Extracting Categories by Hierarchical Clustering Using Global Relational Features

Wail Mustafa(✉), Dirk Kraft, and Norbert Krüger

The Mærsk Mc-Kinney Møller Institute, University of Southern Denmark,
Campusvej 55, 5230 Odense M, Denmark
wail@mmmi.sdu.dk

Abstract. We introduce an object categorization system which uses hierarchical clustering to extract categories. The system is able to assign multiple, nested categories for unseen objects. In our system, objects are represented with global pair-wise relations computed from 3D features extracted by three RGB-D sensors. We show that our system outperforms a state-of-the-art approach particularly when only a few number of training samples is used.

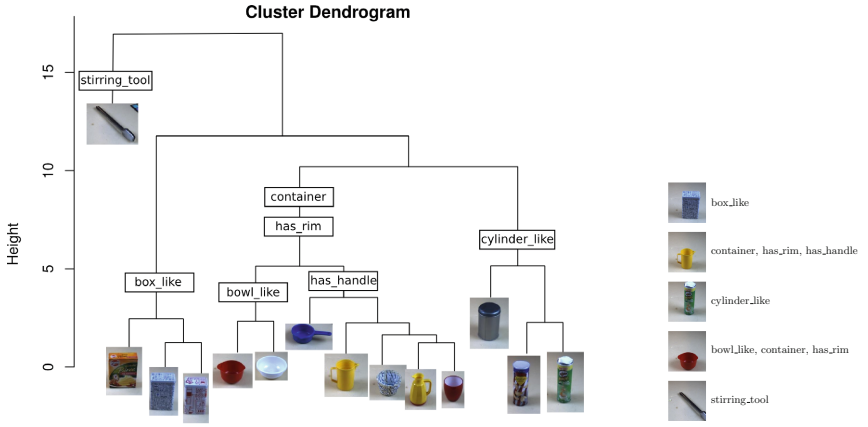
1 Introduction

Object categorization is important for a variety of tasks especially when systems are expected to deal with novel objects based on prior knowledge. For instance in robotic applications, categories can be linked to manipulation actions allowing for performing predefined actions on novel objects (see e.g., [1]). Categorizing novel objects is also useful in other applications such as driver assistance [2] and video surveillance [3]. The prior knowledge is built from previous observations by identifying common structures in the visual data. In this paper, we introduce an object categorization method based on unsupervised clustering of 3D relational features. Clustering [4] is a powerful tool to automatically find structures in the data that can be, in this context, translated into categories.

In our system, visual data are provided in terms of view-point invariant representations of objects extracted from 3D sensors. These representations code the properties of objects by computing global, pair-wise relations from 3D features (i.e., 3D texlets [5]). This space of feature relations is then expressed in histograms, providing unique and specific object descriptors. Moreover, such descriptors provide a fixed-length feature space, which can be directly fed into the clustering algorithm. The representations used here have been found to achieve high performance on object instance recognition [6].

In this paper, we apply hierarchical agglomerative clustering [7]. In contrast to flat clustering algorithms such as k-means, hierarchical clustering allows for overlapping of categories (see Fig. 1a for an illustration). This means that very similar categories are nested within larger clusters forming a structure in which more generic categories are found on top of less generic ones. This provides flexibility in selecting the abstraction level.

R. Paredes et al. (Eds.): IbPRIA 2015, LNCS 9117, pp. 541–551, 2015.
DOI: 10.1007/978-3-319-19390-8_61



(a) Illustrative hierarchical clustering

(b) Examples of labeled objects

Fig. 1. Examples of hierarchical clustering and object labeling. The thumbnails are resized for better visualization and they don't necessarily reflect their actual relative sizes.

Existing object categorization methods assume that objects belong to mutually-exclusive (single) categories [8]. In this paper, we consider scenarios where objects can have multiple, nested categories (see Fig. 1b). Such scenarios are very common when dealing with everyday objects. One important aspect of this approach is that it is inherently capable of providing multiple categories. In contrary, other approaches will not only require learning multiple classifiers (see e.g., [8]) but also—as we show in this paper—perform poorly on nested categories.

To evaluate the system, we hand-labeled our benchmark object set with a number of visual categories. We use the labeled categories from the training subset to find the best matching ones in the hierarchy. Then, we evaluate the performance of the system on the test subset. Note that although the hierarchy is built unsupervised, finding the corresponding categories is done in a supervised way. This is, however, necessary for evaluation. This procedure is repeated using different parameterizations of the visual representations in order to empirically find the best set of parameters for each category.

We compare this approach to a supervised approach using Random Forests [9] using the same visual features. In addition, we make a comparison with a state-of-the-art method (Hierarchical Matching Pursuit, HMP [8]) that works on RGB-D data and extracts distinct visual features. The main achievements of this work can be summarized as follows:

- We introduce an object categorization method that is capable of predicting multiple, nested categories.
- We show that—using hierarchical clustering for finding categories—we perform better than classification with Random Forests. Our method also outperforms a state-of-the-art method on our dataset.

– We demonstrate that the use of our visual features, compared to features extracted a state-of-the-art method, allow the system to have high performance with fewer training samples.

2 Related Work

Early research on object categorization focused on generic object representations that capture shape at high levels of abstraction (such as generalized cylinders [10], superquadrics [11], or geons [12]). The difficulty involved in reconstructing such abstractions from real objects has led to the development of solutions that could recognize only exemplar objects [13] (i.e., object recognition), which require little or no abstraction. Over the years, the gap between the low-level and the high-level abstractions has been narrowed by introducing representations that are invariant to a number of geometrical properties such as view-point, rotation, and scaling. Such representations often make use of local descriptors such the popular SIFT [14] features and various recently developed 3D features [15].

Belongie et al. [16] proposed representing objects using 'shape contexts', which uses relative shape information within a local neighborhood. The shape contexts were later extended to 3D in [17]. In this paper, we use shape relations of 3D features presented in [6], which are similar to shape context but are defined in a global context. Additionally, we go beyond the work in [6] by addressing the scale-invariance to obtain a more abstract representation that is important for object categorization.

Recently, hierarchical approaches for object representation have shown high performance on large dataset [18]. Notably, Bo et al. [8] introduced a multi-layer network that builds feature hierarchies layer by layer with an increasing receptive field size to capture abstract representations. They shows that their method achieves state-of-the-art performance in a large-scale RGB-D dataset of objects [19]. It is worth noting that these results are based on very large training data with significant computational cost.

Existing object categorization systems typically apply supervised learning to recognize object classes that correspond to labeled categories and associate only one category per object (single-label) [1,8]. In our approach, to summarizing the novelty, categories are learned in an unsupervised way using hierarchical agglomerative clustering [7]. Based on that, we built a method capable of associating more than one category per object (multi-label). Building such a hierarchy can be seen as a way to obtain higher levels of abstraction (from the visual features) where more generic categories are formed at the top of the hierarchy.

3 System Description

The components of the object categorization system introduced in this paper are shown Fig. 2. The system operates in a set-up in which three views are captured by three Kinect sensors, which are mounted in a close to equilateral triangular configuration. The process starts with scene preprocessing for table removal and

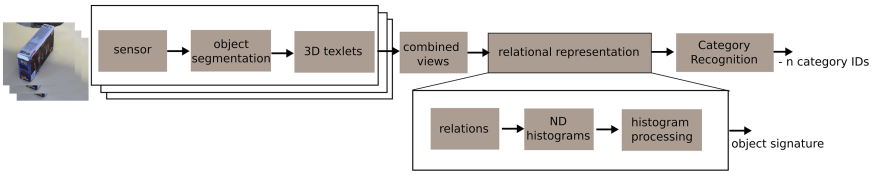


Fig. 2. System Overview: block diagram of the different components.

object segmentation in the 3D point cloud data. In following, we describe in detail the other components.

3.1 Object Representation Using Histogram of Relational Features

For the approach we introduce in this paper, object shapes are described as distributions of *relations* between pairs of 3D features. The relations we use are intrinsically pose-invariant.

From RGB-D data (Kinect sensor), we extract our 3D features—*3D texlets* [5]. The 3D texlet has both position and orientation, and provides absolute informations (relative to an external reference frame) of objects in the 3D space. In our system, we combine 3D texlets from three view resulting in a rather complete object information (see Fig. 3a). To describe an object, we compute a set of pair-wise relations from all pairs of texlets belonging to the object.

Shape relations are similar to the 3D shape context introduced as local descriptors by [17], however, they are used here as global descriptors of objects. Having combined multiple 3D views of objects allows such global descriptors to become robust and rich representations for fast learning.

In [6], we defined in detail three shape relations used for object instance recognition, namely, Angle Relation $R_a(\Pi_i^T, \Pi_j^T)$, Distance (Euclidean) Relation $R_d(\Pi_i^T, \Pi_j^T)$, and Normal Distance Relation $R_{nd}(\Pi_i^T, \Pi_j^T)$—they are also depicted in Fig. 3b. Note that the relations transform an absolute pose-dependent representation into a relative pose-independent one. For instance, the distance relation $\mathcal{R}_d$ transforms texlets' positions into inter-texlet distances.

The two distance relations are scale-variant, which is suitable for object instance recognition where object size matters and shall be encoded. However, for object categorization, because what defines a category is usually independent of scale, scale-invariance is crucial. Therefore, we introduce a new scale-invariant distance relation referred to as *Scaled Distance Relation*, $R_{sd}(\Pi_i^T, \Pi_j^T)$. The scaled distance is computed by dividing the Distance Relation by the maximum distance within an object. For robustness against outliers, the maximum distance is calculated as the median value of the highest 10 % distance relation values. Figure 3c shows an example of two objects with different sizes (belong to the same category), comparing using distance and scaled distance when representing objects. One aspect we investigate in the experiment section is the performance of the system on each category when combinations of different relations are used.

The final object representation is obtained by binning the selected relations in *multi-dimensional histograms*, which model the distributions of the relations

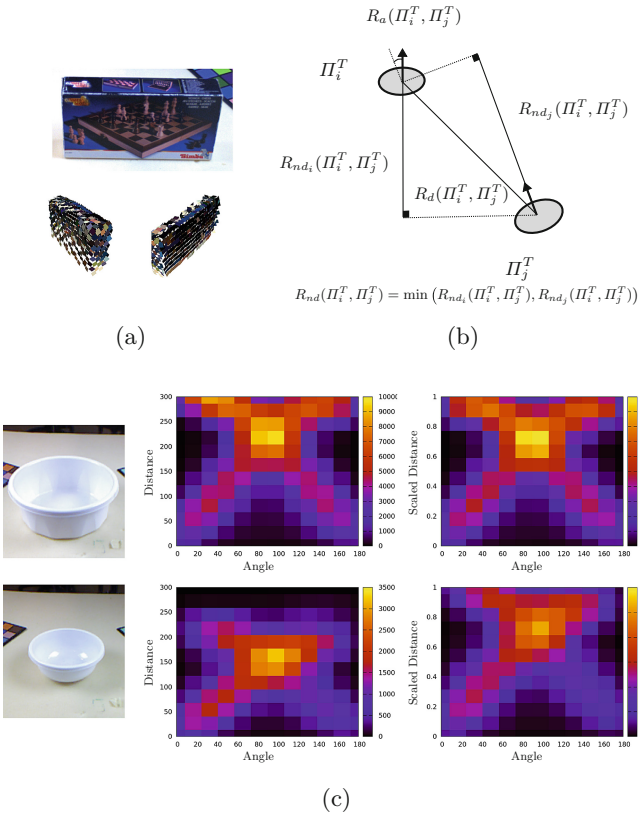


Fig. 3. Texlet's shape relations. (a) extracted 3D texlets of an object. (b) definition of three shape relations. (c) 2D histograms of two objects belonging to the same categorize. In the right column the distance is scaled

in fixed-sized feature vectors fed to the learning algorithm. Examples of 2D histograms are shown in Fig. 3c. Different values of binning size are also investigated in the experiments. Another process that is optionally performed on our histograms is *smoothing* using ND Gaussian filters to reduce the noise.

3.2 Finding Categories Using Hierarchical Clustering

Hierarchical Clustering. The quality and the invariance properties of the object representation presented in the previous section make it attractive for object categorization. We propose using unsupervised category learning through clustering using agglomerative hierarchical clustering [7] (R implementation [20]). By doing so, we build a hierarchy of clusters from unlabeled data where each cluster (branching point in the hierarchy) is considered as a potential category that can be linked (by an autonomous process) to an actual category. Note that we use the Euclidean distance as a dissimilarity measure (between all pairs of data samples or object instances in our case) whereas as a linkage metric, we use Ward's criterion, which aims at minimizing the total within-cluster variance [7].

Finding Categories From Human-Labeled Categories. Particularly for this paper, to validate our approach, we use human-defined labels from the training samples (Fig. 1b shows some examples) and then we find the corresponding categories in the hierarchy. Those definitions of categories are rather subjective and might not correspond to real ones. Therefore, we also compare our approach with other approaches—one of which even extracts different features from the raw RGB-D data.

To find a category in the hierarchy that correspond to a labeled one, we search for the cluster that contains the most similar set of object instances to the set of objects labeled as such. To compute the similarity, we use Jaccard's index [21], which measures the similarity between finite sample sets and is defined as the size of the intersection divided by the size of the union. The Jaccard's index rewards the existence of the object in the prospective cluster and also punishes for the absence thereof. This prohibits assigning categories to very specific (at the bottom of the hierarchy) or very generic clusters (at the top of the hierarchy).

Note that, although building the structure is done in an unsupervised way, finding the learned categories corresponding best to the labeled categories is performed in a supervised fashion (based on labeled data).

Predicting Categories for Novel Objects. To allow the system to categorize objects, the proposed method should provide a prediction mechanism. Traditionally for supervised learning, the learned model is used to make predictions for the novel object. Such a model usually forms a map of the feature space allowing for making predictions based on the features of the novel object. In our method, the principle concept we propose for prediction is to identify where the novel object falls in the learned (previously-built) hierarchy. This requires involving the training samples because the hierarchy is built directly from the training data. This seems computationally inefficient especially for large set of training samples. However, we show in this paper that our method requires few training samples.

To implement this, in the prediction phase, we first add the novel object to the objects previously used for training. Once the hierarchy is built again, we identify the closest sibling of the novel object. The novel object will then inherit all the branching points—including the ones associated with the labeled categories—from the sibling object. Finally, the predicted categories for the novel object will be all the inherited categories.

4 Dataset and Experiments

Dataset. To benchmark our approach, we use a dataset of 100 objects with 30 different samples (random poses) for each object[1]. The dataset was originally created to test the performance of the object instance recognition present in [6]. The selection of objects covers a wide range including industrial and household objects, some of them taken from the KIT dataset [22].

[1] http://caro.sdu.dk/index.php/sdu-dataset.

To validate our approach, we hand-labeled the objects in the dataset with purely visual as well action-related categories (see Fig. 1b). Note that a single object can have multiple (nested) categories. This allows us to study the performances of the different approaches on such cases.

Comparison Methods. The approach we introduce in this paper is compared with two different methods that use classical supervised learning. We apply those methods in N-classifier mode where N refers to the number of categories. This allows the methods to provide multiple categories per object and hence make them comparable with our approach. The first method we compare with is *a Random Forest classifier*, which uses the same features as the introduced approach (see Sect. 3).

The second method is HMP [8], which is a state-of-the-art method. HMP is a multi-layer sparse coding network that builds feature hierarchies layer by layer with an increasing receptive field size to capture abstract representations from raw RGB-D data. Note that HMP was not designed to combine features from different views in the 3D space. Therefore, to make it comparable to our multi-view system, we provide all the three views in the training phase. Additionally, we compare all methods when only one view is used.

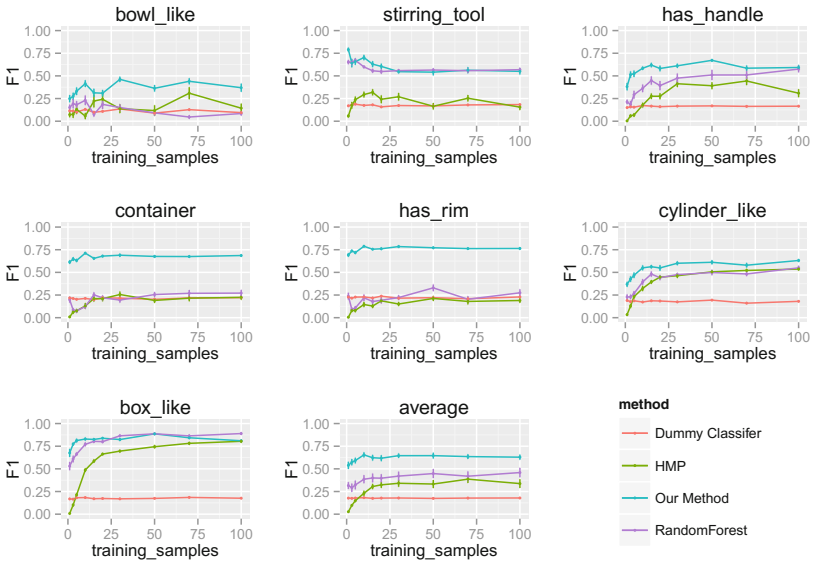
The comparison also includes a 'dummy' classifier that generates uniformly-distributed random category predictions. Comparing the different methods with this classifier may indicate whether a particular method has failed to achieve reasonably good performance on the category in question.

Histogram Variations. In the experiments below, we vary the parametrization of our object representation. The objective is to find out the set of parameters that yields the best performance on each category. Note that those variations are applied to the methods in which our object representation is used (namely, the proposed approach and the Random Forest approach). The object representation was discussed in detail in Sect. 3. The exact list of parameters we vary are the following:

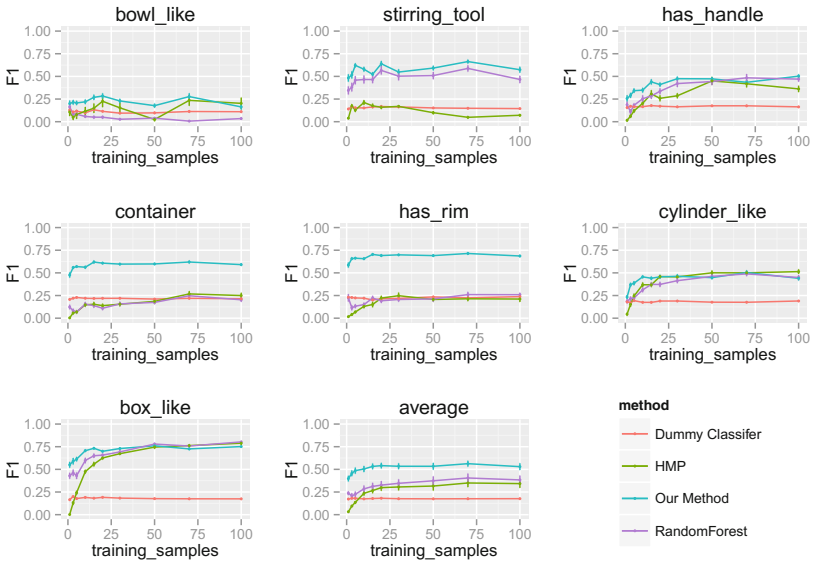
- Set of relations: We vary what relations to use for representing objects from the ones defined in Sect. 3: Angle Relation, Distance Relation, Normal Distance Relation, and Scaled Distance Relation.
- Relational dimensionality: We vary how we combine relations in ND histograms. The combinations we apply are: 1D histograms of the individual relations and 2D histograms of Angle Relation with one of distance relations.
- Histogram binning: Here, we vary the ND histogram bin size among the following values: 10, 20, 50 and 100. For simplicity, the bin size is fixed across dimensions in the 2D case.
- Filtering: We analyze the impact of applying filtering with Gaussian kernel.

In addition to the above-mentioned parameters, we also experiment with the impact of performing vector normalization on the final object representation.

Experimental Procedure. In the following experiments, we study the performance of each method for categorizing novel objects. Therefore, in each experiment, the object dataset is divided into training and test subsets where sampling



(a) three camera views



(b) one camera view

Fig. 4. The performance of object categorization on 7 categories. In (a) three camera views are used whereas in (b) one camera view is used. Using a 2D histogram of angle and scaled distance (with 10 bins at each dimension) yields the best performance in all categories except for 'bowl_like' (in this case, combining two 1D histograms of angle and scaled distance with 12 bins). Also, applying filtering and vector normalization helps achieving the best performance in all cases.

is performed in a way that prohibits the presence of samples from the same object in both subsets. Allowing otherwise, results in performing recognition of object instances rather than object categories in which in our tests we obtained significantly higher performance. The size of the test subset is set to 100 samples per category whereas the size of the training subset is allowed to vary—all samples are randomly chosen. Each experiment is executed 20 times from which the average F1 score and the standard deviation are computed. Note that the same training and test subsets are passed to each method.

In the result shown in Fig. 4b, the size of the training set varies among certain values: 1, 3, 5, 10, 15, 20, 30, 50, 70 and 100 per category. By doing this, we are able to study the performance of each method when only a small number of training samples are available and how that changes when the number increases.

Results. Figure 4 shows the performance of object categorization on 7 categories. Each sub-figure shows the average F1 score and the standard deviation for a varying number of training samples. The average performance on all categories is also shown in a separate sub-figure. The results show that the method introduced in this paper generally achieves the highest performance in identifying the categories particularly when a few training samples are used. This means that our method is able to learn faster and also generalize better when the training samples are limited.

Conclusion. Both the proposed method and the Random Forest approach use the same extracted visual representations of objects. This indicates that finding categories in clusters formed hierarchically in unsupervised way has a better generalization than the supervised learning of categories. Additionally, because both approaches outperform the HMP method, which extracts different visual representations, the results suggest that our visual representation provides strong features for describing object categories. This is particularly clear in the three-view case where our representation allows for combing the three views in 3D resulting in a rather complete object description.

For some categories (namely, 'container' and 'has_rim'), our method achieves relatively good performance in identifying the categories whereas the other methods fail (perform comparatively the same as the dummy classifier). Those categories are nested categories (i.e., in this case, any container also has a rim). This indicates that our approach is able to identify the relation between the two categories. The supervised approaches, on the other hand, try to learn discriminatively the two categories, which for most samples have very similar representations. This may explain their failure in identifying the nested categories.

Acknowledgment. This work has been funded by the EU project Xperience (FP7-ICT-270273).

References

1. Marton, Z.C., Pangercic, D., Rusu, R.B., Holzbach, A., Beetz, M.: Hierarchical object geometric categorization and appearance classification for mobile manipulation. In: 2010 10th IEEE-RAS International Conference on Humanoid Robots (Humanoids), pp. 365–370. IEEE (2010)
2. Laika, A., Stechele, W.: A review of different object recognition methods for the application in driver assistance systems. In: Eighth International Workshop on Image Analysis for Multimedia Interactive Services, p. 10. IEEE (2007)
3. Graham, S., Wood, D.: Digitizing surveillance: categorization, space, inequality. Crit. Soc. Policy **23**(2), 227–248 (2003)
4. Xu, R., Wunsch, D., et al.: Survey of clustering algorithms. IEEE Trans. Neural Netw. **16**(3), 645–678 (2005)
5. Olesen, S.M., Lyder, S., Kraft, D., Krüger, N., Jessen, J.B.: Real-time extraction of surface patches with associated uncertainties by means of kinect cameras. J. Real-Time Image Process. **10**(1), 105–118 (2015)
6. Mustafa, W., Pugeault, N., Buch, A., Krüger, N.: Multi-view object instance recognition in an industrial context. Robotica (accepted)
7. Ward, J.H.: Hierarchical grouping to optimize an objective function. J. Am. Stat. Assoc. **58**(301), 236–244 (1963)
8. Bo, Liefeng, Ren, Xiaofeng, Fox, Dieter: Unsupervised feature learning for RGB-D based object recognition. In: Desai, Jaydev P., Dudek, Gregory, Khatib, Oussama, Kumar, Vijay (eds.) Experimental Robotics. STAR, vol. 88, pp. 387–402. Springer, Heidelberg (2013)
9. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
10. Binford, T.O.: Visual perception by computer. In: IEEE Conference on Systems and Control, vol. 261, p. 262 (1971)
11. Pentland, A.P.: Perceptual organization and the representation of natural form. Artif. Intell. **28**(3), 293–331 (1986)
12. Biederman, I.: Recognition by components: a theory of human image understanding. Psychol. Rev. **94**(2), 115–147 (1987)
13. Campbell, R., Flynn, P.: A survey of free-form object representation and recognition techniques. Comput. Vis. Image Underst. **81**(2), 166–210 (2001)
14. Lowe, D.: Object recognition from local scale-invariant features. In: The Proceedings of the Seventh IEEE International Conference on Computer Vision, vol. 2, pp. 1150–1157 (1999)
15. Alexandre, L.A.: 3d descriptors for object and category recognition: a comparative evaluation. In: Workshop on Color-Depth Camera Fusion in Robotics, IROS (2012)
16. Belongie, S., Malik, J., Puzicha, J.: Shape matching and object recognition using shape contexts. IEEE Trans. Pattern Anal. Mach. Intell. **24**(4), 509–522 (2002)
17. Frome, A., Huber, D., Kolluri, R., Bülow, T., Malik, J.: Recognizing objects in range data using regional point descriptors. In: Pajdla, T., Matas, J.G. (eds.) ECCV 2004. LNCS, vol. 3023, pp. 224–237. Springer, Heidelberg (2004)
18. Bengio, Y., Courville, A.C., Vincent, P.: Unsupervised feature learning and deep learning: a review and new perspectives. CoRR, abs/1206.5538 1 (2012)
19. Lai, K., Bo, L., Ren, X., Fox, D.: A large-scale hierarchical multi-view RGB-D object dataset. In: IEEE International Conference on Robotics and Automation (2011)
20. R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2013). ISBN: 3-900051-07

21. Levandowsky, M., Winter, D.: Distance between sets. Nature **234**(5323), 34–35 (1971)
22. Kasper, A., Xue, Z., Dillmann, R.: The kit object models database: an object model database for object recognition, localization and manipulation in service robotics. Int. J. Robot. Res. (IJRR) **31**(8), 927–934 (2012)