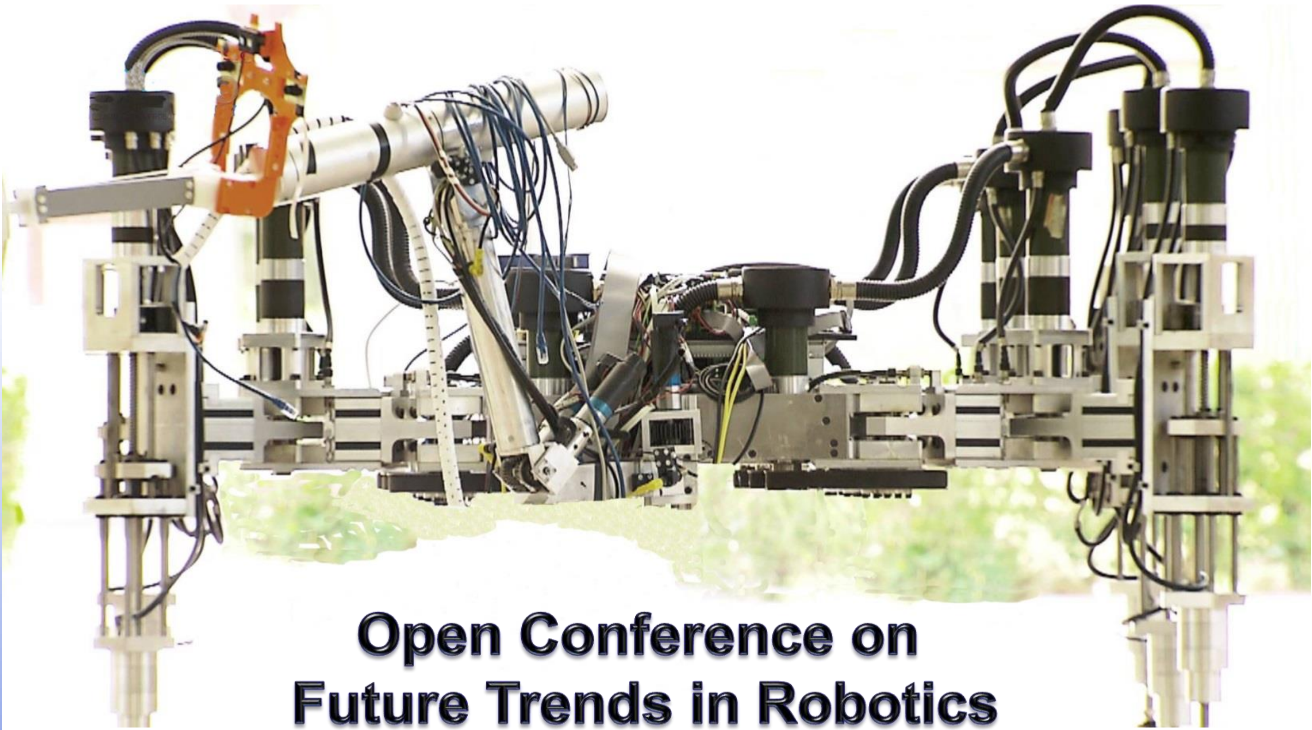
RoboCity16

Robots for citizens



Open Conference on Future Trends in Robotics

Edited by:
Roemi E. Fernández
Héctor Montes

RoboCity2030.org

Comunidad de Madrid

UNIÓN EUROPEA
Fondos Estructurales
Invertimos en su futuro

Universidad Carlos III de Madrid · CSIC Consejo Superior de Investigaciones Científicas · Politécnica · Universidad de Alcalá · Universidad Rey Juan Carlos · UNED

May 2016

RoboCity16 Open Conference on Future Trends in Robotics

Editors
Roemi E. Fernández Saavedra
Héctor Montes Franceschi

Madrid, 26 May 2016

Edited by: Consejo Superior de Investigaciones Científicas
Printed by: AFERTA SG S.L.
Legal Deposit: M-17804-2016
ISBN: 978-84-608-8452-1

CHAPTER 14

3D OBJECT RECOGNITION AND POSE ESTIMATION USING VFH DESCRIPTORS

A. LÁZARO, J. MORIANO, L.M. BERGASA, R. BAREA and E. LOPEZ

Robesafe group, Universidad de Alcalá, alberto2991lazaro@gmail.com

This paper presents the perception system involved in an application for grabbing objects in an industrial environment through a robotic arm by using 3D pointcloud images taken from a RGBD camera. This system recognizes and estimates the pose of the objects placed on the working area. It uses VFH descriptors and Kd-tree classifiers for the recognition and geometrical models and a RANSAC estimator for the pose estimation using this information. Some experimental results in a real environment validate our proposal.

1 Introduction

This work is focused on object recognition and pose estimation in industrial environments using a RGBD camera. This vision system is able to detect objects, recognize them and estimate their position offering the possibility of detecting important features which can be used to automate processes. This target can be achieved using RGB cameras but they have some drawbacks faced with depth cameras which will be explained hereafter. The stages that have to be completed by the vision system to provide the necessary data are segmentation, recognition, pose estimation and scene reconstruction. Segmentation identifies the interested objects to be grabbed. Once these objects are segmented they pass to the recognition algorithm. Shape estimator is used to recognize and obtain the position and orientation of the objects. Taking advantage of ROS communication facilities, the obtained data are published to be reconstructed thanks to RVIZ as it can be seen in Fig. 1.

Fig. 1. Captured environment and virtual reconstruction.

2 State of the art

Regarding to the topic of automatic objects classification, several projects have been carried out. Hereafter we show some of the most important. In (Bdiwi, 2012) a more complex classification process is developed, in this case the system is able to difference among books depending on their shapes and using alphanumeric codes. They use a CCD camera, it uses a rectangle to compare the different shapes. Another example for the object classification is included in (Li, 2014) using a depth sensor. In this work, a multilevel part-based object model was proposed using latent support vector machine as a core learning machine for training.

Our proposal, as difference with the above works, uses a depth camera for obtaining 3D Point Cloud images (Cruz,2012), (Rusu,2011) and then to recognize the shape of the different objects. It provides several advantages, the main ones are the use of an RGB-D camera and a better 3D reconstruction of the scene.

3 Object recognition method

The vision system must detect the objects placed in the working scene. So 3D pointcloud is clustered in foreground and background classes using depth. After that, foreground class is downsampled using Uniform Sampling algorithm named VoxelGrid filter before being clustered in objects.

The aim of clustering the foreground data is to divide an unorganized PCL into smaller parts, which contain the points that belongs to a same object (Muja, 2009). This clustering has been done relying on spatial decomposition techniques that find boundaries and subdivisions. The algorithm used is based on Euclidean distances.

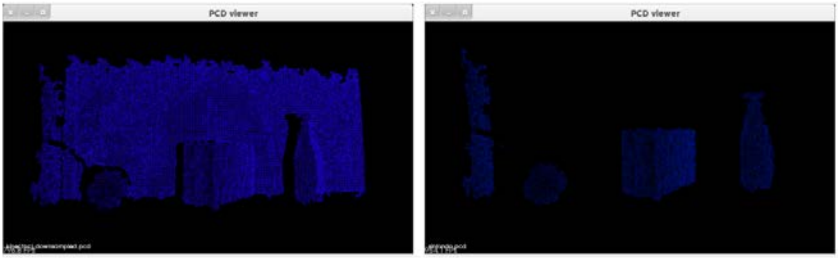


Fig. 2. Background extraction example: in the left image the original capture.

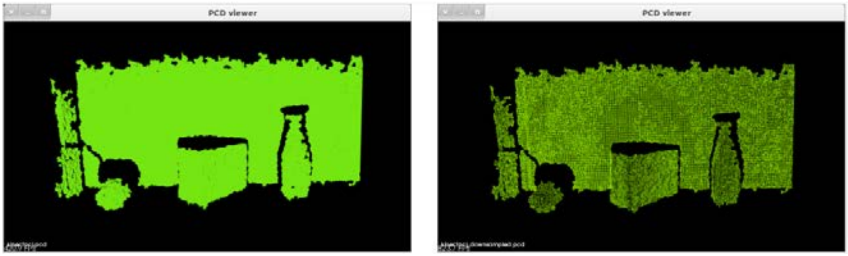


Fig. 3. Downsampling the captured scene example: the PCL depicted on the right.

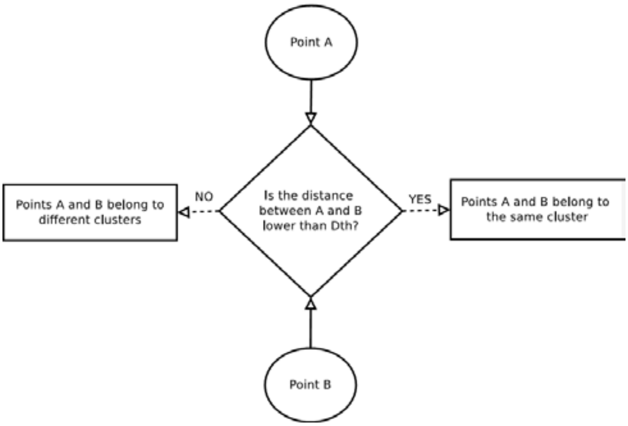


Fig. 4. Cluster extraction diagram: the algorithm works out the distance between two points, depending on this value the evaluated points are considered as part of the same cluster or as part of two different clusters according to the diagram.

Once the scene is clustered, is time to tackle the object recognition of the different objects placed in the scene (Rusu, 2010; Rusu, 2009). To carry out this recognition VFH descriptors, which is based on the FPFH descriptor, are used (Rusu, 2009; Rusu, 2008). This recognition consists in two processes:

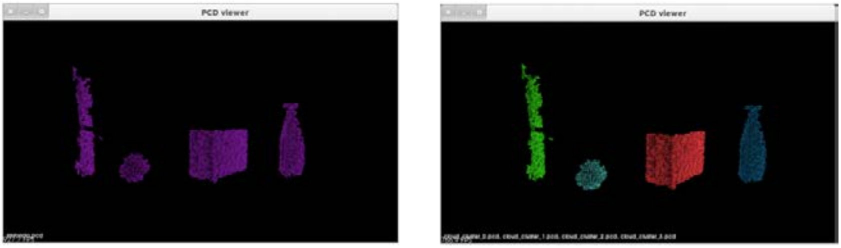


Fig. 5. Cluster extraction example: in the right image the points considered as part of the same cluster are represented in the same color.

3.1 Training process

The first step is to build a dataset that contains some captures (36 in our case) taken from different points of view for each object that is wanted to be recognized by the system and to calculate their VFH descriptor. This dataset is used to train the object recognition system. The training process builds a space-partitioning data structure that stores a set of K-dimensional points in a tree structure that enables efficient range searches and nearest neighbor searches called Kd-tree (Rusu, 2008), (Salti, 2011).

Fig. 6. Objects employed to build the dataset. Each object represents a sample of the three classes needed in this work: ball can and milk (from left to right).

3.2 Testing process

This process compares VFH descriptors for each of the segmented objects extracted from the captured scene and the ones that are saved in the training dataset through K-Nearest neighbors algorithm(k-NN) (Muja, 2011). The result of this comparison is a set of index which values are inversely proportional to the likeness between the analyzed cluster and each element of the dataset (Bariya, 2010). These index are provided to a voting method to determine the class among the trained classes which belongs to the unknown object if the analyzed object belongs to one of the trained classes. If so, the voting system classifies it.

Fig. 7. Example of the testing process: the input used in this working example is one of the PCLs of the dataset.

4 Pose estimation method

Once each of the elements placed in the working scene are recognized, the coefficients that determine their position and orientation are recovered by using RANdom SAmple Consensus (RANSAC). RANSAC algorithm is usually employed to carry out in objects recognition approaches when the set of objects to be distinguished have different shapes (Schnabel, 2007), (Papazov, 2010), but in this work it has been used just for estimating the pose of the objects and their dimensions.

Three models (sphere, cylinder, plane) jointly with their RANSAC estimators are used to estimate the parameters that defines the pose and the dimensions of the different objects (balls, cans, milk cartons) (Aldoma, 2012), (Mishra,2011). To estimate the dimensions of the carton box once the different planes have been estimated geometric constraints are applied.

5 Perception system experiments

The following tests have been carried out to measure the power of the designed vision system evaluating different aspects such as time costs or percentage success. Hereafter we present the different tests carried out to evaluate our vision system performance regarding the processing time an object recognition results. The first experiment tests the performance of our objects recognition system for 5 different objects of each class not used in the training process. As it can be seen in Table 1 the correct recognition ratio is close to 100%.

Table 1. Object recognition results: index shown are the returned by the system.

Object/Candidate	Ball	Can	Milk carton
Ball	100%	0%	0%
Can	1%	95%	4%
Milk carton	1%	5%	94%

The object recognition system runs in a computer with Intel Core i7 CPU 860 @ 2.8GHz x8} processor. Table 2 evaluates the different processing time depending on the recognized elements

Table 2. Object recognition processing times.

Object	No. balls	No. cans	No. Cartons	Time costs
Test 1	1	0	0	190ms
Test 2	2	0	0	275ms
Test 3	3	0	0	410ms
Test 4	0	1	0	345ms
Test 5	0	2	0	650ms
Test 6	0	0	1	890ms
Test 7	1	1	0	515ms
Test 8	1	1	1	1345ms

Fig. 7 shows the results extracted from all tests to determine the dimension, position and orientation errors through the mean and the deviation for the different primitive shapes measured in meters.

6 Conclusions and future work

This paper has addressed the object recognition, pose estimation and dimensions appraisal. Due to the real time constraints geometric restrictions, VFH descriptors and a Kd-tree classifier have been applied over the 3D point cloud images. Object pose estimations are precise enough for a robot to interact with them. The vision system algorithm could provide a robotic system the information needed to move or avoid objects. Although we have presented a real time solution for the addressed problem we will go on working on reducing the processing time in order to make our system able to recognize objects faster than a human.

Acknowledgements

This work was supported in part by the MINECO SmartElderlyCar (TRA2015-70501-C2-1-R) project and by the RoboCity2030 III-CM project (S2013/MIT-2748) funded by Programas de I+D en la Comunidad de Madrid and cofunded by Structured Funds of the UE.

References

Aldoma, A., Marton, Z.-C., Tombari, F., Wohlkinger, W., Potthast, C., Zeisl, B., Rusu, R.B., Gedikli, S., Vincze, M., 2012. Tutorial: Point Cloud Library: Three-Dimensional Object Recognition and 6 DOF Pose Estimation. IEEE Robot. Automat. Mag.

Bariya, P., & Nishino, K. (2010, June). Scale-hierarchical 3d object recognition in cluttered scenes. In Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on (pp. 1657-1664). IEEE.

Bdiwi, M., & Suchý, J. 2012. Robot control system with integrated vision/force feedback for automated sorting system. In Tech. for Practical Robot Applications (TePRA), 2012 IEEE Int. Conf. on (pp. 31-36). IEEE.

Cruz, L., Lucio, D., & Velho, L. 2012. Kinect and rgbd images: Challenges and applications. In Graphics, Patterns and Images Tutorials (SIBGRAPI-T), 2012 25th SIBGRAPI Conference on (pp. 36-49). IEEE.

Li, K., & Meng, M. 2014. Robotic object manipulation with multilevel part-based model in RGB-D data. In Robotics and Automation (ICRA), 2014 IEEE International Conference on (pp. 3151-3156). IEEE.

Mishra, A. K., & Aloimonos, Y. (2012). Visual segmentation of “simple” objects for robots. Robotics: Science and Systems VII, 1-8.

Muja, M., & Lowe, D. G. 2009. Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration. VISAPP (1),2, 331-340.

Muja, M., Rusu, R. B., Bradski, G., & Lowe, D. G. (2011, May). Rein-a fast, robust, scalable recognition infrastructure. In Robotics and Automation (ICRA), 2011 IEEE Intern.l Conference on (pp. 2939-2946). IEEE.

Papazov, C., & Burschka, D. (2010). An efficient RANSAC for 3D object recognition in noisy and occluded scenes. In Computer Vision–ACCV 2010 (pp. 135-148). Springer Berlin Heidelberg.

Rusu, R. B., Blodow, N., & Beetz, M. 2009. Fast point feature histograms (FPFH) for 3D registration. In Robotics and Automation, 2009. ICRA'09. IEEE International Conference on (pp. 3212-3217). IEEE.

Rusu, R. B., Bradski, G., Thibaux, R., & Hsu, J. 2010. Fast 3d recognition and pose using the viewpoint feature histogram. In Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ Intern. Conf. on (pp. 2155-2162). IEEE.

Rusu, R. B., Holzbach, A., Beetz, M., & Bradski, G. 2009. Detecting and segmenting objects for mobile manipulation. In Computer Vision Workshops (ICCV WS), 2009 IEEE 12th Int. Conf. on (pp. 47-54). IEEE.

Rusu, R. B., Marton, Z. C., Blodow, N., & Beetz, M. 2008. Learning informative point classes for the acquisition of object model maps. InControl, Automation, Robotics and Vision, 2008. ICARCV 2008. 10th International Conference on (pp. 643-650). IEEE.

Rusu, R. B., Marton, Z. C., Blodow, N., & Beetz, M. 2008. Persistent point feature histograms for 3D point clouds. In Proc 10th Int Conf Intel Autonomous Syst (IAS-10), Baden-Baden, Germany (pp. 119-128).

Rusu, Radu Bogdan and Cousins, Steve. 2011. 3D is here: Point Cloud Library (PCL). ICRA.

Salti, S., Tombari, F., & Stefano, L. D. 2011. A performance evaluation of 3d keypoint detectors. In 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), 2011 Intern. Conf. on (pp. 236-243). IEEE.

Schnabel, R., Wahl, R., & Klein, R. (2007, June). Efficient RANSAC for point-cloud shape detection. In Computer graphics forum (Vol. 26, No. 2, pp. 214-226). Blackwell Publishing Ltd.