Intelligent and Robotic Systems, Special Issue manuscript No.
(will be inserted by the editor)

Part-based Geometric Categorization and Object Reconstruction in Cluttered Table-Top Scenes

Paper type: categories (7) and (5)

Zoltan-Csaba Marton · Ferenc Balint-Benczedi · Oscar Martinez Mozos · Nico Blodow · Asako Kanezaki · Lucian Cosmin Goron · Dejan Pangercic · Michael Beetz

Received: 25.05.2013 / Accepted: 16.12.2013

Abstract This paper presents an approach for 3D geometry-based object categorization in cluttered table-top scenes. In our method, objects are decomposed into different geometric parts whose spatial arrangement is represented by a graph. The matching and searching of graphs representing the objects is sped up by using a hash table which contains possible spatial configurations of the different parts that constitute the objects. Additive feature descriptors are used to label partially or completely visible object parts. In this work we categorize objects into five geometric shapes: sphere, box, flat, cylindrical, and disk/plate, as these shapes represent the majority of objects found on tables in typical households. Moreover, we reconstruct complete 3D models that include the invisible back-sides of objects as well, in order to facilitate manipulation by domestic service robots. Finally, we present an extensive set of experiments on point clouds of objects using an RGBD camera, and our results highlight the improvements over previous methods.

Zoltan-Csaba Marton
Institute of Robotics and Mechatronics, German Aerospace Center (DLR), Germany
E-mail: zoltan.marton@dlr.de

Ferenc Balint-Benczedi and Michael Beetz
Institute of Artificial Intelligence, Universität Bremen, Center for Computing Technologies (TZI), Germany
E-mail: {balintbe,beetz}@tzi.de

Oscar Martinez Mozos
School of Computer Science, University of Lincoln, United Kingdom
E-mail: omozos@lincoln.ac.uk

Nico Blodow and Lucian Cosmin Goron
Intelligent Autonomous Systems, Technische Universität München, Germany
E-mail: {blodow,goron}@cs.tum.edu

Asako Kanezaki
Intelligent Systems & Informatics Lab, Dept. of Mechano-Informatics, Grad. School of Information Science & Technology, The University of Tokyo, Japan
E-mail: kanezaki@isi.imi.i.u-tokyo.ac.jp

Dejan Pangercic
Autonomous Technologies Group Robert Bosch LLC, United States
E-mail: dejan.pangercic@gmail.com

Keywords Object categorization · 3D geometry · Part-graph hashing · Clutter · Domestic robotics

Mathematics Subject Classification (2000) 68 · 60

1 Introduction

Object recognition is a basic capability for service robots working in real domestic environments where cluttered object arrangements often occur, as objects usually touch or occlude each other. In these cases, an accurate segmentations of complete objects is difficult to achieve.

This paper presents an approach for object detection and categorization on such table-top scenes by using 3D point clouds obtained by a RGBD sensor mounted on a service PR2 robot as shown in Fig. 1.

Our method categorizes the different objects according to their general geometric shape. In particular, we distinguish between spheres, boxes, flat objects, cylindrical objects, and disks/plates. These shapes represent the majority of objects found on tables in typical household environments. Since a robotic household assistant could encounter new objects during its operation, no matter how large a training database is, geometric-based categorization and perceptual grouping can be an important step before more particular approaches for instance-level recognition are applied afterwards [9,31].

Inspired by the ideas introduced by Biederman [6], we base our approach on the detection and description of the different parts that may compose an object. We also take into account the different spatial arrangements of the parts by using graph representations. The nodes in the graph indicate the parts of the objects, and the edges represent their connectivity. Matching of graphs is sped up by using additive feature descriptors and hashing look-up tables. The advantage of additive descriptors for our part-grouping method is that we only need to create the descriptor for each part, and all the possible part combinations can be described by the sum of the corresponding additive descriptors. Based on our previous work [29], we over segment the scene into smaller parts and decide based on the arrangement of these parts what kind of object they form. An example of this process is shown in Fig. 1. Finally, complete geometric model reconstruction is performed in order to produce graspable 3D models.

Purely image-based approaches may fail for textureless objects, or under bad lighting conditions for example, as seen in Figures 2 and 3. Our geometric-based part categorization approach lends itself easily for solving such problems. Learning the different parts and their combinations is a scalable way to capture the different object categories a service robot could encounter in a domestic environment. For example, a mug is typically a cylindrical part next to a handle, or a teapot is a combination of different rounded shapes. This idea is supported by research on visual perception [6], and by our previous results on part-based object categorization [33,36,29].

As was shown previously by Lai *et al.* [22], geometric features are more appropriate for categorization, while color features for instance recognition. In this paper we employ geometric, color, and combined features for categorization and find that color features do not generalize as well as geometric ones.

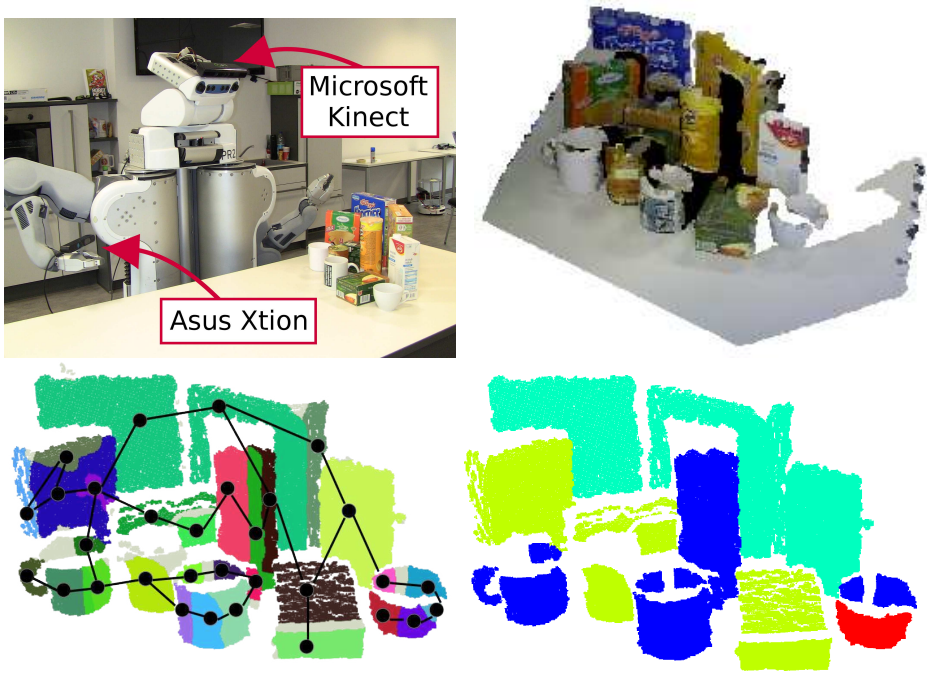


Fig. 1. The top-left image shows our service robot PR2 equipped with two RGBD cameras. The top-right image depicts a 3D scan from a table-top scenario. The images on the bottom show the process of categorization of the different segments in the scene. On the left, the scene is over segmented and a graph is created with neighboring segments. On the right, the final categorization results is shown where cylinders are marked with blue, boxes with yellow, rectangular flat faces with cyan, and (half) spheres with red.

Since multiple objects of the same type could be located in close proximity, assigning the correctly identified parts to two separate objects is difficult using our method and the one of Lai *et al.* [24]. For this, the spatial relations between identified parts and the object center, followed by a geometric validation step would be needed, as we presented in [36]. However, this requires (approximate) CAD models of the objects to be known a priori. Therefore, we incorporate a general geometric model reconstruction method that takes the output of the categorization and produces completed (but simplified) 3D models that are directly usable for grasp planning.

Embodiment is a key aspect of a cognitive system, so that it can gather experiences, as argued by Vernon *et al.* [57]. For a robotic household assistant this implies that it should exploit its exploration capabilities and gather as much information about its environment as possible, in order to be able to successfully recognize the objects in it. Therefore, in addition to our part-based approach, we present how to improve the final categorization results by incorporating multiple views of the same scene and multiple interpretation of the data, and by enriching the training database with objects that are different from those in the testing set. The whole process is designed such that it can handle cases where objects are only partially visible and that it assigns a label to each detected part, in contrast to methods

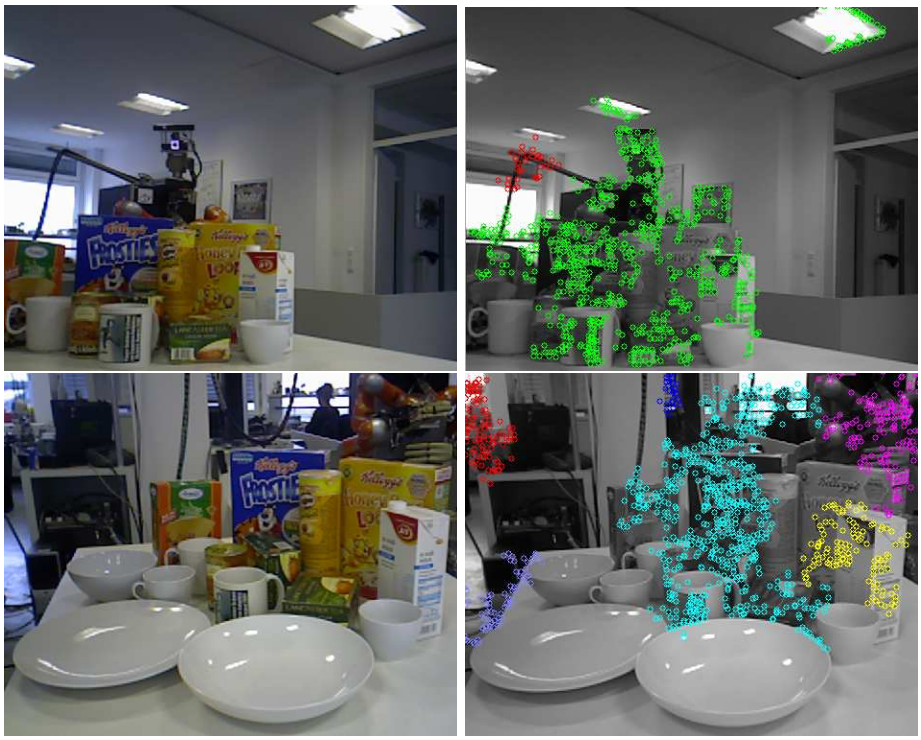


Fig. 2. Clustered keypoints detected using the SIFT-based ODUfinder [38], showing the difficulties encountered by texture-only methods.

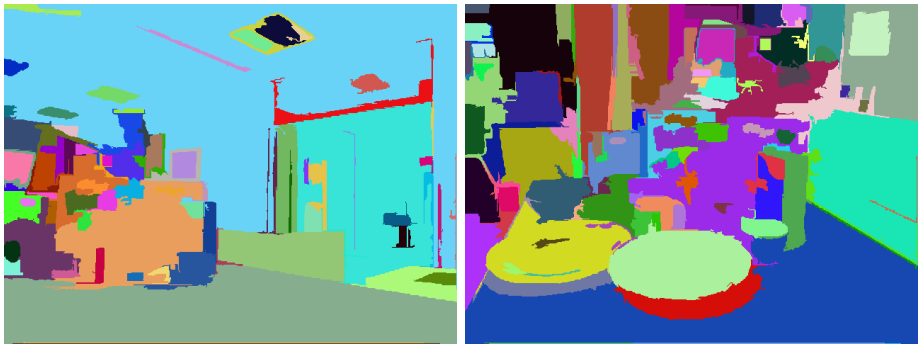


Fig. 3. Image-based segmentation results of cluttered scenes using [11]

like [22, 23]. We evaluated our geometric categorization approach on RGBD scans of cluttered tabletop scenes of previously unknown objects, and experimented with enriching the training set by combining different databases.

As a summary, we propose the use of over-segmentation and accumulation of information to achieve object categorization in clutter. The main contributions of this work are:

– a part-graph hashing method for efficient geometric categorization in cluttered scenes;

– enabling robots to take advantages of multiple views, multiple segment groupings, and multiple data sources;
– an evaluation of geometric, color, and multi-modal features and classification approaches on generated and real world scenes.

After an overview of the related work in the next section, we shortly introduce the additive depth, RGB and combined descriptors in Section 3. The method introduced in [29] is shortly presented in Section 4, followed by a motivation for the selection of the geometric categories in Section 5. The evaluation on real world scenes, and large-scale tests on generated ones, along with comparison of descriptor and methods are presented in Section 6. Sections 7 and 8 present the model reconstruction and the merging of results from different views, respectively. Finally, we summarize the findings in Section 9 and discuss further research paths and extensions.

2 Related Work

Template matching approaches like the ones presented by [27, 55, 54] focus on the detection and pose estimation of known objects in the scenes. On the other hand, our method generalizes well to novel objects, thus it can be used to recognize new instances of given object categories. Additionally, in many color or depth image based methods, detection is typically limited to cases where they are fully visible [44, 23].

We wanted to avoid such limitations, and focus on 3D features of object parts, and their spacial relationships. An overview of global and local 3D features is given in [2]. Unlike many of the global feature based approaches, our method does not require a correct segmentation of the objects. Still, it is distinguished from local feature based approaches as well, since it operates on complete parts of objects, not just keypoints.

As discussed in [18], perceptual organization should be captured using models that take account of the part structure of objects and capture the properties of 3D shapes. As argued for example by Huber [17], part-based detection has the advantage of generalizing to unknown instances of object types. While in [17], [33] and for the part-based VFH (called CVFH) feature [1], objects need to be singulated first, other part-based approaches like [24, 36] can efficiently detect objects in clutter. This typically requires over-segmenting the scene, possibly multiple times, such that object boundaries are respected. Because an object can be split into multiple parts, a correct and fully reproducible segmentation is not needed, thus simpler segmentation methods can be employed, usually based on detecting properties like concavity, that are known to delimit objects [6, 49].

While there are many promising advances in correctly segmenting objects, with and without some human aid, like [4, 35, 13, 8], as pointed out by Malisiewicz *et al.* [28], considering multiple over-segmented views is preferable to relying on a single, possibly erroneous, segmentation. There are several other approaches to clutter segmentation, but with limited exploration of the object recognition problem, like [51, 46].

Recent approaches to image segmentation and annotation are presented by Socher in [50] and shown to be outperformed by the part-based approach proposed by the authors. After over-segmenting the image they create a binary tree

representation by recursive merging nodes and assigning semantic features to them. The authors apply the same method to parsing natural language as well. While the approach looks extremely promising, the merges are not guaranteed to be performed first inside a single object in the case of multiple objects of the same type being next to each other. Instead of a fixed grouping, we enumerate all the possible subgraphs a node can be part of, and classify each of them separately. This is followed then by geometric model reconstruction to create graspable object models.

Fergus *et al.* [12] model objects as flexible constellations of parts in an unsupervised scale-invariant learning approach for detecting objects in images. Our method can be viewed as the 3D application of the presented principles, with the difference that we use additive features and consider part combinations, aided by hashing of graph features that capture spacial topology. Our approach is similar to the one presented by Felzenszwalb in [10], but which uses only RGB data. However, the core idea that objects are represented by mixtures of deformable part models was used in this work, by capturing relations between parts identified in an unsupervised manner by a classifier. Shotton *et al.* [48] incorporate poses and viewpoints, texture, layout, and context information for image segmentation based object recognition. In a complementing publication [47] they address the problem of categorical objects recognition and localization in space and scale using a sliding window classifier. Although the method is image based, in its formulation and its use of geometry related image features it is similar to 3D approaches, that become more and more popular.

In the 3D domain, a related approach by [40] uses abstract shape class representations that capture the spatial relationships between the object parts. Instead of point signatures, we compute descriptors for complete parts, thus can take advantage of more descriptive features.

As in [36], we use a probabilistic voting approach for assigning class probabilities, a method that is also used in the computer vision community, for example in [26]. In another example, Sun, Min and Bradski [52] use voting for object detection, through relating image patches to depth measurements.

In [24] the authors also propose a similar system for understanding cluttered scenes. Our approach combines the over-segmentation from [36] with an extended version of creating multiple groupings of these "parts" [24], and was designed to handle multiple instances of objects from several categories, that were labeled according to their general 3D shape. While in [36] information coming from the different parts of the object was combined by a Hough voting scheme for identifying the object's 2D centroid, the approach presented here is more close to [33]. Identifying to what object does each part belong consists of considering its descriptor and that of neighboring parts, together with the local topology of the scene. In this sense, it is an improvement over the vocabulary of parts and simple vote accumulating approach from [36]. Furthermore, this work focuses on objects relevant to pick and place scenarios, which can have various 6 degrees of freedom poses instead of 3 for furniture pieces.

Since we do not assume having CAD models of the test objects, fitting of simple geometric shapes is required for creating graspable models for robotic interaction. In [53], an exhaustive nonlinear optimization is employed for reconstructing boxes, cylindrical and spherical items. This approach is relatively slow, and therefore often RANSAC is preferred [45,41]. Recently, Richtsfeld *et al.* [39] presented a multi-

level approach to fit planar or curved surfaces to over-segment parts, and then define inter-segment relations to decide if they should be merged or not. Unlike our approach, they consider relations between non-touching parts as well, but the method performs best for merging touching segments and for convex shapes. This work employs the simpler reconstruction from [15], but the categorization method provides geometric labels, and can easily deal with concave objects. Therefore combining the two approaches would be interesting.

Detection of small objects in clutter using a sliding window was explored by Kanezaki *et al.* [20] using an additive feature. If a feature is additive, the descriptor that would be computed for the object is (approximately) the same as the sum of the features of its parts. Thus it is especially useful for detecting objects based on features computed only for parts of it, for example by using the Linear Subspace Method (LSM) on the feature space, as presented by Watanabe *et al.* [58]. However, the Linear Subspace Method does not exploit the relations between the different parts of the objects. We used the additive property of 3 features (GRSD- [29], C^3-HLAC [21] and VOSCH [19]) to compute the descriptor of grouped parts by summing up the parts' descriptors, and here we compare our method to those presented in [20] and [36].

An independently developed work by Mueller *et al.* [37], based on similar ideas than the ones in [56,36,29,39], appeared after the acceptance of this article. A comparison would have been interesting but could not be included here, and the strength of their method is difficult to asses without it. In their evaluations they show promising results for excluding irrelevant parts of the environment (so they don't rely on a tabletop-type scene), but they focus on three distinctive object types. Unlike the method presented here, the one in [37] is explicitly limited to convex objects, and only one part combination is used for the categorization task.

3 Additive Descriptors for Object Categorization

One key component of our approach is the use of additive feature descriptors. As we indicated in the introduction, the advantage of additive features for our part-grouping method is that we only need to create the descriptor for each part, and all the possible part combinations can be described by the sum of the features of the constituent parts[1], as shown in the last column of Figure 4. In this paper we focus on different types of additive feature descriptors: the 3D-only GRSD- and its color-based extension, VOSCH.

The GRSD- [19] descriptor is the additive version of the Global Radius-based Surface Descriptor (GRSD). The GRSD descriptor is represented by a histogram that counts the number of transitions between different types of surface voxels: free space, plane, cylinder, sphere, rim, and noise [32]. To compute the GRSD- feature vector, a simple local surface classification is performed to obtain the voxel types, and the numbers of times different surface types (and empty space) are in neighboring voxels is recorded (instead of the ray tracing for GRSD). Thus, if this descriptor is computed for different parts of an object separately, the sum of these descriptors is approximately the same as the descriptor of the object. As evaluated

[1] Another important property of additive descriptors is that partial segments of objects can be matched against complete objects in a database by looking for the corresponding part in the original object. This is not explored in this paper, however.

in [29], there was no loss information for the categorization task with respect to GRSD (but a minimal increase of under 0.1%), and both features performed close to the high-dimensional VFH feature.

The Voxelized Shape and Color Histogram (VOSCH) [19] is composed of a concatenation of the Circular Color Cubic Higher-order Local Auto-Correlation (C^3-HLAC) descriptor [21] and the GRSD- descriptor. The C^3-HLAC is an additive descriptor based on the color information contained in a voxelized representation of the point cloud representing the image. The C^3-HLAC descriptor is variant to rotations, therefore artificial rotations need to be included during training. Both GRSD- and C^3-HLAC create histograms of inter-voxel relationships, therefore their combination is a straightforward way of creating an RGBD feature. (First, C^3-HLAC has to be adjusted to be rotation invariant, by not considering the relative orientation of neighboring cells when building the histogram.)

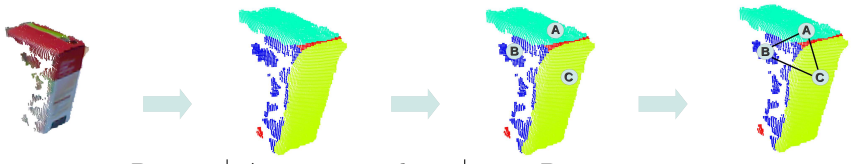
VOSCH and GRSD- are rotation invariant, unlike C^3-HLAC, and all three descriptors can be applied to general 3D data, not only single depth images. Therefore, the presented method can be applied to other types of data as well, like 3D registered laser scans for example.

4 Part-graph Hashing Based Categorization

In our previous work [33,36,29] we found that a part-based approach lends itself easily for solving object detection when segmentation is problematic. Our geometric categorizations' basic idea (detailed in [29]) is that segmenting objects accurately does not always work robustly and will result in labeling mistakes, but over-segmentation is easily realizable [28,24]. Learning the different parts/segments and their combinations that form objects is a scalable way to capture the different object categories a robot would encounter. For example, a mug is typically a cylindrical part, next to a handle, or a teapot is a combination of different rounded shapes with a top and a large handle. The obtained segments represent only a sub-part of objects but can be used to compute features, and combined to build up object candidates, as shown in Figure 4.

As reviewed in [49], there are certain principles that should guide the search for perceptually salient parts, of which we rely in this work on the "hypothesis of normalized curvature" and the "hypothesis of turning angle". We use a segmentation by region-growing to over-segment the scans based on estimated surface normals (using the criteria presented in [36]), such that patches with a relatively small curvature are considered. In a typical scene consisting of around 10^5 points, this method created around 50 parts, and over 100 groupings (see Figure 1).

There are of course several ways of combining parts, not all of them creating a valid object. However, testing the validity of a combination is possible by checking if the combined feature vector is known. We also exploit the fact that parts and their connections (neighborhood relations) can be treated as a graph, and only certain types of sub-graphs are present in the graph representing an object (formed by its parts). Checking for subgraph isomorphism is not practical, but there are several descriptors one can employ to rule out isomorphism. Thus, during training we decompose our objects into parts, compute the features for each part, build the part-graph, and generate all sub-graphs along with their combined additive descriptors. Each sub-graph has an "arrangement key", which in our case



Part groupings	Arrangement keys (for hash table)	Descriptors (and their sums)
A	1	$d_A = (f_A^1, f_A^2, f_A^3 ... f_A^n)$
B	1	$d_B = (f_B^1, f_B^2, f_B^3 ... f_B^n)$
C	1	$d_C = (f_C^1, f_C^2, f_C^3 ... f_C^n)$
	11	$d_{AB} = d_A + d_B$
	11	$d_{AC} = d_A + d_C$
	11	$d_{BC} = d_B + d_C$
	222	$d_{ABC} = d_{AB} + d_C$

Fig. 4. Overview of part-graph hashing (using a single object, as during training)

is formed of the connectivity degrees of its nodes (see Figure 4), and this can be used to build a hash table that partitions the training data. Therefore we can avoid confusions between subgraphs that do not have the same number of nodes, and shorten training and prediction time.

The whole process is shown in Figure 5. The input to the hash table is a set of subgraphs with different connectivities, and thus different arrangement keys. Each arrangement key points to a list of training objects containing subgraphs with the same connectivity. The set of subgraphs for each object is represented by a list using its corresponding additive descriptor, which was obtained by summing up the descriptors for each node in the graph as shown in Figure 4.

When testing, the procedure of segmentation (decomposition into parts) and part-graph building is repeated, and starting at each part, all the subgraphs are grown that are not larger than anything seen during training. The part groupings' descriptors are then classified for which objects can they be parts of. Therefore a nearest neighbors classifier is built for each entry in the hash table, and the obtained probability distributions (representing the similarity to each geometric category) are accumulated in the constituent parts for final classification, as detailed in [29]. Lai *et al.* [24] used the product of the class probabilities for each grouping, but we found that the confidence weighted voting approach performs better, as supported by the experiments in [34] as well.

5 Selection of Geometry-Based Categories

Our categorization method labels the parts as forming an object of the following general geometric categories: *sphere*, *box*, *flat rectangle*, *cylindrical*, *disk/plate*, or *other*. These intuitive categories match most of the objects for which we had appropriate training data (and the remaining ones were assigned to the *other* cat-

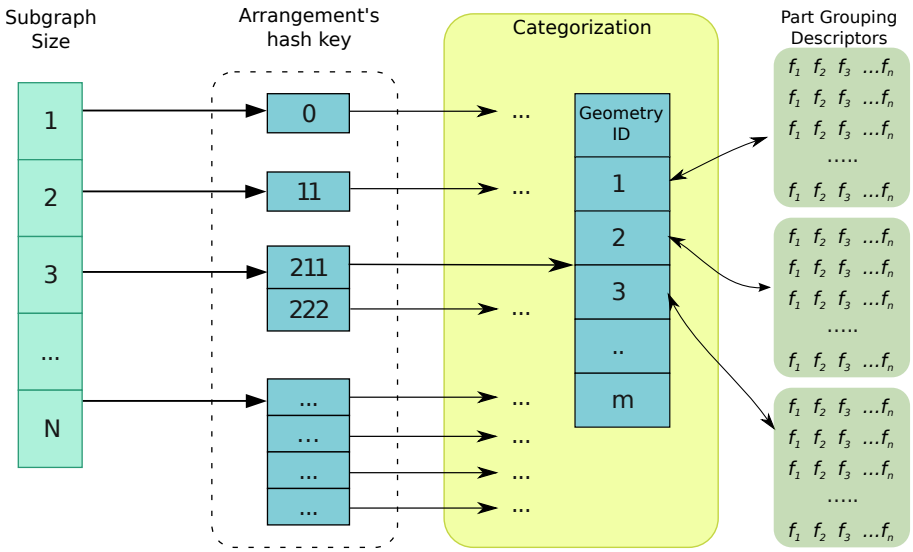


Fig. 5. Structure of the hash table holding the classifiers build based on the training examples.

Fig. 6. Unsupervised RIM clustering compared to the manually defined geometric categories. Each cell represents the amount of overlap between the manually defined clusters (rows) and the result of the unsupervised clustering (columns), with blue meaning no overlap and red meaning large overlap. Clusters overlap well with the used categories, with two geometrically similar pairs merged.

egory), and also the categories we found in public household objects databases, as detailed in [31]. As in our previous works, the categories are given by human intuition, but results using unsupervised clustering of geometric features show that they make sense also based on the data, as detailed below.

We used the Regularized Information Maximization (RIM) unsupervised clustering and classification technique [14] to find meaningful clusters of GRSD- descriptors obtained from objects in our training data and assign testing instances to these clusters. (For this test, a descriptor was computed for the complete object, which corresponds to the summed up descriptor of their maximal part grouping.) We measured how well do the clusters overlap with the given categories by computing the Adjusted Rand Index (ARI), which is the analogue of the true positive rate for (unsupervised) data partitioning methods. Its values range from 0 to 1, with 1 meaning that the two labelings are identical.

We ran RIM using different values for its λ parameter, and for different number of seed clusters. The best ARI (0.36) is obtained using $\lambda = 90$ and 6 clusters, with stable results around these values. The matrix in Figure 6 shows how well the six unsupervised obtained clusters (columns) match the clusters created by hand using

our 6 predefined geometric classes (rows). Entries in this matrix with colors close to red indicate a good match, i.e. same objects in both clusters, while colors close to blue indicate that the clusters contain different objects. As shown in Figure 6, the clusters are quite clean, and also the categories are grouped nicely with clusters, but cylindrical objects were merged with boxes (into the unsupervised cluster 1) and flat ones with plates (into cluster 3). This makes sense given that GRSD-encodes only relations between neighboring voxels, thus features like the contour are not captured. Additionally, small boxes and cylinders can look quite similarly in Kinect scans, especially after smoothing. However, we chose to keep these two pairs as separate categories as they are semantically different (using a different feature they still could be separated) and could provide relevant information for model fitting and grasping applications.

As seen in Figure 6, smaller unsupervised clusters are created as well by RIM, into which parts of the object categories are separated, suggesting that more views of object instances from a category could be grouped together (e.g. side and front views of flat rectangular objects like cereal boxes). Such a strategy was used in [31] to increase the geometric categorization accuracy.

6 Evaluation and Discussion

For acquiring and preprocessing our scans we rely on the Point Cloud Library (PCL) [42], and the Robot Operating System[2] (ROS). Noise removal, smoothing, normal estimation and object cluster detection were addressed already exhaustively in previous work [43,31], and here they are applied as described, apart from the real-time implementation detailed in subsection 8.2. Before we detail the experiments and discuss their results, we will give a short overview of the used datasets. Part of the experiments described in this work were reported in a shortened form at a workshop [30].

6.1 Datasets of Individual Objects

As detailed in [29], for our experiments we used a part of the large RGBD dataset from [22], which contains over 200,000 scans of 300 objects from 51 object categories. Additionally, we used the dataset from [19] to add knowledge about the objects in our environment. These datasets already contain segmented models of individual objects. We use these individual models to classify objects in cluttered scenes without having to model multiple touching, and/or occluding scenarios.

6.1.1 RGBD-Large

As in [22], we use every fifth point cloud from the dataset in our experiments, because the similarity between consecutive point clouds is extremely high. Since in this work we focus on categorization into general geometric shapes, we selected those object categories that have good 3D data (and excluded very small, shiny or transparent objects) and grouped them into geometric categories. The categories

[2] `http://www.ros.org/wiki/`

and the respective objects are as follows, with the number of objects shown in parentheses:

- **Sphere**: bowl (6), ball(6)
- **Box**: food box (4), sponge (8)
- **Flat (rectangular)**:notebook(5),cereal box(5), food box(8)
- **Cylindrical**: coffee mug(8), food cup(3), soda can(4), food can(14), food jar(6), water bottle(6)
- **Plate (disk)**: plate(7)
- **Other**: cap (4), Kleenex (5), pitcher(3)

6.1.2 RGBD-Small

In order to be able to test and compare our method and features, for some of the more time-intensive tests we reduced the dataset to roughly 7000 scans of 57 objects from 9 object categories. The objects in this reduced version are:

- **Sphere**: ball(6)
- **Box**: food box (4), sponge (8)
- **Flat (rectangular)**: food box(8)
- **Cylindrical**: coffee mug(8), food cup(3), soda can(4)
- **Plate (disk)**: plate(7)
- **Other**: cap (4), pitcher(3)

6.1.3 VOSCH Dataset (VDB)

Because some of the objects used in our scenarios differ a lot from the ones in the RGBD datasets, in order to diversify our training data we combined the RGBD datasets with the "VOSCH" Kinect scan dataset (VDB) used in [19]. The VDB consists of 34 similar objects to the ones in our scenes, captured from different viewpoints with an angular step of 15 degrees.

- **Sphere**: bowl(3)
- **Box**: milk (5), tea box (2)
- **Flat (rectangular)**: coffee filter(2), cereals(2)
- **Cylindrical**: mug(4), food cup(2), soda can(2),Pringles chips(2)
- **Plate (disk)**: plate(5)
- **Other**: shampoo (1), tea pot(1), pancake mix(1), yogurt(1), juice(1)

Fig. 7 shows object from the datasets described for each of the categories.

6.2 Complete Cluttered Scenes

In this section we present categorization results of several cluttered table-top scenes containing different objects.

Since manually labeling these scenes with ground truth categorization is a time-consuming process, we could evaluate only a couple of them. We present results on 3 frames in this subsection, and a sequence of 6 scans of a fixed scene will be used in the next section. Figure 8 show three tabletop scenes on which we tested our approach.

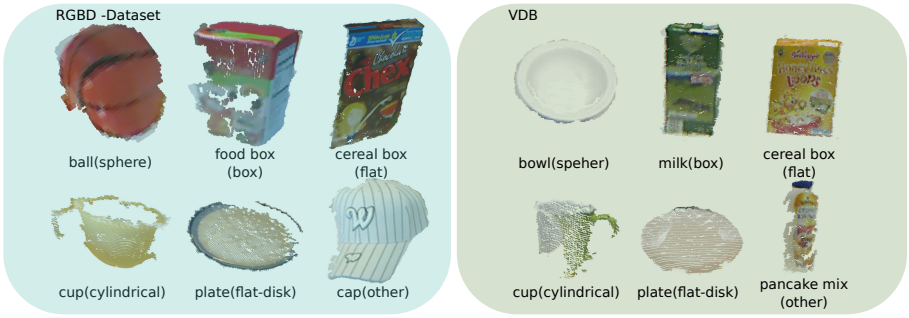


Fig. 7. Examples of standalone objects in the RGBD and VDB datasets

Testing on the cluttered scenes was run using different datasets (or combinations) as training data, as shown in the columns of Tables 1 and 2. For each of the three scenes, we report the true positive rate obtained by categorizing parts. Additionally, the same result is used to compute a per point true positive rate as well, in order to compensate for the differences in object part sizes. As it is expected, the results vary depending on the type of feature descriptor and on the training dataset (best results are marked in bold).

Fig. 8. Segmentation (middle row) and geometric categorization (bottom row) on three cluttered scenes (top row). In the last row, the color red represents the *sphere* category, blue *cylinder*, yellow *box*, and cyan the *flat* category.

In order to diversify our training data we combined the RGBD datasets with the "VOSCH" Kinect scan dataset (VDB) used in [19], consisting of 63 similar objects than in our scenes, captured from different viewpoints with an angular step of 15 degrees. Similarly to [24], we found that this "domain adaptation" improves results, as seen in Table 1. However, as the results on the larger RGBD dataset suggest, identifying the correct weighting of the two data sources is necessary,

possibly based on an evaluation set. Apparently, as the number of objects increases, confusions get more frequent, therefore the weight of the domain specific objects need to be increased.

In the case of the smaller dataset, the combination with the scans from VDB improved over the results on both separate training sets, highlighting the importance of mixing various sources of information while keeping specific specialties. Thanks to the hashing approach, handling large databases and dynamically adding new objects is alleviated, as only affected groups have to be re-trained.

Datasets:	RGBD-Small	RGBD-Large	VDB	Small+VDB	Large+VDB
Scene 1					
per point	73%	47%	76%	**84%**	54%
per part	**83%**	49%	67%	**83%**	54%
Scene 2					
per point	76%	54%	70%	**80%**	58%
per part	**76%**	41%	72%	74%	47%
Scene 3					
per point	70%	42%	78%	**88%**	70%
per part	76%	45%	**84%**	80%	76%
AVERAGE					
per point	73%	48%	75%	**84%**	61%
per part	78%	45%	74%	**79%**	59%

Table 1. Results in clutter using different training datasets with GRSD-

Datasets:	RGBD-Small	RGBD-Large	VDB	Small+VDB	Large+VDB
Scene 1					
per point	29%	39%	**77%**	60%	55%
per part	47%	43%	**75%**	64%	50%
Scene 2					
per point	26%	25%	**79%**	52%	40%
per part	36%	27%	**72%**	50%	32%
Scene 3					
per point	73%	**81%**	45%	73%	**81%**
per part	56%	**68%**	60%	56%	**68%**
AVERAGE					
per point	43%	48%	**67%**	62%	59%
per part	46%	46%	**69%**	57%	50%

Table 2. Results in clutter using different training datasets with VOSCH

Lai *et al.* reported results on the comparison of visual and geometric features using the database presented in [22]. Their tests highlight the fact that geometric features are more suitable for categorization and visual ones for instance recognition, but they found that visual features outperformed geometric ones both at instance and category recognition, while a combination of both works best. Using our experiments this was not the case, suggesting that their conclusion does not hold in every case. When using the color-dependent VOSCH feature, the fact that many of the test objects are from VDB becomes reflected in higher success

rates, as shown in Table 2. However, these results are worse than the corresponding results using GRSD- and much worse than the best results obtained with the purely geometric feature (despite the large difference in dimensionality). This was also confirmed using the large-scale evaluation in Table 5, using the methodology of [22]. We believe that the contradicting results are due to the fact that in [22] some categories show little variation among the instances (at least with the employed features).

Run-times vary depending on the dimensionality of the extracted feature and the scale of the used dataset, with classification on the small VDB dataset using the only 20 dimensional GRSD- feature yielding the fastest results, due to the fact that the VDB contains only around 900 individual scans of objects. The average classification times for the three scenes, shown in Table 3, were obtained on a single core of a 2.4 GHz CPU. The overall time to generate all part groupings, classify them and obtain the final categorization results was in the range of 5 seconds.

Runtimes using diff. datasets	RGBD-Small	RGBD-Large	VDB	Small+ VDB	Large+ VDB
GRSD- [20d]					
per point	2.4E-05	4.4E-0.5	0.41E-05	2.88E-05	4.76E-05
per part	0.043	0.083	0.007	0.053	0.089
VOSCH [137d]					
per point	1.4E-04	2.3E-04	0.19E-04	1.6E-04	2.5E-04
per part	0.27	0.43	0.03	0.30	0.47

Table 3. Average classification times in seconds for the scenes from Figure 8

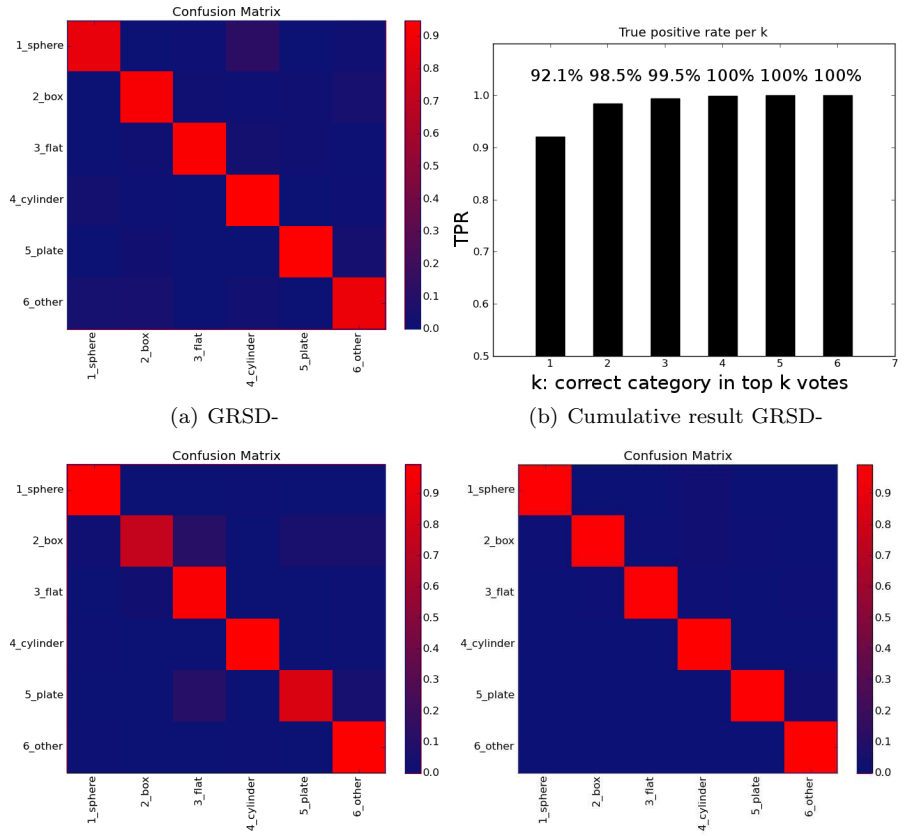
For a more detailed evaluation, the next subsections will focus on large scale tests using the RGBD dataset, using separated objects as queries. The RGBD-Small set was split 2:1 into a training and testing scans [29], except for the cross-validation test that was performed using the methodology from [22]. Given that the objects are already segmented, our approach can take advantage of the fact that only a single object needs to be categorized, and merge the results obtained for the different parts by weighting the label probabilities by the number of points in the part.

6.3 Evaluation of Features

In our earlier work we tested different distance metrics for nearest neighbors classification and found that the Jeffries-Matsuhita distance performs best [29]. Due to the hashing procedure, the separate classifiers for each hash key combination have an easier job in distinguishing parts coming from different categories. Thus results are on par with that obtained with Support Vector Machines: our method was 0.1% better in the test from [29]. However, the hashed nearest neighbors approach has considerably shorter training time.

	GRSD- [%]	C^3-HLAC [%]	VOSCH [%]
RGBD-Large	92.1	98.48	94.59

Table 4. Categorization results on the 2:1 split of the RGDB-Large set



(a) GRSD- (b) Cumulative result GRSD-

(c) VOSCH (d) C^3-HLAC

Fig. 9. Confusion matrices and cumulative score on the 2:1 split of the RGBD-Large set. Please note the different scale of the confusion matrices used for better visualization. The success rates are shown in 4.

Here we present again the results obtained by our method on the RGBD-Large dataset, but extend it with a comparison to the C^3-HLAC and VOSCH additive features. For this test, we left out every third scan of an object from the training set, and used it for testing. After checking how often the correct category obtained the highest score for each object, we also checked how often it was amongst the first k probabilities ("cumulative" results). Results are shown in Figure 9 and Table 4, with an interesting observation relating to the cumulative results in Figure 9 (b): the two most likely categories are 6% more often correct than the ones reported as most likely (i.e. in most of the failure cases, the correct label was ranked second). This suggests that in case we obtain similar top scores, re-segmenting the test scene into parts (with different random seeds) could improve the labeling, by merging the votes from different segmentations. This approach was employed in the next section in the case of different views.

We also compared GRSD- to rotation-invariant C^3-HLAC [21], and their combination VOSCH [19] using the Linear Subspace Method, as that method is also

employed to locate objects in clutter based on additive features [20]. In the training process of LSM, we divided the whole voxel grid of each object segment into cubic subdivisions of a certain size (14 cm × 14 cm × 14 cm with 10 cm × 10 cm × 10 cm overlapping in our case) and then extracted feature vectors from all of the subdivisions to perform Principal Component Analysis. In the testing process, we extracted one feature vector from the whole voxel grid of each object segment. (Note that we do not need to divide the voxel grid in the testing process, because we can calculate the similarity value of the query object to a reference object by simply projecting its feature vector to the subspace, owing to the additive property of the used features.) Results are shown in Figure 10, for different number of principal components selected to define the subspace.

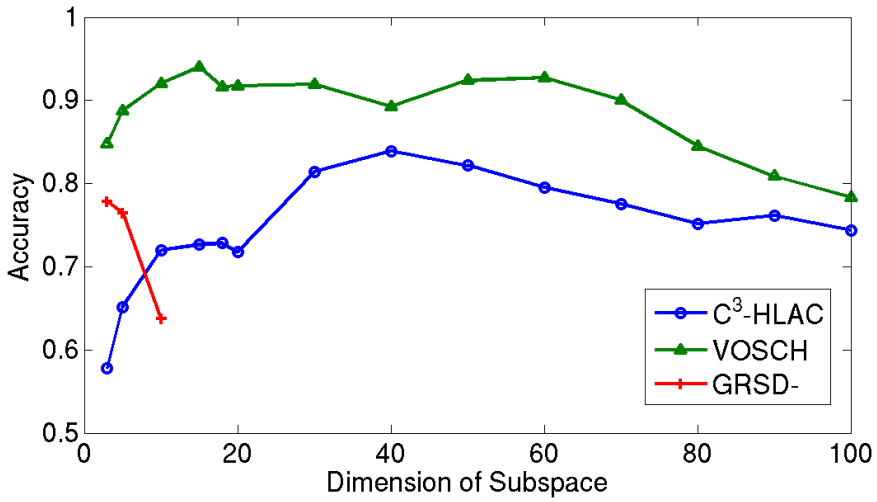


Fig. 10. Accuracy vs. number of subspace dimension. The red cross shows the result with GRSD-, blue circle with C^3-HLAC and green triangle with VOSCH.

The reason why GRSD- performed poorly compared to the features that captured color information, is that in these tests scans in the training and testing set came from different views of the same object. Therefore, we also performed a cross validation experiment (following the methodology in [22]) to test how well these additive features generalize to unknown objects. See Table 5 for results. As it was to be expected, the purely color based C^3-HLAC feature performs the worst (except for the typically white plates), with an average success rate of 59.19%. The VOSCH feature is aided by its geometric part, and achieves 70.88%, while in this experiment GRSD- performed best, with an average of 72.06%. There are large variations in the success rates for certain classes, due to the fact that we used only the RGBD-Small set, and some classes are more varied than others. Still, the smallest variance is produced by the RGB-insensitive GRSD- feature descriptor.

In conclusion, these tests underline the fact discussed earlier in 6.2, that geometry is more stable between instances from the same category as color. This is also supported by findings suggesting that geometric knowledge (shape primitives and

	Sphere [%]	Box [%]	Flat [%]	Cylinder [%]	Plate [%]	Other [%]	**Total**
GRSD-	**67.5±26.2**	52.5±**15.2**	**95.8±2.7**	89.5±**2.6**	47.1±**22.5**	**79.9±14.2**	**72.06**
C^3-HLAC	53.0±28.8	32.1±17.6	80.2±8.6	77.4±10.7	**65.1**±32.2	47.3±22.7	59.19
VOSCH	63.2±27.2	**60.4**±25.2	90.6±7.2	**90.1**±9.5	50.9±27.7	70.1±24.5	70.88

Table 5. Per category leave-one-out cross validation tests on the RGBD-Small set

their spacial relations) have a modality-independent representation in the human brain [59].

6.4 Comparison to Previous Methods

In our previous work [29] we performed a comparison to segmentation-based categorization, by segmenting round and rectangular objects using the method from [15], and found a significant drop in accuracy due to segmentation mistakes. Since we consider multiple segmentation possibilities and the relations between parts, the results were more robust than for a single segmentation and global feature based approaches.

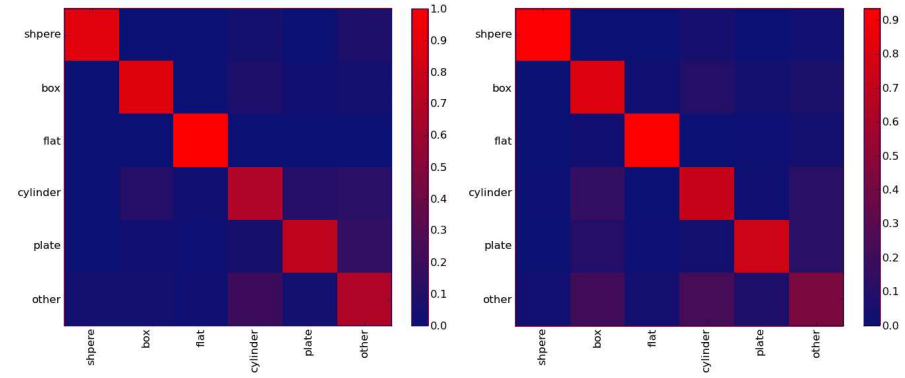
Here we compared our results to those obtained with the statistical features and method described in [36], considering only the part voting step, without the geometric object (pose) identification, as CAD models and ground truth poses are not available for our objects. A vocabulary of size 400 was created out of the descriptors of the parts from the training dataset using K-Means, and used to assign class probabilities to parts in the testing dataset. These votes cast by the different parts are weighted by their similarity to the activated cluster, and the final class is assigned to the highest scoring one.

Both the statistical features used in the original publication and GRSD- were tested using this method, and we obtained a mean success rate of 80.45% for the former and 75.86% for the latter. As seen from the corresponding confusion matrices in Figure 11, the difference is due to the fact that the miscellaneous "other" class is handled considerably better by the statistical features – if this class is ignored, the two features give practically the same result. Since the original features are not additive, using them in the current method would require its repeated re-computation, something we would like to avoid. Moreover, some of the statistical features are orientation dependent, thus would require training objects in multiple poses.

Our method and LSM was also evaluated on the same data, using GRSD- (due to the findings from Table 5). Overall, the results indicate a clear advantage of the part-based categorization process, as shown in Table 6.

	our method	part vocabulary [36]	LSM [20]
Success rate	95.5	75.9	77.8

Table 6. Results using different methods on the RGDB-Small datasets, with RGBD-



(a) Statistical Feature (b) GRSD-

Fig. 11. Confusion matrices of the vocabulary of parts method.

Fig. 12. Synthetic scene automatically created and labeled from individual object scans.

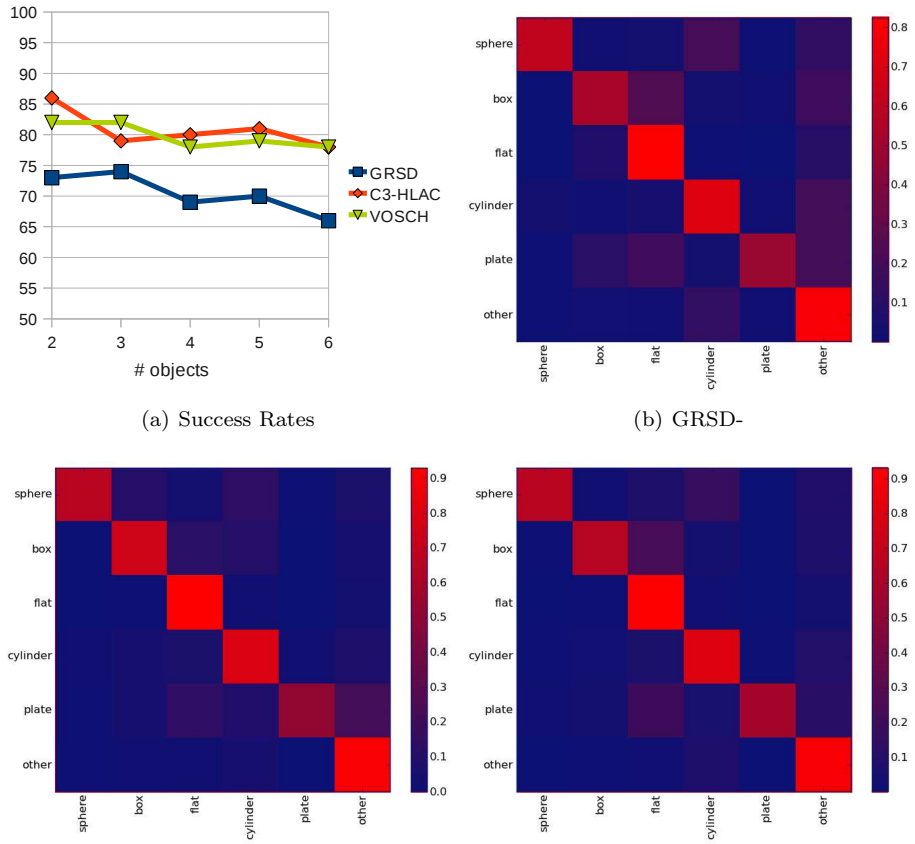
6.5 Generated Scenes Containing Touching Objects

This subsection presents results on a large scale test on scenes containing touching objects (without occlusions). For a realistic simulation of cluttered scenes, CAD models of objects would be required, and an algorithm to generate tabletop scenes with the objects touching and occluding each other, followed by a realistic sensor simulation and automatic point cloud labeling. As all this would be challenging to obtain, we instead generated scenes containing from 2 to 6 real object scans from the testing dataset (100 scenes from each type) and labeled them with the known object category, as shown in Figure 12. This way we can quantitatively evaluate the effect of scene complexity on the results, as shown in Figure 13. Since in this test unknown objects are not considered, the features incorporating color outperform GRSD-.

The generated scenes do not contain occlusions, but the results are indicative about the performance drop when more and more false groupings are considered by our method. Considering more than 6 touching objects should affect the results less and less, as the number of parts that are grouped is limited. Best results on the real scenes were obtained for 3-4 parts being considered [29].

7 Geometric Verification and Model Fitting

Since the geometric categorization of parts does not give the correct grouping of these parts to form objects, simply grouping the parts of the same category



(a) Success Rates (b) GRSD-

(c) C^3-HLAC (d) VOSCH

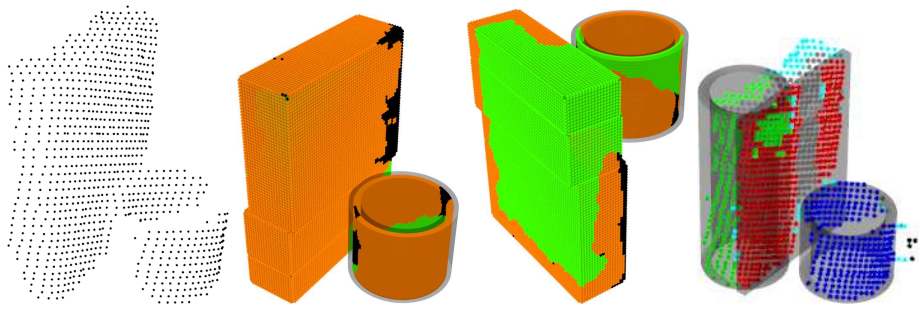
Fig. 13. Per-segment results on the 600 generated scenes from test scans

together does not always separate the objects, especially if classification errors occur as well.

However, the categorization method was already successfully employed to pre-segment scenes and to signal the presence of remaining under-segmented parts to an interactive segmentation system [16]. In case under-segmentation is likely[3], we could exploit the capability of the robot to interact with its environment and perform changes that ease the perception task, as it was done earlier in the case of doors and drawers [7], and more recently for household objects [5] (where we tracked parts that move together when pushed, and thus individuate the objects). Before this, however, fitting of object models should be attempted in order to disambiguate, verify and correct the segmentation. We described a method for voting for object centroids followed by a model fitting step in [36], but that would require CAD models of the objects known a priori.

[3] The number of parts forming certain object types was found to follow a Poisson distribution, and this can be used to judge if a group of parts of the same geometric category forms a single object or not.

Therefore, here we focus on obtaining a grouping of parts into objects by geometric fitting and grouping. We extend the model reconstruction method from [15] to use the obtained probability distributions over the geometric labels as priors when selecting the type of object. The work deals only with upright rectangular and cylindrical shapes for now, but this covers a large percentage of objects in our household table-top settings, as discussed earlier and in [31].



(a) (b) (c) (d)

Fig. 14. Reconstruction results using [15] for the two touching objects in (a). Regions of the final reconstructed surface are labeled: marking verified (orange), invalid (black) and occluded parts (green), as shown in (b) and in the back-side view (c). Even with these verifications, in some cases objects are incorrectly reconstructed, without violating the visibility checks: the cereal box is approximated by a flat box and a cylinder in (d).

The method works by selecting the geometric model that best explains the data in a combination of RANSAC and Hough voting. There are several checks to filter out bad models, but as seen in Figure 14 these not always suffice. We observed these bad fits in out table-top scans as well, as shown in the top row of Figure 15. Thus we used the categorization results to better guide the model selection, by incorporating the shape probabilities as weights into the sampling (which used to be random) and the scoring as well. To match the methods capabilities, the categories were collapsed into rectangular and round shapes. As shown in the bottom row of Figure 15, this is able to correct some of the inaccuracies of the fitting.

Figure 16 presents a comparison of the part grouping results obtained by this geometric verification to the naive approach mentioned at the beginning of the section. As we can see, the objects are correctly grouped together from their parts, but there are still some segments missing which were not inliers to the fitted models. This is either due to incorrect fitting, or because they are not such simple shapes, but handles for example. Handling such cases is difficult, as each heuristic might fail in some cases. Nonetheless, most of the objects are individuated, and have a geometric model. By reconstructing completed 3D models that include the back side as well, the robot is equipped with the necessary information to grasp the objects (reactively, and avoiding the regions labeled as invalid) instead of pushing them for individuation. This way the segmentation and the models could be checked and improved. That constitutes a completely new research direction, however, so we will focus here on the passive observation of the scene, but incorporate scans from multiple viewpoints.

Fig. 15. Geometric modeling results. Top: the three test scenes from Figure 8. Middle: original method [15] with fitting errors marked with red. Bottom: correct reconstruction using the priors from the categorization.

Fig. 16. The parts from the example scene (left) are grouped into objects using the naive grouping based on the categorization results (middle), and using the proposed model fitting (right).

8 Incorporating Multiple Views

Since we found that the highest votes are close to each other, additional information is needed for choosing the correct label. As hinted in [36], this extra information could come from a second scan of the scene from a new viewpoint. Because the preprocessing part of our pipeline takes the most time, in order to obtain results while the robot is scanning, we have to employ the improvements presented in 8.2.

8.1 Accumulating Results

Here, the advantage of incorporating multiple views is evaluated on six views of a scene. We used GRSD- and the "Small+VDS" dataset combination for training, as that performed best in our earlier experiments. As the robot is calibrated, all the scans can be placed into the same coordinate system, with only small misalignments (that could be fixed by an Iterative Closest Point algorithm). Then a 5 mm voxel grid was used to assign points from different frames to each other. The votes were accumulated for each voxel, and a per-point success rate is calculated both for the individual frames, and for the merged RGBD point cloud, presented in Figure 17.

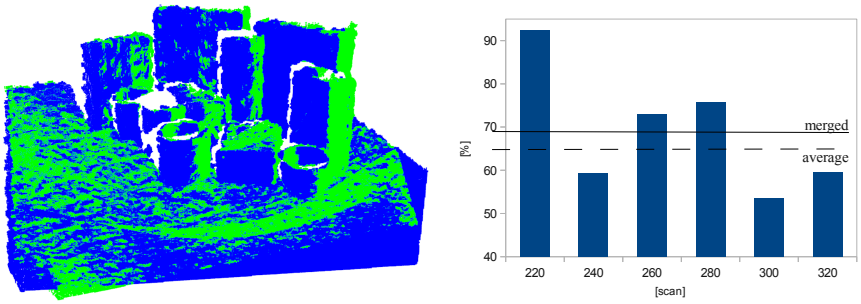


Fig. 17. Left: a moving camera captures multiple frames that cover different parts of the objects in the scene. Right: results for a cluttered scene with 7 frames from multiple viewpoints (denoted by angles around the table's normal).

The robot's end-effector was pointing the camera towards the scene while moving along a circle that respects the minimum range requirement. Still, some of the scenes were not captured fully, or not from an optimal angle, so large variations in accuracy can be observed (as large regions get a good or bad label). By incorporating multiple views however, the overall success rate could be improved by nearly 5%.

An interesting aspect would be to combine results obtained by different features (as evaluated in [34]) or different segmentations in a stacking approach for ensemble learning. We will explore this topic further on the basis of multiple labeled scenes. However, as suggested by [25], voting seems to be the most robust choice for creating ensembles[4].

8.2 Real-time Preprocessing

In order to achieve real-time over-segmentation that is similar to the one used for training, we rely on mean-shift segmentation that processes the normal space computed of the acquired depth images, as shown in Figure 18. We estimate a surface normal for every valid point in the input depth image, and treat the resulting normal map as an RGB image by scaling the normal coefficients from their original range $\langle -1, 1 \rangle$ to fit into $\langle 0, 1 \rangle$, and assigning the x, y and z components to the red, green and blue image channels, respectively.

It is desirable to compute the normals from small point neighborhoods to decrease processing time, however normal vectors estimated from very small neighborhoods suffer from the sensor noise and discretization issues inherent in our acquisition devices. We therefore first perform a smoothing step in disparity, followed by a clamping step that assures that the smoothed disparity value does not deviate further from the measured disparity than the sensor model would allow. These two steps (smoothing, clamping) are alternated for a low number of iterations (e.g. 5 to 10 iterations proved very effective in our experiments), and can be implemented very efficiently in CUDA to harness the computing power of modern

[4] They found that trainable combination methods (like stacking) performed better than voting on the dataset partition which they were trained on, but obtained worse results on a second partition, thus voting generalizes better.

GPU architectures. The effect is that the depth of each point p is influenced by its pixel neighborhood, where the effect of a neighboring point q on p decreases exponentially with pixel distance in the image. Thus, estimating the surface normal at p from its 4 direct neighbors (if present) actually incorporates the depth information of a larger support region that would be far more expensive to compute directly. In our experiments, 7 iterations of edge-aware smoothing and clamping for a full Kinect frame (VGA resolution) can be performed in under $3ms$ on a NVIDIA GTX 560, and the normal estimation step performs in under $0.15ms$. This is in contrast to a kd-tree based CPU implementation, which can take in the order of $500ms$ to one second, depending on neighborhood size and scene geometry.

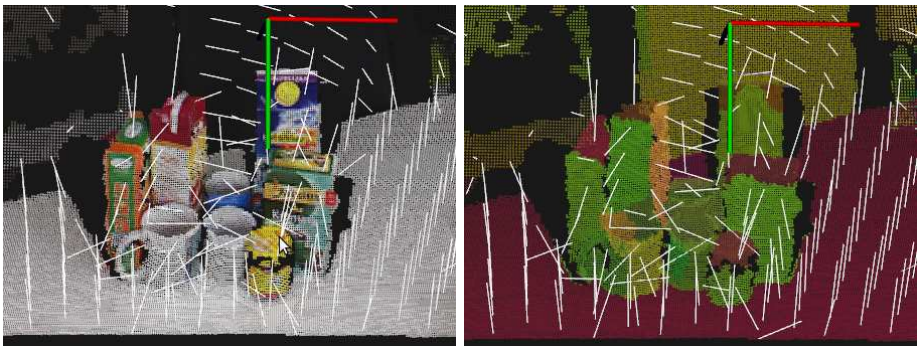


Fig. 18. Kinect frame with visualized normals and same scene segmented.

The mean-shift segmentation [8] of the normal image is also implemented on the GPU, and groups pixels into contiguous regions that share a common surface orientation. If the background and foreground at a depth discontinuity have the same normal orientation (e.g. the top side of a box standing on a table), they can be grouped into the same cluster, however this can be easily remedied by subsequently breaking regions apart at these jump edges. It is also thinkable to incorporate depth directly into the mean-shift segmentation. This segmentation step requires between 20 and $40ms$, depending on the scene (large, flat surfaces vs. finely detailed, complicated geometry), which means we can create the required over-segmentation at near frame rate. A detailed analysis of the segmentation approach falls outside the scope of this work and will be addressed in a separate topical publication.

9 Conclusion and Future Work

In this paper we have shown the advantages of exploiting multiple frames and part-graph descriptors to deal with object categorization in clutter. The proposed methods were evaluated on a large RGBD dataset, and on Kinect scans of cluttered tabletop scenes. They showed promising results when compared to alternative approaches.

The advantage of geometric features was shown for the cases when testing objects that are very different from the trained ones needed to be categorized. Using the CUDA implementation and thanks to hashing, it is possible to obtain

classification results fast for complete scenes. This allows the merging of results coming from multiple views of the same scene in order to improve detection.

As we can produce multiple segmentations by choosing different (random) seed points, different part decompositions can be used for training, which improved classification rates by 5% in our early experiments [3].

Most importantly, the inclusion of a more advanced geometric grouping method needs to be considered, for example by combining this work with the one presented in [39] (which is currently most suitable for convex shapes).

Future work will focus on quantifying the effect of occlusions, the development of a more descriptive additive geometric feature, and more advanced domain adaptation. Employing stronger classifiers than nearest-neighbors with our hashing method could also improve results.

Acknowledgements This work is supported in part by the EU FP7 projects *RoboHow* (grant number 288533) *SAPHARI* (grant number 287513) and *ACAT* (grant number 600578) The authors would like to thank Florian Seidel, Ronny Bismark and Kevin Lai.

References

1. Aldoma, A., Blodow, N., Gossow, D., Gedikli, S., Rusu, R., Vincze, M., Bradski, G.: CAD-Model Recognition and 6 DOF Pose Estimation Using 3D Cues. In: ICCV Workshop on 3D Representation and Recognition (3dRR11). Barcelona, Spain (2011)
2. Aldoma, A., Marton, Z.C., Tombari, F., Wohlkinger, W., Potthast, C., Zeisl, B., Rusu, R.B., Gedikli, S., Vincze, M.: Tutorial: Point Cloud Library – Three-Dimensional Object Recognition and 6 DoF Pose Estimation. Robotics & Automation Magazine **19**(3), 80–91 (2012)
3. Balint-Benczedi, F., Marton, Z.C., Beetz, M.: Efficient part-graph hashes for object categorization. In: 5th International Conference on Cognitive Systems (CogSys) (2012)
4. Bergström, N., Björkman, M., Kragic, D.: Generating Object Hypotheses in Natural Scenes through Human-Robot Interaction. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 827–833 (2011)
5. Bersch, C., Pangercic, D., Osentoski, S., Hausman, K., Marton, Z.C., Ueda, R., Okada, K., Beetz, M.: Segmentation of textured and textureless objects through interactive perception. In: RSS Workshop on Robots in Clutter: Manipulation, Perception and Navigation in Human Environments. Sydney, Australia (2012)
6. Biederman, I.: Recognition-by-components: A theory of human image understanding. Psychological Review (1987)
7. Blodow, N., Goron, L.C., Marton, Z.C., Pangercic, D., Rühr, T., Tenorth, M., Beetz, M.: Autonomous semantic mapping for robots performing everyday manipulation tasks in kitchen environments. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). San Francisco, CA, USA (2011)
8. Comaniciu, D., Meer, P., Member, S.: Mean shift: A robust approach toward feature space analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence **24**, 603–619 (2002)
9. Dickinson, S.: The evolution of object categorization and the challenge of image abstraction. In: S. Dickinson, A. Leonardis, B. Schiele, M. Tarr (eds.) Object Categorization: Computer and Human Vision Perspectives (2009)
10. Felzenszwalb, P.F., Girshick, R.B., McAllester, D.A., Ramanan, D.: Object detection with discriminatively trained part-based models. IEEE Trans. Pattern Anal. Mach. Intell. **32**(9) (2010)
11. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. Int. J. Comput. Vision **59**(2), 167–181. DOI http://dx.doi.org/10.1023/B:VISI.0000022288.19776.77
12. Fergus, R., Perona, P., Zisserman, A.: Object class recognition by unsupervised scale-invariant learning. In: In CVPR, pp. 264–271 (2003)

13. Fowlkes, C.C., Martin, D.R., Malik, J.: Local figure-ground cues are valid for natural images. Journal of Vision **7**(8) (2007)
14. Gomes, R., Krause, A., Perona, P.: Discriminative clustering by regularized information maximization. Advances in Neural Information Processing Systems 23 pp. 1–9 (2010)
15. Goron, L.C., Marton, Z.C., Lazea, G., Beetz, M.: Segmenting cylindrical and box-like objects in cluttered 3D scenes. In: 7th German Conference on Robotics (ROBOTIK 2012). Munich, Germany (2012)
16. Hausman, K., Balint-Benczedi, F., Pangercic, D., Marton, Z.C., Ueda, R., Okada, K., Beetz, M.: Tracking-based interactive segmentation of textureless objects. In: IEEE International Conference on Robotics and Automation (ICRA). Karlsruhe, Germany (2013). Best Service Robotics Paper Award Finalist
17. Huber, D., Kapuria, A., Donamukkala, R.R., Hebert, M.: Parts-based 3d object classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 04) (2004)
18. Jacobs, D.W.: Perceptual Organization As Generic Object Recognition. In: From Fragments to Objects - Segmentation and Grouping in Vision, chap. IV. Models Of Segmentation And Grouping, pp. 295–329 (2001)
19. Kanezaki, A., Marton, Z.C., Pangercic, D., Harada, T., Kuniyoshi, Y., Beetz, M.: Voxelized Shape and Color Histograms for RGB-D. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Workshop on Active Semantic Perception and Object Search in the Real World. San Francisco, CA, USA (2011)
20. Kanezaki, A., Nakayama, H., Harada, T., Kuniyoshi, Y.: High-speed 3d object recognition using additive features in a linear subspace. In: Proc. of International Conference on Robotics and Automation (ICRA), pp. 3128–3134 (2010)
21. Kanezaki, A., Suzuki, T., Harada, T., Kuniyoshi, Y.: Fast object detection for robots in a cluttered indoor environment using integral 3D feature table. In: Proc. IEEE ICRA (2011)
22. Lai, K., Bo, L., Ren, X., Fox, D.: A large-scale hierarchical multi-view rgb-d object dataset. In: Proc. of International Conference on Robotics and Automation (ICRA) (2011)
23. Lai, K., Bo, L., Ren, X., Fox, D.: Sparse distance learning for object recognition combining rgb and depth information. In: Proc. of International Conference on Robotics and Automation (ICRA) (2011)
24. Lai, K., Fox, D.: Object recognition in 3d point clouds using web data and domain adaptation. The International Journal of Robotics Research **29**(8), 1019–1037 (2010). URL `http://ijr.sagepub.com/cgi/doi/10.1177/0278364910369190`
25. Lam, L., Suen, C.Y.: Optimal combinations of pattern classifiers. Pattern Recognition Letters **16**(9), 945–954 (1995)
26. Leibe, B., Leonardis, A., Schiele, B.: Robust object detection with interleaved categorization and segmentation. International journal of computer vision **77**(1-3), 259–289 (2008)
27. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision **60**, 91–110 (2004)
28. Malisiewicz, T., Efros, A.A.: Improving Spatial Support for Objects via Multiple Segmentations. In: Proceedings of the British Machine Vision Conference (2007)
29. Marton, Z.C., Balint-Benczedi, F., Blodow, N., Goron, L.C., Beetz, M.: Object categorization in clutter using additive features and hashing of part-graph descriptors. In: Proceedings of Spatial Cognition 2012. Abbey Kloster Seeon, Germany (2012)
30. Marton, Z.C., Balint-Benczedi, F., Mozos, O.M., Pangercic, D., Beetz, M.: Cumulative Object Categorization in Clutter. In: 2nd Workshop on Robotics in Clutter, at Robotics: Science and Systems (RSS) (2013)
31. Marton, Z.C., Pangercic, D., Blodow, N., Beetz, M.: Combined 2D-3D Categorization and Classification for Multimodal Perception Systems. The International Journal of Robotics Research (2011)
32. Marton, Z.C., Pangercic, D., Rusu, R.B., Holzbach, A., Beetz, M.: Hierarchical object geometric categorization and appearance classification for mobile manipulation. In: Proceedings of 2010 IEEE-RAS International Conference on Humanoid Robots. Nashville, TN, USA (2010)
33. Marton, Z.C., Rusu, R.B., Jain, D., Klank, U., Beetz, M.: Probabilistic Categorization of Kitchen Objects in Table Settings with a Composite Sensor. In: Proceedings of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems. USA (2009)
34. Marton, Z.C., Seidel, F., Balint-Benczedi, F., Beetz, M.: Ensembles of Strong Learners for Multi-cue Classification. Patt. Rec. Letters, Special Issue on Scene Understandings and Behaviours Analysis (2012)

35. Mishra, A.K., Aloimonos, Y.: Visual Segmentation of "Simple" Objects for Robots. In: Robotics: Science and Systems (RSS) (2011)
36. Mozos, O.M., Marton, Z.C., Beetz, M.: Furniture Models Learned from the WWW – Using Web Catalogs to Locate and Categorize Unknown Furniture Pieces in 3D Laser Scans. Robotics & Automation Magazine **18**(2), 22–32 (2011)
37. Mueller, C.A., Pathak, K., Birk, A.: Object recognition in rgbd images of cluttered environments using graph-based categorization with unsupervised learning of shape parts. In: International Conference on Intelligent Robots and Systems (IROS) (2013)
38. Pangercic, D., Haltakov, V., Beetz, M.: Fast and robust object detection in household environments using vocabulary trees with sift descriptors. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Workshop on Active Semantic Perception and Object Search in the Real World. San Francisco, CA, USA (2011)
39. Richtsfeld, A., Morwald, T., Prankl, J., Zillich, M., Vincze, M.: Segmentation of unknown objects in indoor environments. In: Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on, pp. 4791–4796 (2012). DOI 10.1109/IROS.2012.6385661
40. Ruiz-Correa, S., Shapiro, L.G., Meila, M.: A new paradigm for recognizing 3-D object shapes from range data. In: Int. Conf. on Computer Vision (2003)
41. Rusu, R.B., Blodow, N., Marton, Z.C., Beetz, M.: Close-range Scene Segmentation and Reconstruction of 3D Point Cloud Maps for Mobile Manipulation in Human Environments. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). St. Louis, MO, USA (2009)
42. Rusu, R.B., Cousins, S.: 3D is here: Point Cloud Library (PCL). In: IEEE International Conference on Robotics and Automation (ICRA). Shanghai, China (2011)
43. Rusu, R.B., Marton, Z.C., Blodow, N., Dolha, M., Beetz, M.: Towards 3D Point Cloud Based Object Maps for Household Environments. Robotics and Autonomous Systems Journal (Special Issue on Semantic Knowledge in Robotics) **56**(11), 927–941 (2008)
44. Scavino, E., Wahab, D.A., Basri, H., Mustafa, M.M., Hussain, A.: A genetic algorithm for the segmentation of known touching objects **5**, 711–716 (2009)
45. Schnabel, R., Wahl, R., Klein, R.: Efficient ransac for point-cloud shape detection. In: Computer Graphics Forum, vol. 26, pp. 214–226. Wiley Online Library (2007)
46. Schuster, M., Okerman, J., Nguyen, H., Rehg, J., Kemp, C.: Perceiving clutter and surfaces for object placement in indoor environments. In: Proceedings of the 10th IEEE-RAS International Conference on Humanoid Robots (Humanoids), pp. 152 –159 (2010). DOI 10.1109/ICHR.2010.5686328
47. Shotton, J., Blake, A., Cipolla, R.: Multiscale categorical object recognition using contour fragments. IEEE Transactions on Pattern Analysis and Machine Intelligence **30**(7) (2008)
48. Shotton, J., Winn, J., Rother, C., Criminisi, A.: Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context. International Journal of Computer Vision (2007)
49. Singh, M., Hoffman, D.D.: Part-Based Representations Of Visual Shape And Implications For Visual Cognition. In: From Fragments to Objects - Segmentation and Grouping in Vision, chap. IV. Models Of Segmentation And Grouping, pp. 401–459 (2001)
50. Socher, R., Lin, C.C.Y., Ng, A.Y., Manning, C.D.: Parsing natural scenes and natural language with recursive neural networks. In: 28th International Conference on Machine Learning, pp. 129–136 (2011)
51. Somanath, G., Rohith, M., Metaxas, D., Kambhamettu, C.: D - clutter: Building object model library from unsupervised segmentation of cluttered scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2783 –2789 (2009). DOI 10.1109/CVPR.2009.5206579
52. Sun, M., Bradski, G., Xu, B.X., Savarese, S.: Depth-encoded hough voting for joint object detection and shape recovery. In: Proceedings of the 11th European conference on Computer vision: Part V, ECCV'10, pp. 658–671. Springer-Verlag, Berlin, Heidelberg (2010). URL `http://dl.acm.org/citation.cfm?id=1888150.1888201`
53. Taylor, G., Kleeman, L.: Chapter 4: 3D Object Modelling and Classification. In: Visual Perception and Robotic Manipulation – 3D Object Recognition, Tracking and Hand-Eye Coordination, *Springer Tracts in Advanced Robotics*, vol. 26, pp. 57–83. Springer Berlin Heidelberg (2006)
54. Tombari, F., Di Stefano, L.: Object recognition in 3d scenes with occlusions and clutter by hough voting. In: Proceedings of the Fourth Pacific-Rim Symposium on Image and Video Technology (PSIVT), pp. 349 –355 (2010). DOI 10.1109/PSIVT.2010.65

55. Torres, M.M., Romea, A.C., Srinivasa, S.: MOPED: A Scalable and Low Latency Object Recognition and Pose Estimation System. In: Proc. of International Conference on Robotics and Automation (ICRA) (2010)
56. Triebel, R., Shin, J., Siegwart, R.: Segmentation and unsupervised part-based discovery of repetitive objects. In: Proceedings of Robotics: Science and Systems. Zaragoza, Spain (2010)
57. Vernon, D.: Cognitive vision: The case for embodied perception. In: Image and Vision Computing. Elsevier (2005)
58. Watanabe, S., Pakvasa, N.: Subspace method in pattern recognition. In: Proc. of 1st International Joint Conf. on Pattern Recognition (1973)
59. Yildirim, I., Jacobs, R.A.: Transfer of object category knowledge across visual and haptic modalities: experimental and computational studies. Cognition **126**(2), 135–148 (2013). DOI 10.1016/j.cognition.2012.08.005. URL `http://dx.doi.org/10.1016/j.cognition.2012.08.005`