



HAL
open science

Indoor Home Furniture Detection with RGB-D Data for Service Robots

Oscar Alonso-Ramirez, Antonio Marin-Hernandez, Michel Devy, Fernando M Montes-Gonzalez

► **To cite this version:**

Oscar Alonso-Ramirez, Antonio Marin-Hernandez, Michel Devy, Fernando M Montes-Gonzalez. Indoor Home Furniture Detection with RGB-D Data for Service Robots. International Conference on Electronics, Communications and Computers (CONIELECOMP), Feb 2014, Puebla, Mexico. 10.1109/CONIELECOMP.2014.6808586 . hal-02049147

HAL Id: hal-02049147

<https://hal.science/hal-02049147v1>

Submitted on 26 Feb 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Indoor Home Furniture Detection with RGB-D Data for Service Robots

Oscar Alonso-Ramirez¹, Antonio Marin-Hernandez¹, Michel Devy² and Fernando M. Montes-Gonzalez¹.

Abstract—Home furniture detection is a very important topic enabling a robot to provide useful services at home. This paper presents an algorithm to identify and detect home furnitures by an autonomous service robot. The furniture considered in this paper includes large objects (e.g. beds, sofas, etc.) that can be moved by humans or by the robot on common tasks. 3D data acquired from an RGB-D camera mounted on the robot are analyzed to find discriminant features that characterize the pieces of furniture to be detected. The proposed methodology avoids the processing of the complete frame by the use of a small set of random points. These points are learned and classified in function of several attributes: color, 3D position and 3D normals. A function of random region growing and partial 3D modeling is then applied to validate the detection of a specific piece of furniture regarding the set of known furniture models. The process runs in real-time and can be easily incorporated to service robots.

I. INTRODUCTION

Over the last years, mobile robotics research has provided many successful applications. Thanks to the very fast improvements in robotics, it is expected that robots can soon provide useful services at homes. However, as homes are very dynamic environments, it is important to provide robots with efficient detection and localization algorithms to realize their tasks in a very robust way.

Nowadays, there are successful algorithms for: the localization and mapping (SLAM), planning collision avoidance trajectories and to recognize and track people. In advanced platforms as the PR2 robot (Figure 1), many of this tasks are already integrated. Therefore, many research institutes have begun to look at more complex problems dealing with a high level cognitive knowledge representation. To successfully perform more complex tasks, a robot must understand the usefulness of things and their spatial relationships.

The detection of pieces of furniture and the estimation of their current location at home are very relevant for many tasks that a robot can do. Nevertheless, a problem to which mobile robots have to face is the changes that humans do; i.e. humans change the furniture around, sometimes they change the bed spreads and place objects on top of the furniture. Therefore, it is necessary that a robot can detect the diverse pieces of furniture, according to individual discriminant characteristics.

In this paper, we present a methodology to detect home furniture that can be moved by humans or by the robot while



Fig. 1: The PR2 robot used in a home like environment

doing its tasks; therefore it is not suitable to integrate such pieces of furniture in commonly used maps (e.g. occupancy grids). Some furniture like chairs or sofas, are subject to bigger and more common displacements than for example beds. We have included beds and other similar furnitures in where movements are more rare but are still feasible due to some tasks like cleaning.

The proposed methodology uses 3D data obtained from a Kinect like sensor aboard the robot. Among all pieces of furniture commonly available in homes, we have initially selected the following three types: couches (or sofas), chests of drawers and beds. These large objects have different characteristics that can be used to discriminate them.

This paper is organized as follow; next section present some related works. In section III, are analyzed the different characteristics that will be used to detect the selected furniture. Section IV explains the proposed methodology to cope with the furniture detection, and in section V are presented the test and results. Finally section VI gives conclusions and future works.

II. RELATED WORK

Recently, 3D processing has acquired relevance to solve problems such as object recognition, scene modeling, maps construction and localization. This 3D modeling boom was largely due, to the sudden burst of the Kinect sensor in late 2010. This sensor has become very popular despite its lower accuracy, compared to other sensors like Time of Flight cameras (ToF), first due to its low cost and then, because the Kinect provides both depth and appearance data on the observed scene. Another tool that has helped

¹ Authors are with the Department of Artificial Intelligence, Universidad Veracruzana, Sebastián Camacho No. 5, Xalapa, Ver., 91000, Mexico. oscalra_820 at hotmail.com, [anmarin, fmontes] at uv.mx

²M. Devy is with CNRS; LAAS; Université de Toulouse; 7 avenue du Colonel Roche, F-31077 Toulouse Cedex, France. devy at laas.fr



(a) The living room

(b) The dining room

(c) The bedroom

Fig. 2: Different scenes of the environment

in the development of 3D applications, is the library `pcl` [1], which is a library for point clouds processing in 2D and 3D and which contains several algorithms for filtering, reconstruction, segmentation, etc.

Since the utilization of 3D information without any kind of previous processing is computationally very expensive, there have been emerging works which use geometric elements that are easy to recognize and provide enough information to work with, for example: the planes. Biswas et al. in [2] detect planes in a scene with a depth camera; to later merge those planes and obtain convex polygons which can help to obtain a representation of the environment. In [3] an approach to obtain planes with a laser scanner, instead of a depth camera, is shown. In this case, line segments are detected in order to comprise the planes.

The object detection has also profited by the use of 3D sensors. In [4] some 3D points descriptors are evaluated, to probe their functionality for classifying surfaces' primitives such as cylinders, edges or corners. Choi and Christensen in [5], use the color and depth information from a RGB-D camera to improve the object detection and determine its position on the environment. In [6], it is presented an approach to detect and track objects in 3D from edge detection of objects without texture. Ruhnke et al. [7] propose a technique to learn the objects' models from a 3D point cloud of partial views of the objects.

For the semantic representation of the environment some approaches merge information from both 2D and 3D sensors, for example in [8], where it is combined a laser and a 3D point cloud to obtain a semantic map using SLAM techniques. 3D information is used to detect planes with a RANSAC algorithm, the main planes are detected and small clusters are deleted in order to separate different regions. The 2D information is used to detect lines in analogous as for plane detection. Both results are merged to create the map. The different scales used for the sensors are used to enrich the map's information.

In [9] a robust SLAM algorithm is proposed using a RGB-D camera. Stucker et al. [10] also presents an algorithm which uses 3D information and a SLAM technique to obtain a semantic map, but in this case it performs a classification of the objects found in the scene using a decision forest. In [11] it is also performed a semantic representation of the scene based on objects, although it does not perform

any kind of assumption of the type of object. In this case only the objects that are above the main planes are detected by clustering techniques. The objects are only tagged with information about its size or color in order to identify them in the scene and detect any change of place.

There have been a few works about furniture detection. Like the one done by Wünnel and Moratz [12], which focus on an office environment to detect chairs and tables. This approach is based on geometric properties, uses 3D information from a SICK laser mounted on a pan-tilt platform. The space is divided in three horizontal sections of interest, depending its position from the ground. The object models are done with multiple segments with properties like height or size, organized on a directed acyclic graph. The graph can be iterated looking the main characteristics of a certain object and searching a match with the object segments found by the sensor.

In [13], a segmentation algorithm to detect objects of interest (chairs in this case) based on color and depth similarity is proposed, using the information provided by a Kinect sensor.

Rusu et al. has developed many methods about recognition the furniture on a kitchen. In [14], it is proposed an algorithm for the construction of object models from a 3D point cloud, for the construction of a semantic map which could deliver useful information to the robots for manipulation tasks. For the detection of the furniture, it is assumed the existence of large planes which later are divided on the different furniture through an edge detection. The rest of the objects that does not adapt to the previous description, for example a chair, are simply considered as obstacles.

In [15] the previous work is expanded to generate an hybrid map containing the 3D position of the objects which are in a fixed position in the environment, and it is combined with a surface to avoid collisions, calculated from the navigation and manipulation tasks.

These works show the relevance of the 3D information for a wide range of applications and the importance of the use of different techniques to handle this information efficiently. It is also shown the viability of the use of these techniques to detect and label furniture.

III. FURNITURE FEATURES DESCRIPTION

Detection and classification of pieces of furniture can provide useful information to enable a cognitive localization.

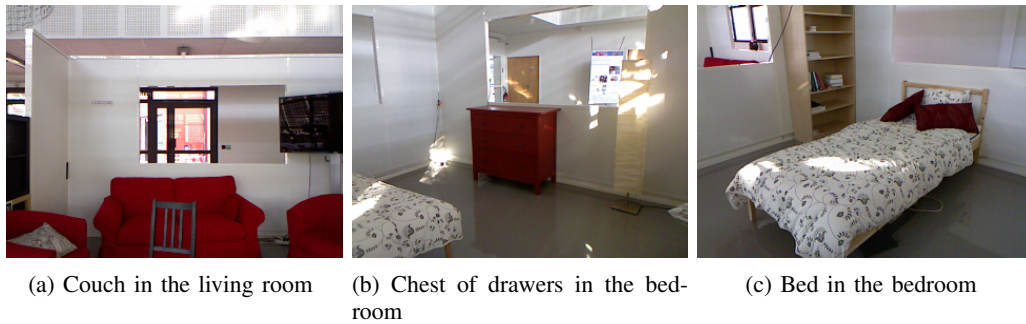


Fig. 3: Pieces of furniture selected for detection and classification

Furniture recognized can provide useful semantic information since, as show in Fig. 2, humans can infer the kind of room or place, in function of the furniture contained in the scene.

As it has been mentioned, it is expected to detect pieces of furniture that due to its own nature humans can displace in the environment, therefore its place can change between different robots executions or even during the execution of a task. It is important to detect not only the existence or not of a given piece of furniture in the scene, but also its position relative to other furniture. Thus the use of 3D sensors as RGB-D cameras (Kinect like) can ease the task, as it provides a point cloud with useful RGB color for each 3D point detected on the scene.

In this work, we have used a PR2 robot from Willow Garage, working under ROS environment. The system is completed calibrated, so position and orientation of the Kinect sensor placed on the head of the robot (Fig. 1) can be recovered in the robot reference frame at any time. Using 3D space transformations and the TF ROS package, it is possible to erase 3D points corresponding to the floor, reducing points in the cloud, but moreover, thanks to TF transformation it is possible to have the height (in the world reference frame) of all 3D points in the cloud. If exists a map to locate the robot, the walls could be matched and deleted from the cloud (i.e. as with an occupancy grid). So despite we consider that the robot can be autonomous localized, we do not use this information for locate furnitures, or erase 3D points corresponding to the walls.

In order to have a robust algorithm, it is very important to select the main characteristics to detect of each piece of furniture, and avoiding conflicts. As for example, table detection can be made using detection of horizontal planes between a height range, as in [12], but as seen on images of Fig. 2, the chest of drawers has also an horizontal plane more or less at the same height. But, together with other features it is possible to detect both kind of furnitures without problems.

We have selected at this stage of the work, only three pieces of furniture, because according to their main characteristics the problem of detection is not reduced only to planes detection. The pieces of furniture selected are: couches, chest of drawers and beds (Fig. 3).

The learning of the furniture 3D models has been achieve

by an implementation based on Iterative Closest Point algorithm (ICP). We take several point cloud images of a given piece of furniture from different points of view. Using this algorithm, a set of point clouds can be matched and the transformation between them can be found and the furniture models can be recovered. However, a detailed implementation is out of the scope of this paper. 3D Models learned of the proposed pieces of furniture are showed in Fig. 4.

The proposed methodology does not search to compare directly 3D models learned with raw data provided by Kinect sensor. It seeks to find evidence of a given piece of furniture is present in the environment, by assigning the most probably class to a random set of 3D points.

In order to find the class of individual 3D points in the cloud, we have notice the following considerations and characteristics as belonging to each piece of furniture:

- *Couch*: Couches are furniture that rarely change in color or texture. While it is possible to fit some predominant planes, the precision of them are not enough to classify individual points that could belong to it. As we can see on Fig. 4a 3D point normals can be practically in any direction, unless applying a smoothing parameter.
- *Chest of drawers*: Color or texture can be also considered as in the case of the couch almost invariant. Unlike the couch, in this case planes are very well defined, particularly the upper plane, as it has been said, could be found at given height range (Fig. 4b). Vertical planes, while can be very well defined are difficult to segment from walls by considering only normals.
- *Bed*: In the case of the bed, the color and/or texture can not be considered as regular, people usually change the bedspread. As in the case of the couch if exist a predominant plane the scale to fit this plane could vary from observations in time, because depends on the wrinkles of the quilt. A multi-scale process can be applied to the 3D point cloud, but it could take more time, than desirable.

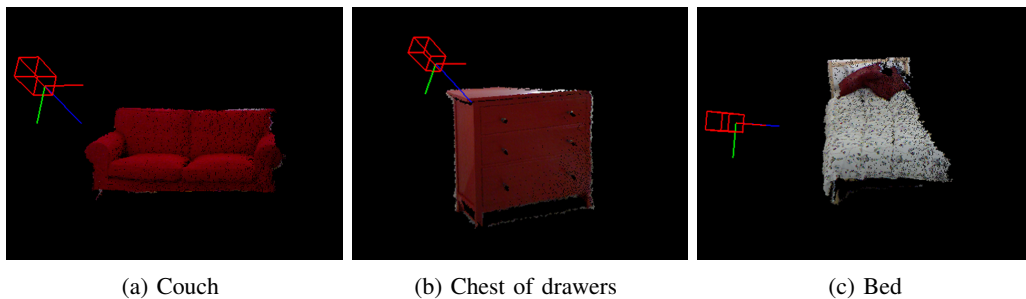


Fig. 4: Models learned for the furniture

IV. FURNITURE DETECTION

To combine the different characteristics and to provide more flexibility for detecting the diverse pieces of furniture, we have assign a feature vector to each point we are going to process. This vector has been defined as follow:

$$C = \{H, S, V, x, y, z, nx, ny, nz\}$$

where:

H, S, V correspond to the color components,

x, y, z correspond to the 3D point position

nx, ny, nz correspond to the components of the normal vector at the given point.

However, normals are not computed for all points in the cloud, allowing to save processing time. RGB color and 3D position are provided directly by the sensor and PCL library, so only color conversion space has to be done.

The process of detection begins with the selection of a random set of points in the cloud. This set is generated, such that all points are below a given height threshold, since we can assume that all the furniture of our interest is below the 1.5m of high. For each point in the set, the characteristics vector C is obtained, then all the points are classified.

The classification is done by a set of artificial neural networks. Each furniture class has its own neural network for deciding if a point could belongs or not to its class. A feed-forward backpropagation network was used for all the classes. Each point is evaluated in the neural network of each of the classes, allowing, at this point, the possibility of belonging to more than one class. This ambiguity help us to solve the fact that if we were using only one neural network, some points could be classified incorrectly because its characteristics are very similar to more than one furniture. For example, the cushions on the bed, are easily taken as a part of the couch because of their color, or some points in the couch could be detected as part of the bed because of their normals.

The training process of the neural networks was done offline, for each network it was used one point cloud (the ones shown in figure 3) as the training data. For each point on the point cloud it was created the input vector, based on the characteristics' vector, but in this case, the three components of the 3D point position were replaced for the height value, so the input vector for the neural networks has

seven variables. All the networks were constructed with one hidden layer with five neurons, and the hyperbolic tangent sigmoid as transfer function. The class value provided for the training was assigned by hand based on our observation.

After the random generated set of points is evaluated in the neural networks, it is verified how many points are for each furniture class and the standard deviation of each class. Based on these information, the class with the most points and a low standard deviation is selected as the most promising.

Now, for each point of the selected class, its defined a new search window in which new random points are generated. The size of the search window would depend of the characteristics of the furniture class selected previously as the most promising. All the points which belong to the other classes are eliminated, and those who were labeled with two different classes only keep the selected as the most promising.

These new "child" points are evaluated only in the neural network of the selected class and only those which have the same class of its "father" will be preserved. Finally a small area around each "child" point (5 points of radius) is fully analyzed, all the points within the area are evaluated, except those who were already analyzed in a previous process.

The algorithm 1 shows a pseudocode of this approach.

Algorithm 1 Pseudocode of the proposed algorithm

```

load_point_cloud();
r_points = select_random_points();
classify_points(r_points);
select_most_promissing_class();
for each point in selected class do
    define_new_window_search();
    while there are points to analyze do
        new_points = select_random_points();
        classify_points(new_points);
        for each new_points do
            if new_point.class == point.selected then
                out_points = +new_point;
            end if
        end for
    end while
end for

```

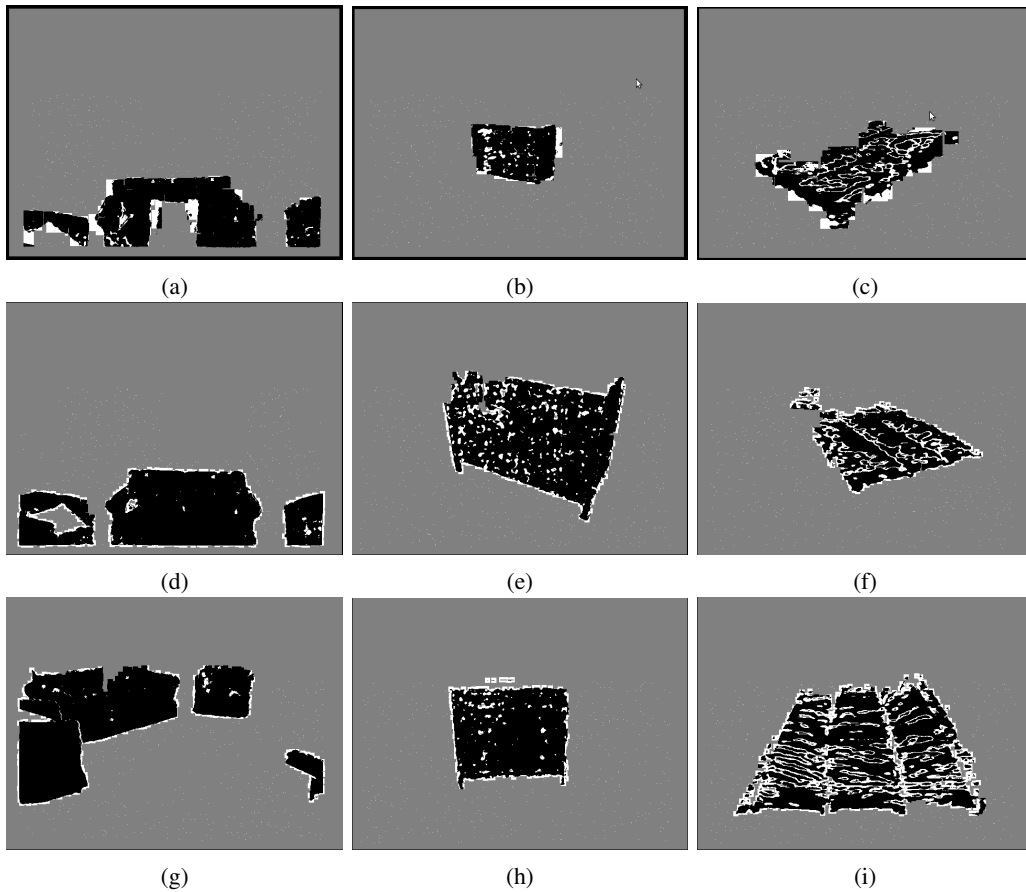


Fig. 5: Result masks for different images of the couch, the chest drawer and the bed.

V. RESULTS

The figures 5 and 6 show the results obtained for nine different point clouds, three for each class. The Fig. 5 shows the masks obtained for the detection. The black and white points were analyzed, the blacks are those classified as part of the furniture, and the whites are those who were not. All the gray area was not analyzed. The figure 6 shows the sections of the cloud points identified positively as the furniture.

The results shown in figures 5 and 6 with the indexes a , b and c correspond to the images shown in figure 3 which are the training images.

The images with indexes a , d and g in figures 5 and 6 shown the results for the detection of the couch. It can be seen that this approach can detect the furniture even in the presence of other objects, for example, the chair in front of the large couch in the images with index a . In the figure 6g it can be seen a false positive, which correspond to a box placed on the floor, these points could be later eliminated by comparing the cluster with the models of the furnitures.

The detection of the chest of drawers (shown in the images with indexes b , e and h of the figures 5 and 6) did not present many problems but the horizontal surface, in this case, the reflected light prevented the 3D information could be obtained.

For the detection of the bed (shown in the images with

indexes c , f and i of the figures 5 and 6) it can be seen how the most of the horizontal surface of the bed was detected even when it is an irregular surface and a plane detector algorithm would not be able to detect all the surface as one plane.

VI. CONCLUSION

In this paper it was proposed an approach to detect furniture in a home like and changing environment. This approach proved to be successful in detecting different types of furniture with different types of characteristics, such as a couch, a bed and a chest of drawers. The fusion of the color and the 3D information proved to be an efficient way to achieve the detection and since it is a stochastic method, it has been proved that it is faster than other methods, for example based on plane detection. Images were analyzed at a rate higher than 10 hz. The proposed methodology can deal with occlusions and partially detected objects. As future works, it is expected to expand the classification to other types of furniture (e.g. tables and chairs), and to incorporate new models suitable for different furniture of the same type, so that the algorithm could adapt to a more generic furniture model.

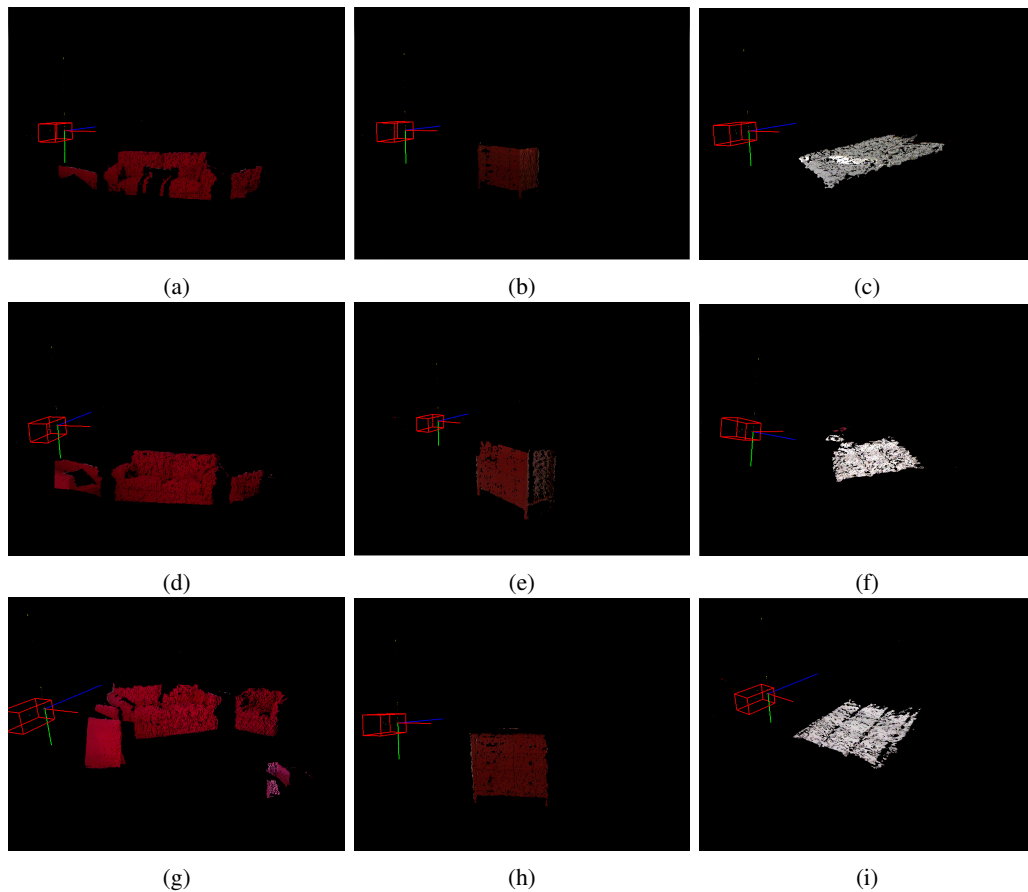


Fig. 6: Point cloud results for different images of the couch, the chest drawer and the bed.

REFERENCES

- [1] R. B. Rusu and S. Cousins, "3d is here: Point cloud library (pcl)," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1–4.
- [2] J. Biswas and M. Veloso, "Planar polygon extraction and merging from depth images," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 3859–3864.
- [3] S.-Y. An, L.-K. Lee, and S.-Y. Oh, "Fast incremental 3d plane extraction from a collection of 2d line segments for 3d mapping," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 4530–4537.
- [4] G. Arbeiter, S. Fuchs, R. Bormann, J. Fischer, and A. Verl, "Evaluation of 3d feature descriptors for classification of surface geometries in point clouds," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 1644–1650.
- [5] C. Choi and H. I. Christensen, "3d pose estimation of daily objects using an rgb-d camera," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 3342–3349.
- [6] C. Choi and H. Christensen, "3d textureless object detection and tracking: An edge-based approach," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, 2012, pp. 3877–3884.
- [7] M. Ruhnke, B. Steder, G. Grisetti, and W. Burgard, "Unsupervised learning of 3d object models from partial views," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE, 2009, pp. 801–806.
- [8] A. J. Trevor, J. Rogers, and H. I. Christensen, "Planar surface slam with 3d and 2d sensors," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 3041–3048.
- [9] G. Hu, S. Huang, L. Zhao, A. Alempijevic, and G. Dissanayake, "A robust rgb-d slam algorithm," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 1714–1719.
- [10] J. Stuckler, N. Biresev, and S. Behnke, "Semantic mapping using object-class segmentation of rgb-d images," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 3005–3010.
- [11] J. Mason and B. Marthi, "An object-based semantic world model for long-term change detection and semantic querying," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 3851–3858.
- [12] M. Wüstel and R. Moratz, "Automatic object recognition within an office environment." in *CRV*, vol. 4. Citeseer, 2004, pp. 104–109.
- [13] J.-J. Hernandez-Lopez, A.-L. Quintanilla-Olvera, J.-L. López-Ramírez, F.-J. Rangel-Butanda, M.-A. Ibarra-Manzano, and D.-L. Almanza-Ojeda, "Detecting objects using color and depth segmentation with kinect sensor," *Procedia Technology*, vol. 3, pp. 196–204, 2012.
- [14] R. B. Rusu, N. Blodow, Z. Marton, A. Soos, and M. Beetz, "Towards 3d object maps for autonomous household robots," in *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*. IEEE, 2007, pp. 3191–3198.
- [15] R. B. Rusu, Z. C. Marton, N. Blodow, A. Holzbach, and M. Beetz, "Model-based and learned semantic object labeling in 3d point cloud maps of kitchen environments," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*. IEEE, 2009, pp. 3601–3608.