Cumulative Object Categorization in Clutter

Zoltan-Csaba Marton*, Ferenc Balint-Benczedi†, Oscar Martinez Mozos‡, Dejan Pangercic§ and Michael Beetz†

*Institute of Robotics and Mechatronics German Aerospace Center (DLR), Germany *zoltan.marton@dlr.de*

†Institute of Artificial Intelligence, Unversity of Bremen, part of Centre for Computing Technologies (TZI) {*balintbe,beetz*}@*tzi.de*

‡School of Computer Science University of Lincoln United Kingdom, *omozos@lincoln.ac.uk*

§Autonomous Technologies Group Robert Bosch LLC, United States *dejan.pangercic@gmail.com*

***Abstract*— In this paper we present an approach based on scene- or part-graphs for geometrically categorizing touching and occluded objects. We use additive RGBD feature descriptors and hashing of graph configuration parameters for describing the spatial arrangement of constituent parts. The presented experiments quantify that this method outperforms our earlier part-voting and sliding window classification. We evaluated our approach on cluttered scenes, and by using a 3D dataset containing over 15000 Kinect scans of over 100 objects which were grouped into general geometric categories. Additionally, color, geometric, and combined features were compared for categorization tasks.**

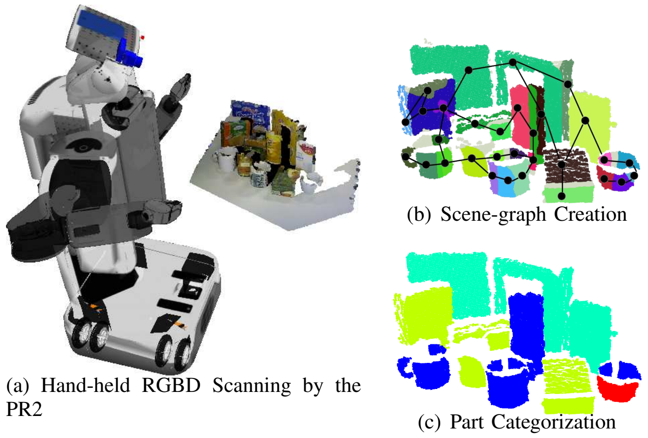
I. Introduction

This paper considers categorization of previously unknown objects in cluttered scenes, where accurate segmentation can be difficult to achieve, as the objects are touching, occluding each other. In the context of robotic perception, additional robustness to varying lighting conditions and to multiple similar objects having no unique texture is required. For such tasks, RGBD camera based approaches are a promising addition to the repertoire of image understanding. Since a household assistant could encounter new objects during its operation, no matter how large a training database is, geometric (edge-based or 3D) categorization and perceptual grouping can be an important step before template-based (image processing) approaches can be applied for instance-level recognition [1, 2]. In our previous [3] work we proposed a method to detect and categorize possible object parts in cluttered scenes based on their shape. Its steps are shown in Figure 1 and detailed in III. The main idea is to over segment the scene into parts, and decide what kind of object do they form, based on the arrangement of its parts. Image-based approaches often fail for textureless objects, or under bad lighting, as seen in Figure 2. Therefore we perform 3D-based geometric categorization in a recognition-by-components approach.

In this paper we complement our findings in [3] focusing on testing RGB and RGBD features, comparisons to alternative approaches, quantitative evaluations, and enabling the robot to accumulate information about the scene. We also validate the choice of our geometric categories. Evaluation was performed on RGBD scans of cluttered tabletop scenes of previously unknown objects, and we experimented with enriching the training set by combining different databases.

II. Related Work

As discussed in [5], 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 [6], part-based detection has the advantage of generalizing to unknown instances of object types. While in [6], [7] and for the part-based VFH (called CVFH) feature [8], objects need to be separated first, approaches like [9, 10] can efficiently detect objects in clutter. This typically requires over-segmenting the scene, possibly multiple times, while respecting object boundaries. 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 [11, 12].

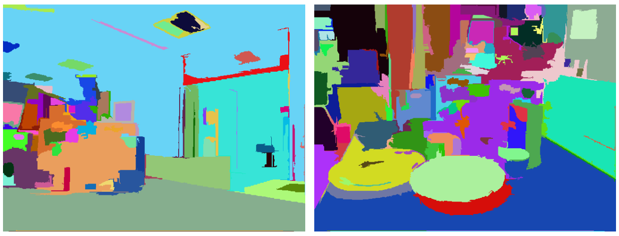


(a) Hand-held RGBD Scanning by the PR2

(b) Scene-graph Creation

(c) Part Categorization

1. Overview of the process of scanning, segmenting and categorizing objects in clutter. In the final result cylinders are marked with blue, boxes with yellow, rectangular flat faces with cyan, and (half) spheres with red.



2. Image-based segmentation results of cluttered scenes like in Fig. 1 using [4]

Our approach is similar to the one presented by Felzenszwalb in [13], 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 unsupervisedly identified parts by a classifier. Shotton *et al.* [14] incorporate poses and viewpoints, texture, layout, and context information for image segmentation based object recognition. In a complementing publication [15] 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 [9] the authors also propose a similar system for understanding cluttered scenes. Our approach combines the over-segmentation from [10] with an extended version of creating multiple groupings of these "parts" [9], and was designed to handle multiple instances of objects from several categories, that were labeled according to their general 3D shape. While in [10] 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 [7]. Identifying to what object does each part belong to, consists of considering its descriptor (and that of neighboring parts), together with the local topology of the scene. Thus it improves on the vocabulary of parts and simple vote accumulating approach from [10]. Furthermore, this work focuses on objects relevant to pick and place tasks, which have 6 degrees of freedom poses instead of 3 as furniture pieces.

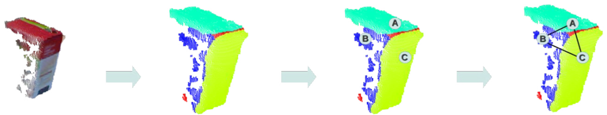
Recently, Richtsfeld *et al.* [16] presented a multi-level approach to fit planar or curved surfaces to over-segment parts, and then define inter-segment relations to decide if they should be merged or not. Unlike our approach, they consider relations between non-touching parts as well, but the method performs best for merging touching segments and for convex shapes. Other approaches also focus on creating surface models by fitting shape primitives or superquadrics and considering the spacial relations between them [17, 18, 19], but in slightly simpler scenarios.

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 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.* [21]. We used the additive property of 3 features (GRSD- [3], C^3-HLAC [22] and VOSCH [23]) to compute the descriptor of grouped parts by summing up the parts' descriptors, and here we compare our method to that presented in [20] and [10].

III. Part-graph Hashing Based Recognition

In our previous work [7, 10, 3] 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 [3]) is that segmenting objects accurately does not always work robustly and will result in labeling mistakes, but over-segmentation is easily realizable [24, 9]. 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 3.

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. 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 formed by the parts of an object. 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 features. Each sub-graph has an "arrangement key", which in our case is formed of the degrees of its nodes, and this can be used for hashing them into several categories before classification. Therefore we can avoid confusions between subgraphs that don't have the same number of nodes and speed up training/testing.



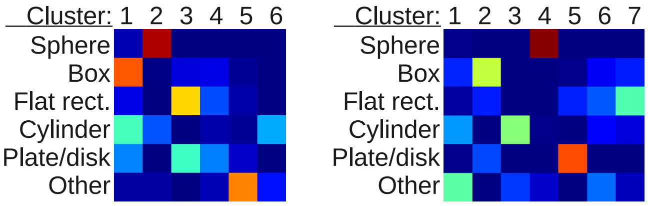
Groupings	Arrangement Key	Descriptors
A	1	$d_a = (f_a^1, f_a^2, f_a^3 ... f_a^n)$
B	1	$d_b = (f_b^1, f_b^2, f_b^3 ... f_b^n)$
C	1	$d_c = (f_a^1, f_a^2, f_c^3 ... f_a^n)$
B A	11	$d_{ab} = d_a + d_b$
A C	11	$d_{ac} = d_a + d_c$
B C	11	$d_{bc} = d_b + d_c$
A B C	222	$d_{abc} = d_{ab} + d_c$

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

As reviewed in [12], there are certain principles that should guide the search for perceptually salient parts. We rely in this work on the "hypothesis of normalized curvature" and the "hypothesis of turning angle". The segmentation criteria used to over-segment the scans is presented in [10], 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 segments, and over 100 groupings of parts.

When processing a test scene, the same segmentation and hashing procedure is repeated for the query, and the part groupings' features are classified. The obtained probability distributions are accumulated in the constituent parts, giving lower weight to larger groups. In contrast to [9], where the product of the class probabilities for each grouping was used, we found that the (confidence weighted) voting approach performs better. Similar findings supporting voting were made in [25] when evaluating combinations of classification results.

The 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* category), and also the categories we found in public household objects databases [2]. 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.



4. Unsupervised RIM clustering compared to the manually defined geometric categories (left: GRSD-, right: VFH). Clusters overlap well with the used categories, with two geometrically similar pairs merged using GRSD-. However, in the higher dimensional VFH feature space these can be distinguished.

We used the Regularized Information Maximization (RIM) technique [26] to find meaningful clusters of our training data and assign testing instances to these clusters in the GRSD- and VFH feature spaces. We measured how well do the clusters overlap with the given categories by computing the Adjusted Rand Index (ARI), using different parameters.

For GRSD- the best ARI (0.36) is obtained using 6 clusters and $\lambda = 90$, with stable results around these values. As shown in Figure 4 (left), the clusters are quite clean, and also the categories are grouped nicely with clusters, but cylindrical objects were merged with boxes and flat ones with plates. 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 and provide relevant information for model fitting and grasping applications.

Using VFH these clusters could be separated, thanks to the increased descriptiveness given by the higher dimensionality and viewpoint variance. Here the best obtained ARI (0.42) is obtained using 7 or 8 clusters and λ between 75-80, but the results are not as stable as for GRSD-, suggesting that the clustering depends very much on the random initialization.

In both cases, smaller clusters are created as well, 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 [2] to increase the geometric categorization accuracy.

Since we label only parts now, future work will focus on obtaining a grouping of parts into objects by geometric fitting and grouping. We plan to extend fitting methods to use the geometric labels as priors when selecting models for fitting. The method was already successfully employed to pre-segment scenes and to signal the presence of remaining under-segmented parts to an interactive segmentation system [27]. The robot's manipulation capability was used to track parts that move together when pushed, thus individuating objects.

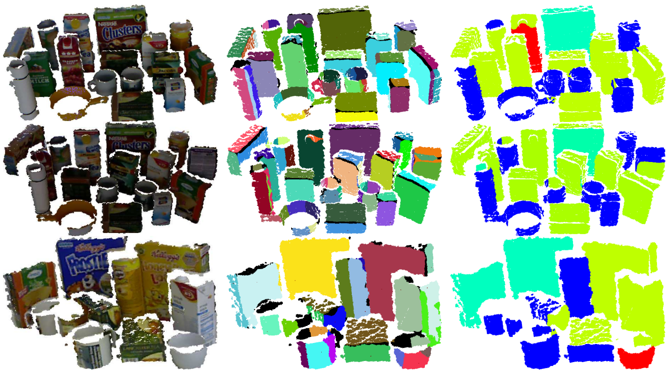
IV. Evaluation and Discussion

For our tests we used a part of the large RGBD dataset from [28]. As in [28], we use every fifth point cloud, 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 as described in [3] ("RGBD-Large"). 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 ("RGBD-Small") [3]. Additionally, we used the dataset from [23] to add knowledge about the objects in our environment.

A. Complete Cluttered Scenes

As labeling scenes is a time-consuming process, we could evaluate only a couple of them, extending the results from [3]. 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 5 show three tabletop scenes on which we tested our approach. The color red represents the *sphere* class, blue *cylinders*, yellow *boxes*, and cyan the *flat* class.

Testing on the cluttered scenes was run using different datasets (or combinations) as training data, as shown in Tables I and II. As it is expected, results vary depending on the type of feature descriptor and on the training dataset.



5. Segmentation and geometric categorization on three cluttered scenes.

In order to diversify our training data we combined the RGBD datasets with the "VOSCH" Kinect scan dataset (VDB) used in [23], consisting of 63 similar objects to the ones in our scenes, captured from different viewpoints with an angular step of 15 degrees. Similarly to [9], we found that this "domain adaptation" improves results, as seen in Table I. 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[1]. Related ideas are discussed by Horswill et.al. [29] as well (task and environment adaptation improving perception capabilities). Another interesting observation is that upon combining the datasets the per point result improve much more then the per segment ones. This is due to the fact that the parts resulting from flat and box like objects consist of a greater number of points then those that come from the other categories, and that for these parts in general we have better classification results.

Average success rates	RGBD-Small	RGBD-Large	VDB	Small+ VDB	Large+ VDB
per point	73%	48%	75%	**84%**	61%
per segment	78%	45%	74%	**79%**	59%

I. Results in clutter using different training datasets with GRSD-

Average success rates	RGBD-Small	RGBD-Large	VDB	Small+ VDB	Large+ VDB
per point	43%	48%	**67%**	62%	59%
per segment	46%	46%	**69%**	57%	50%

II. 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 [28]. 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 II. 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). We believe that the contradicting results are due to the fact that in [28] 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 classification times shown in Table III were obtained on a single core 2.4 GHz CPU.

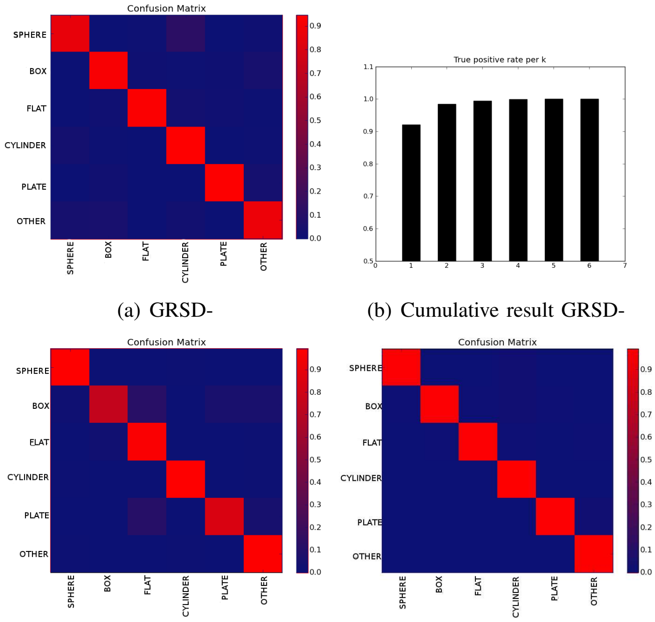
For a more detailed evaluation, the next subsections will focus on large scale tests using the RGBD dataset, using separated objects as queries. The RGBD-Small set was split 2:1 into a training and testing scans [3], except for the cross-validation test that was performed using the methodology from [28]. Given separated objects, we 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.

Runtimes using diff. datasets	RGBD-Small	RGBD-Large	VDB	Small+ VDB	Large+ VDB
GRSD- [20d]					
per point	0.24E-04	0.44E-0.4	0.041E-04	0.28E-04	0.47E-04
per segment	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 segment	0.27	0.43	0.03	0.30	0.47

III. Average classification times in seconds for the scenes from Figure 5

B. *Evaluation of Features*

In our earlier work we tested different distance metrics for nearest neighbors classification and found that the Jeffries-Matsushita distance performs best. 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, but using a simple nearest neighbors approach, which has considerably shorter training time [3].



(a) GRSD-

(b) Cumulative result GRSD-

(c) VOSCH

(d) C^3-HLAC

6. Confusion matrices and cumulative score on the RGBD-Large set.

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. Results are shown in Figure 6, with an interesting observation relating to (b): the two most likely results are by 5% better than the ones reported as most likely. This suggests that in case we obtain similar top scores, re-segmenting the test scene (with different random seeds) could improve the labeling, by

[1] Thanks to the hashing approach, handling large databases and dynamically adding new objects is alleviated, as only affected groups have to be re-trained.

merging the votes from different segmentations. This approach was employed in the next section in the case of different views.

We also performed a cross validation experiment to test how well these additive features generalize to unknown objects. See Table IV 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%.

	Sphere [%]	Box [%]	Flat [%]	Cylinder [%]	Plate [%]	Other [%]
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
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
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

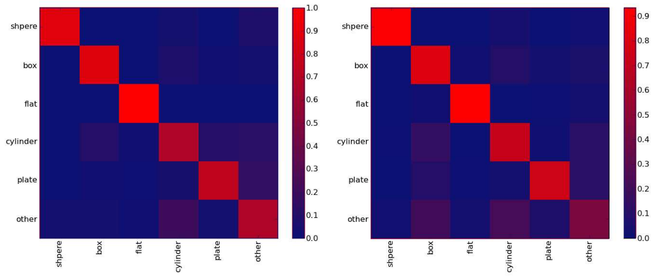
IV. Per class leave-one-out cross validation tests on the RGBD-Small set

C. Comparison to Previous Methods

In our previous work [3] we performed a comparison to segmentation-based categorization, by segmenting round and rectangular objects using the method from [30], 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 [10], 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.

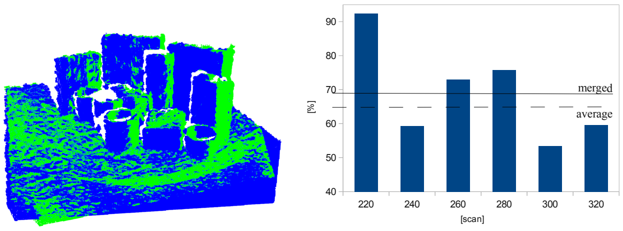
Both the statistical features and GRSD- were tested using this method, and we obtained a mean success rate of 80.45% for the former and 75.86% for the latter. As seen from the corresponding confusion matrices in Figure 7, the difference is due to the fact that the miscellaneous "other" class is handled 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. Moreover, some of the statistical features are orientation dependent, requiring training objects in multiple poses.



(a) Statistical Feature

(b) GRSD-

7. Confusion matrices of the vocabulary of parts method.



8. 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).

Our method and the sliding window based Linear Subspace method was also evaluated on the same data, using the GRSD- descriptor. Overall, the results indicate a clear advantage of the part-based categorization process, as shown in Table V.

	Part-graph Hashing	Part Vocabulary [10]	LSM [20]
Success rate	95.5	75.9	77.8

V. Results using different methods on the RGDB-Small datasets

D. Synthetic Scenes

This subsection presents results on a large scale test on scenes containing touching objects (without occlusions). As ground truth data is difficult to obtain, we generated scenes containing from 2 to 6 object scans from the testing dataset (100 scenes from each type) and labeled them with the known object category. This way we can quantitatively evaluate the effect of scene complexity on the results, as shown in Table VI.

Nr. objects:	2	3	4	5	6
Success rate:	73%	74%	69%	70%	66%

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

The generated scenes do not contain occlusions, but the results are indicating the performance drop as more false groupings are considered. 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 [3].

V. 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 [10], this extra information could come from a second scan of the scene from a new viewpoint.

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 places 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 8.

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 from a non-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 improved by nearly 5%.

An interesting aspect would be to combine results obtained by different features (as evaluated in [31]) 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[2].

VI. 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, and 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.

Most importantly, the inclusion of a geometric grouping method needs to be considered, using or extending some of the existing solutions relying on different assumptions: [10] (using available CAD models), [30] (upright boxes and cylinders), [16] (mostly 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. More powerful classifiers combined with our hashing method could also improve results.

References

[1] S. Dickinson, "The evolution of object categorization and the challenge of image abstraction," in *Object Categorization: Computer and Human Vision Perspectives*, S. Dickinson, A. Leonardis, B. Schiele, and M. Tarr, Eds., 2009.

[2] Z. C. Marton, D. Pangercic, N. Blodow, and M. Beetz, "Combined 2D-3D Categorization and Classification for Multimodal Perception Systems," *The International Journal of Robotics Research*, 2011.

[3] Z.-C. Marton, F. Balint-Benczedi, N. Blodow, L. C. Goron, and M. Beetz, "Object categorization in clutter using additive features and hashing of part-graph descriptors," in *Proceedings of Spatial Cognition 2012*, Abbey Kloster Seeon, Germany, 2012.

[4] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *Int. J. Comput. Vision*, vol. 59, no. 2, pp. 167–181.

[5] D. W. Jacobs, "Perceptual Organization As Generic Object Recognition," in *From Fragments to Objects - Segmentation and Grouping in Vision*, 2001, ch. IV. Models Of Segmentation And Grouping, pp. 295–329.

[6] D. Huber, A. Kapuria, R. R. Donamukkala, and M. Hebert, "Parts-based 3d object classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 04)*, July 2004.

[7] Z. C. Marton, R. B. Rusu, D. Jain, U. Klank, and M. Beetz, "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, October 11-15 2009.

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

[9] K. Lai and D. Fox, "Object recognition in 3d point clouds using web data and domain adaptation," *The International Journal of Robotics Research*, vol. 29, no. 8, pp. 1019–1037, 2010. [Online]. Available: http://ijr.sagepub.com/cgi/doi/10.1177/0278364910369190

[10] O. M. Mozos, Z. C. Marton, and M. Beetz, "Furniture Models Learned from the WWW – Using Web Catalogs to Locate and Categorize Unknown Furniture Pieces in 3D Laser Scans," *Robotics & Automation Magazine*, vol. 18, no. 2, pp. 22–32, 2011.

[11] I. Biederman, "Recognition-by-components: A theory of human image understanding," *Psychological Review*, 1987.

[12] M. Singh and D. D. Hoffman, "Part-Based Representations Of Visual Shape And Implications For Visual Cognition," in *From Fragments to Objects - Segmentation and Grouping in Vision*, 2001, ch. IV. Models Of Segmentation And Grouping, pp. 401–459.

[13] P. F. Felzenszwalb, R. B. Girshick, D. A. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, 2010.

[14] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context," *International Journal of Computer Vision*, 2007.

[15] J. Shotton, A. Blake, and R. Cipolla, "Multiscale categorical object recognition using contour fragments," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 7, July 2008.

[16] A. Richtsfeld, T. Morwald, J. Prankl, M. Zillich, and M. Vincze, "Segmentation of unknown objects in indoor environments," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, 2012, pp. 4791–4796.

[17] M. V. Georg Biegelbauer, "Efficient 3d object detection by fitting superquadrics to range image data for robots object manipulation," *IEEE International Conference on Robotics and Automation (ICRA)*, 2007.

[18] Y. Li, X. Wu, Y. Chrysanthou, A. Sharf, D. Cohen-Or, and N. J. Mitra, "Globfit: Consistently fitting primitives by discovering global relations," *ACM Transactions on Graphics*, vol. 30, no. 4, pp. 52:1–52:12, 2011.

[19] M. Nieuwenhuisen, D. Droeschel, D. Holz, J. Stückler, A. Berner, J. Li, R. Klein, and S. Behnke, "Mobile bin picking with an anthropomorphic service robot," *IEEE Int. Conf. on Rob. and Autom. (ICRA)*, May 2013.

[20] A. Kanezaki, H. Nakayama, T. Harada, and Y. Kuniyoshi, "High-speed 3d object recognition using additive features in a linear subspace," in *Proc. of Int. Conf. on Rob. and Autom. (ICRA)*, 2010, pp. 3128–3134.

[21] S. Watanabe and N. Pakvasa, "Subspace method in pattern recognition," in *Proc. of 1st International Joint Conf. on Pattern Recognition*, 1973.

[22] A. Kanezaki, T. Suzuki, T. Harada, and Y. Kuniyoshi, "Fast object detection for robots in a cluttered indoor environment using integral 3D feature table," in *Proc. IEEE ICRA*, 2011.

[23] A. Kanezaki, Z.-C. Marton, D. Pangercic, T. Harada, Y. Kuniyoshi, and M. Beetz, "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, September, 25–30 2011.

[24] T. Malisiewicz and A. A. Efros, "Improving Spatial Support for Objects via Multiple Segmentations," in *Proceedings of the British Machine Vision Conference*, 2007.

[25] L. Lam and C. Y. Suen, "Optimal combinations of pattern classifiers," *Pattern Recognition Letters*, vol. 16, no. 9, pp. 945–954, 1995.

[26] R. Gomes, A. Krause, and P. Perona, "Discriminative clustering by regularized information maximization," *Advances in Neural Information Processing Systems 23*, pp. 1–9, 2010.

[27] K. Hausman, F. Balint-Benczedi, D. Pangercic, Z.-C. Marton, R. Ueda, K. Okada, and M. Beetz, "Tracking-based interactive segmentation of textureless objects," in *IEEE International Conference on Robotics and Automation (ICRA)*, Karlsruhe, Germany, May 6–10 2013, best Service Robotics Paper Award Finalist.

[28] K. Lai, L. Bo, X. Ren, and D. Fox, "A large-scale hierarchical multi-view rgb-d object dataset," in *Proc. of International Conference on Robotics and Automation (ICRA)*, 2011.

[29] I. Horswill, "Integrating vision and natural language without central models," in *In Proceedings of the AAAI Fall Symposium on Embodied Language and Action*, 1995.

[30] L. C. Goron, Z. C. Marton, G. Lazea, and M. Beetz, "Segmenting cylindrical and box-like objects in cluttered 3D scenes," in *7th German Conference on Robotics (ROBOTIK 2012)*, Munich, Germany, 2012.

[31] Z.-C. Marton, F. Seidel, F. Balint-Benczedi, and M. Beetz, "Ensembles of Strong Learners for Multi-cue Classification," *Patt. Rec. Letters, Special Issue on Scene Understandings and Behaviours Analysis*, 2012.

[2] 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.