Graph Kernels for Object Category Prediction in Task-Dependent Robot Grasping

Marion Neumann
KD, Fraunhofer IAIS
Sankt Augustin, Germany
marion.neumann@iais.fraunhofer.de

Plinio Moreno
Instituto Superior Tecnico
Lisbon, Portugal
plinio@isr.ist.utl.pt

Laura Antanas
CS, KULeuven
Leuven, Belgium
laura.antanas@cs.kuleuven.be

Roman Garnett
CS, University of Bonn
Bonn, Germany
rgarnett@uni-bonn.de

Kristian Kersting
KD, Fraunhofer IAIS
Sankt Augustin, Germany
kristian.kersting@iais.fraunhofer.de

ABSTRACT

Robot grasping is a critical and difficult problem in robotics. The problem of simply finding a stable grasp is difficult enough, but to perform a useful grasp, we must also consider other aspects of the task: the object, its properties, and any task-related constraints. The choice of grasping region is highly dependent on the category of object, and the automated prediction of object category is the problem we focus on here. In this paper, we consider manifold information and semantic object parts in a graph kernel to predict categories of a large variety of household objects such as cups, pots, pans, bottles, and various tools. The similarity based category prediction is achieved by employing propagation kernels, a recently introduced graph kernel for partially labeled graphs, on graph representations of 3D point clouds of objects. Our work highlights the importance of moving towards the use of structured machine learning approaches in order to achieve the dream of autonomous and intelligent robot grasping: learning to map low-level visual features to good grasping points under consideration of object–task affordances and high-level world knowledge. We evaluate propagation kernels for object category prediction on a (synthetic) dataset of 41 objects with 11 categories and a dataset of 126 point clouds derived from laser range data with part labels estimated by a part detector. Further, we point out the benefit of leveraging kernel-based object category distributions for task-dependent robot grasping.

1. INTRODUCTION

Graph data is abundant nowadays. Examples include social networks, the world-wide-web, biological networks, communication and transportation networks, and energy grids. Kernel-based learning techniques leveraging kernels between graphs (graph kernels) or kernels between nodes (kernel on graphs) have widely been studied [22, 10, 21]. Successful real-world applications, however, can almost exclusively be found in the field of bioinformatics, e.g. [5, 17]. An application area that has seen less influence from graph-based learning is robotics, in particular vision-based grasping. Objects can be grasped in different ways. For manipulation tasks in arbitrary and dynamic environments, it is essential to perform a *good* grasp. That is, the grasp depends on the specific manipulation scenario: the task, the object, its properties, as well as grasp constraints (e.g., gripper configuration). Thus, autonomous and intelligent robot grasping heavily relies on reasoning about world knowledge, in the form of object ontologies and object–task affordances, visual features, object categorical and task-based information.

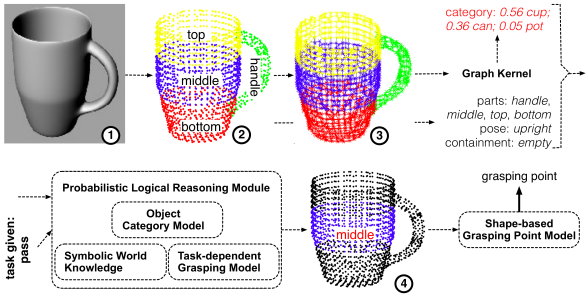


Figure 1: Task-dependent grasping pipeline for a cup. Top row: ① object; ② pose and symbolic parts (with labels *top* (yellow), *middle* (blue), *bottom* (red), and *handle* (green)); ③ k-nn graph[2] with part labels; graph kernel-based category distribution. Bottom row: probabilistic logical module; ④ predicted pre-grasp (*middle*) and grasping point module.

Consequently, it is difficult – if not impossible – to learn an unstructured model that directly maps visual perceptions to task-dependent grasps, as for instance introduced in [4, 14, 11, 19]. This may be the case especially if the robot acts in highly dynamical real-world environments handling a huge range of objects having different categories, functionalities and task affordances, such as objects found in households, supermarkets, or industrial sites. Instead, we investigate an intermediary probabilistic logical module to semantically reason about the most likely object part to be grasped, given the scene description, object category, and task constraints. Then, a mapping is learned from part-related local visual features to good grasping points. We propose a probabilistic logical pipeline, illustrated in Figure 1, exploiting graph kernels, object/task ontologies, semantic reasoning, and also perceptual low-level learning. By leveraging world knowl-

[2]The edges are colored according to the colors of the adjacent nodes.

edge and relations to encode compact grasping models, our pipeline can generalize over similar object and task categories, thus offering a natural way to encode the non-trivial realization of high-level knowledge. This allows us to experiment with a wide range of object categories and is therefore a critical aspect of autonomous agents acting deliberately in new environments. In this context, the exploitation of object ontologies, task affordances, and grasping constraints, and thus, successful generalization, heavily depends on a good object category estimation. For example, a *cup* cannot be stored upside-down in a cupboard if it contains liquid (is full), whereas for a full *can* this should be possible. As an example for task-dependent grasping, a *knife* should be grasped by its handle if its intended use is for cutting, whereas it has to be grasped at the blade if the task is passing it to a human. For the same task a *screwdriver*, however, could be grasped in any region as there is no danger for the interacting human to cut him/herself. That is why the main focus of this paper is on object category prediction, cf. the top row of Figure 1. The main contribution here is the use of propagation kernels on 3D point clouds for object categorization in robot grasping.

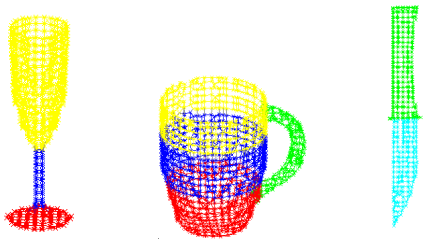
Object categories are predicted by employing object similarity based on manifold and semantic part information via propagation kernels, a recently introduced family of graph kernels capable to handle partially labeled graphs [16]. Given a (partial) 3D point cloud, the surface normals, and part labels of each point, we construct a k-nearest neighbour (k-nn) graph and compute an object similarity based on the propagation kernel – more specifically the diffusion graph kernel – among the respective labeled k-nn graphs. Object category prediction is achieved by applying a weighted vote on the categories of the most similar objects. Figure 2 shows k-nn graphs for three different objects. To enhance robustness, we compute a distribution over categories rather than hard predictions. Knowing the object category allows the robot to reason about object–task affordances and to generalize over object parts in similar task-dependent grasping situations. Generalization is achieved by utilizing an object ontology as illustrated in Figure 3.

We evaluate propagation kernels for object category prediction on two scenarios comprised of 41 and 126 3D point clouds and point out the benefit of leveraging kernel-based object category distributions for task-dependent robot grasping. We proceed as follows. We start off by reviewing related work on graph kernels and object categorization. Next, we give a detailed explanation of the graph construction and the kernel-based object category prediction. Before concluding, we present our experimental results.

2. RELATED WORK

Object categorization is an important problem in computer vision and robotics with numerous and versatile techniques. An extensive overview of related work is therefore beyond the scope of this paper. In the following, we review approaches based on graph kernels as well as some recent ones for 3D point cloud data.

Considerably previous work addresses the problem of object categorization via graph kernels for 2D images [9, 13, 20, 1, 7, 2, 23]. Although the approach is quite successful for 2D



(a) *glass_champagne* (b) *mug_ikea* (c) *knife_butcher*

Figure 2: Semantic k-nn graphs ($k = 4$) for three objects, (a) champagne glass, (b) coffee cup, and (c) butcher knife, with part labels *top* (yellow), *middle* (blue), *bottom* (red), *usable_area* (cyan), and *handle* (green). (The edges are colored according to the colors of the adjacent nodes.) Note, that the figures do not depict the edge weights being proportional to the change in curvature of the adjacent nodes.

data, it is not clear how to generalize them to handle general 3D point clouds, where exploiting the manifold structure of the data is more challenging. Thus, we introduce a new graph representation of 3D point clouds and a graph kernel that is able to deal with noise and missing information, commonly encountered in point clouds derived from laser range sensors.

Closely related is the work in [3], which introduces a subtree-pattern kernel for point clouds. This graph kernel is a special instance of the Weisfeiler–Lehman subtree kernel [21] which has proven to be successful on various bioinformatics benchmark datasets. Employing the Weisfeiler–Lehman test of isomorphism to compare subtree-patterns is highly efficient and hence, bounding the number of child nodes in the subtrees as suggested in [3] is not necessary for the sake of efficiency. Further, experimental evaluation in [3] is only carried out for 2D character recognition and not for more challenging problems, such as general object categorization. Therefore, again, it is not clear how the kernel performs on real 3D point cloud data. Our approach to object categorization for vision-based robot grasping is robust with respect to noisy input data and, uniquely, relies on semantic labels derived by a part detector. The point clouds may be incomplete and part label information may be only partially available. This is the main reason for the use of propagation kernels [16] which are explicitly designed for partially labeled graphs.

Several other closely related lines of related work tackle the problem of 3D object recognition [8, 12, 18]. However, they do not exploit manifold information in a graph kernel. Thus, to the best of our knowledge we present the first approach utilizing manifold-based graph kernels for 3D point cloud categorization and real-world robotic grasping.

3. OBJECT CATEGORY PREDICTION VIA GRAPH KERNELS

As the main contribution of this paper, we estimate the object category by retrieving the objects with the most similar global properties based on a graph kernel leveraging mani-

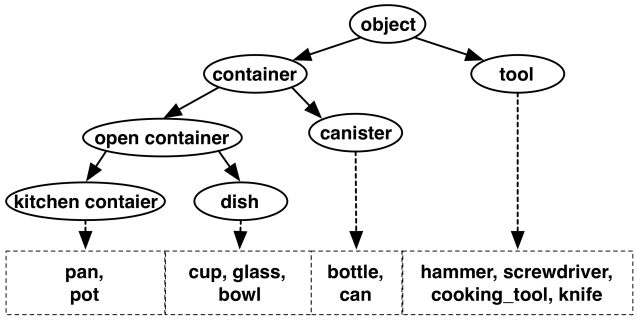


Figure 3: Object Ontology for the 11 categories considered in this paper.

fold information and semantic object parts. While beneficial for grasping point prediction [15], global object similarity also ensures a strong enough appearance-based predictor for object category. This prediction is then in the form of a distribution on object categories is used as prior for the probabilistic logical reasoning for task-dependent grasping. To incorporate global object similarity for object category prediction, we leverage propagation kernels, a recently introduced graph kernel designed for classification and retrieval of partially labeled graphs [16].

To get the distribution on object categories for a particular query object, we first represent its 3D point cloud as a graph. We then compute a graph kernel between this graph and graphs built from objects in an object database, and retrieve the objects deemed most similar. We represent each object by a labeled graph where the labels are the semantic parts and the graph structure is given by a k-nearest neighbour (k-nn) graph, cf. Figure 2. For each object point cloud we derive a weighted k-nn graph by connecting the k nearest points w.r.t. Euclidean distance in 3D. We use a four-neighbourhood (i.e., $k = 4$) and assign an edge weight reflecting the tangent plane orientations of its incident nodes to encode changes in the object surface. The weight of edge (i, j) between two nodes is given by $w_{i,j} = |\mathbf{n}_i \cdot \mathbf{n}_j|$, where $\mathbf{n}_i$ is the normal of point i. The nodes have five semantic classes encoding object part information: *top*, *middle*, *bottom*, *handle* and *usable_area*. To be able to capture manifold information as graph features in the presence of full label information, we use a diffusion scheme of the labels corresponding to the *diffusion graph kernel* proposed in [16], in the following, simply referred to as the *propagation kernel*. We stress that the graphs of the 3D point clouds as illustrated in Figures 1③ and 2. capture both manifold information (geodesic distance) via their structure and semantic information (part labels) via their node labels.

The similarity measure among objects is a kernel function over counts of similar node label distributions per diffusion iteration. The T-iteration propagation kernel between two graphs G' and G'' is defined as:

$$K_T(G', G'') = \sum_{t=0}^{T} \ker(G'_t, G''_t), \tag{1}$$

where T represents the maximum number of label propagation interactions considered and ker is a linear base kernel, defined as:

$$\ker(G'_t, G''_t) = \langle \phi(G'_t), \phi(G''_t) \rangle. \tag{2}$$

The main ingredients of propagation kernels are the distribution based graph features $\phi(G_t)$. They are essentially computed from node label distributions of running label diffusion on the respective graphs. Hence, the node label distributions of G_t are updated according to $L_t \leftarrow T\,L_{t-1}$, where the transition matrix T is the row-normalized adjacency matrix $T = D^{-1}A$ and D being the diagonal degree matrix with $D_{ii} = \sum_j A_{ij}$. Based on the $L_t (0 \leq t \leq T)$, we compute for each graph the counts of similar distributions among the respective graphs' nodes. As the node label distributions are $m-$dimensional continuous vectors, where m is number of semantic labels (in this case $m = 5$), propagation kernels use locality sensitive hashing (LSH) [6] as a quantization function to ensure the acquisition of meaningful features. We employ a quantization function for distributions approximately preserving the total variation distance (for more details, see [16]). The bin width parameter of LSH is fixed to $w = 10^{-4}$ in all experiments.

Propagation kernels leverage the power of evolving continuous node label distributions as graph features and hence capture both manifold and semantic information. Given a new object G^* that the robot aims to grasp, we first select the top n most similar graphs $\{G^{(1)}, \cdots, G^{(n)}\}$, where similarity is given by the respective row of the correlation matrix $\hat{K}$, where

$$\hat{K}_{ij} = \frac{K_{ij}}{\sqrt{K_{ii} K_{jj}}}. \tag{3}$$

Note, that by using $\hat{K}$ instead of K we achieve a normalization w.r.t. the number of points in the point clouds and hence, w.r.t. the scale of the objects. We set $n = 10$ in all our experiments. Second, we build a weighted average over the categories of the objects corresponding to $\{G^{(1)}, \cdots, G^{(n)}\}$, where the weight function is defined as

$$f(x) = \exp(-x), \tag{4}$$

with x being the rank after sorting the kernel row $\hat{K}_{*,:}$. This average is finally used as a prior distribution on the object category for a query object with graph representation G^* in the probabilistic logical module. The prior distribution over object categories for the cup in Figure 1 using $T = 10$ is:

`0.56:cup; 0.36:can; 0.05:pot; 0.02:pan; 0.002:bottle.`

Now, we will proceed by presenting experimental results object category prediction given 3D point cloud information and semantic part labels.

4. EXPERIMENTS

Our intention here is to investigate the power of graph kernels, namely the propagation kernel, for object category prediction and thus its applicability in a probabilistic logical approach to task-dependent robot grasping.

The two main questions we aim to answer in this section are:

(Q1) Does incorporating manifold information, i.e. graph structure, improve upon label-based category prediction?

(Q2) Are propagation kernels competitive with state-of-the-art graph kernels for the task of object category prediction for task-dependent robot grasping?

method	dataset/scenario	
	DB	SEMI/REAL
# of graphs	41	126
avg. # of nodes per graph	1 377	1 442
# node labels	5	5
# categories	11	11

Table 1: Statistics for all datasets. Note, that SEMI and REAL differ in the node labeling. In SEMI nodes are manually labeled, whereas in REAL semantic node labels are estimated by a part detector. DB is used in all experiments as database, so that graph kernels for the real scenarios (SEMI and REAL) are computed between the 41 database graphs and the respective query object.

method	dataset/scenario		
	DB	SEMI	REAL
RAND	9.1	9.1	9.1
K_{LABELS}	63.4	61.1	57.9
K_{WL} (best T)	73.2 (4)	56.3 (1)	47.6 (14)
K_{PK} (best T)	**87.8** (12)	**66.7** (6)	**58.7** (6)

Table 2: Average accuracies (%) on all objects in the datasets DB, SEMI, and REAL for graph kernels K_{PK} and K_{WL}, and baselines RAND and K_{LABELS}. We report results for K_{PK} and K_{WL} for the best T on the respective evaluation scenario, the maximum number of iterations T resulting in the best average accuracy is given in brackets. Note, that K_{LABELS} corresponds to a linear kernel among label counts ($T = 0$), i.e., no graph structure is incorporated in the kernel computation. Bold indicates best performance.

4.1 Datasets

The three datasets for quantitative evaluation are generated with the help of the ORCA simulator[3] and their statistics are summarized in Table 1.

DB. For the first dataset data acquisition is fully simulated. That is, we have the complete point cloud of the input object and the semantic part labels are provided as ground truth. The point cloud is obtained from a previously defined 3D mesh of the object by up-sampling points using midpoint surface subdivision. Object parts are extracted manually from the point cloud. This "perfect scenario" is then used twofold. First, we use this dataset to evaluate object category prediction in a synthetic scenario. The dataset denoted as DB contains 41 objects belonging to 11 categories modeled by the ontology illustrated in Figure 3. Second, we use these 41 objects $G^{\text{DB}} = \{G^{(1)}, ..., G^{(41)}\}$ as database to compute the propagation kernel $K_T(G^*, G^{\text{DB}}) = \sum_{t=0}^{T} \ker(G_t^*, G_t^{\text{DB}})$ to get the similarity among any query object G^* and the database objects. Hence, this DB is used as object database in all experiments. Further, we plan to use it in execution time on a robot performing task-dependent grasps.

SEMI and **REAL**. For the other two datasets the point clouds are simulated laser range data and the following steps of obtaining the scene description are executed in the ORCA simulator. The point cloud of the query object is estimated from several view points. The 3D data for each view is acquired from a simulated range camera (with the parameters of the Kinect sensor), placed on the robot platform. For each object we use between one and eight views depending on whether the object is symmetric or not. We further consider two possible settings. In a first setting, we obtain the labels manually from the realistic point cloud, that is we use ground truth label information. We denote this dataset as SEMI. In a second setting we also replace ground truth parts with more realistic part detector estimates.[4] We denote this dataset as REAL. The two more realistic datasets SEMI and REAL contain 126 labeled point clouds which are instances of 26 objects of all categories except *cooking_tool*. The point cloud estimation from a single view on the object is described in Section 3.

[3] `http://orca-robotics.sourceforge.net`

[4] Note, that the part detector estimates are corrected for various tool objects as the current version only detects the correct parts but not the correct location.

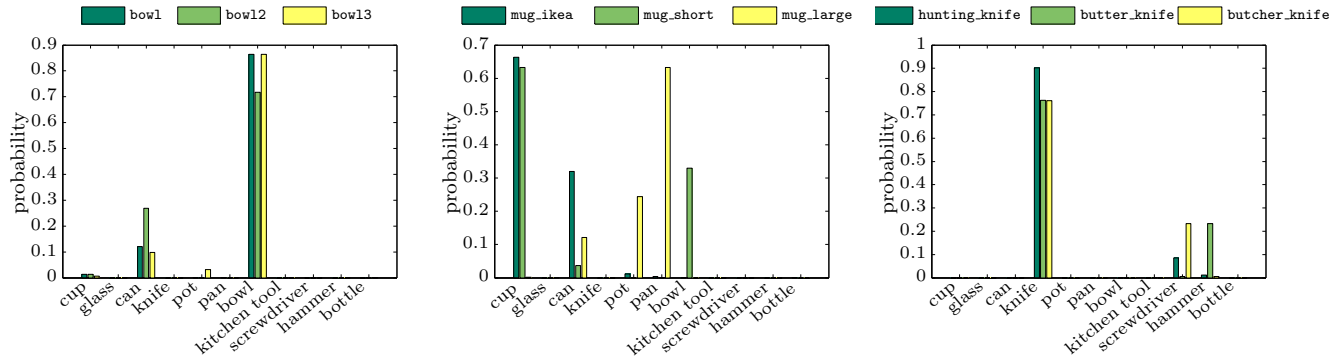
4.2 Experimental Protocol

We compare propagation kernels (K_{PK}) to the the Weisfeiler–Lehman subtree kernel (K_{WL}) and to two baselines: random category selection (RAND), and a linear kernel (K_{LABELS}) taking as input only the labels counts but no graph structure. For all kernel-based methods we select the 10 most similar objects from the database DB based on the respective row of the normalized kernel matrix, Eq. (3), and build a weighted average of their categories using Eq. (4). Note that predicting the object category utilizing a kernel machine directly (we tried support vector machine classification) did not give competitive results. This might be caused by the large number of classes in combination with few training examples. For missing part label information on the nodes we initialize the label distribution uniformly. For K_{WL} we use an additional label as suggested in [21]. For K_{PK} and K_{WL} we present results where the maximum number of iterations $T \in \{1, ..., 15\}$ is chosen to give the best performance. Note, that setting $T = 0$ in K_{PK} and K_{WL} gives the same results and corresponds to K_{LABELS}. For DB, performance is measured by a leave-one-out cross validation.

4.3 Quantitative Results on Object Category Prediction

The predictive performances for all methods on all scenarios are summarized in Tables 2 and 3. We report on average accuracy (Table 2) to evaluate the object category prediction task. Here, we take the category with the highest probability mass as prediction. On DB we clearly see that both graph kernels K_{PK} and K_{WL} improve upon using label information only. On SEMI only propagation kernels are able to improve the performance of the baseline methods. In the third scenario (REAL) average accuracies are only slightly higher for K_{PK}. In general, however, question **(Q1)** can be answered affirmatively. Note, that for K_{WL} setting $T = 0$ gives the best performance, then $K_{\text{WL}} = K_{\text{LABLES}}$.

Besides analyzing the most probable category, our aim here is also to evaluate the category distributions and their use in the probabilistic logical approach to task-dependent grasp-



(a) *bowl* (b) *cup* (c) *knife*

Figure 4: Category distributions for example objects from DB with classes *bowl* (a), *cup* (b), and *knife* (c) derived from K_{PK} with $T = 12$.

method	dataset/scenario		
	DB	SEMI	REAL
K_{LABELS}	1.88	1.78	2.09
K_{WL} (best T)	1.59 (5)	1.57 (14)	**1.75** (14)
K_{PK} (best T)	**1.39** (13)	**1.53** (4)	1.87 (4)

Table 3: Average rank of the true category (the lower the better) for all objects in the datasets DB, SEMI, and REAL for graph kernels K_{PK} and K_{WL}, and baseline K_{LABELS}. We report results for K_{PK} and K_{WL} for the best T (provided in brackets) on the respective evaluation scenario. Bold indicates best rank.

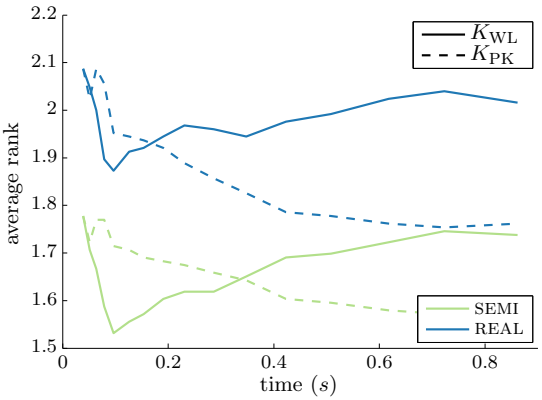


Figure 5: Time vs. average rank of the true category in the distribution computed from K_{PK} and K_{WL}. For both axes lower values are better.

ing. First, we illustrate example distributions for objects of three different categories *bowl, cup,* and *knife* in Figure 4. The category distributions for all bowls have its highest probability at the correct category (Panel (a)). Predicting the category of `mug_large`, however, fails (Panel (b)). This might be because the body of this particular cup is shaped more like one of the pans and some pots in the database. In general it is reasonable that the cups look more like pans and pots as they have a similar shape[5] and the same parts (*top*, *middle*, *bottom*, and *handle*). The distributions of the knives clearly show their similarity to other knives but also to other tool objects like hammers and screwdrivers. This is consistent with their semantic parts (*usable_area* and *handle*). The higher probability mass for the category *knife* is caused by their shape, i.e. the graph structure of the corresponding point clouds. This behaviour is exactly desired in the probabilistic logical grasping model as task affordances and object ontology can be used to achieve generalization.

In order to quantitatively compare the distributions we compute the average rank of the true category in the distributions obtained from all kernels K_{LABELS}, K_{WL}, and K_{PK}. The results are summarized in Table 3. Again, we see that leveraging the graph structure improves the average rank especially for the third scenario REAL. K_{PK} achieves a lower average rank on DB and SEMI. On REAL 14 iterations of K_{WL} result in the lowest average rank for the Weisfeiler-Lehman subtree kernel. However, K_{PK} results in a not much worse rank after 4 kernel iterations and hence in faster time. In robotics computation time is an important criteria. Obviously we do not want the robot pausing too long before actually grasping the desired object appropriately. Hence, evaluating applicability in robotics also involves analyzing the trade-off between qualitative performance and computation time. Figure 5 shows the average time in seconds needed to compute the graph kernels versus the respective average rank for the realistic grasping scenarios SEMI and REAL. Note, that we use a hashing-based implementation of the Weisfeiler-Lehman kernel as introduced in [16]. The plot clearly shows a superior behaviour for propagation kernels as a lower rank is reached in shorter time. These results answer question **(Q2)** affirmatively.

[5]Note, that we use the correlation matrix (Eq. (3)) which yields a normalization w. r. t. the size (number of nodes).

4.4 Benefits for Task-dependent Grasping

In this section we aim to investigate the importance of using the object category distributions derived from the propagation kernel in the probabilistic logical pipeline described in the Introduction. We compare the performance of incorporating kernel-based distributions versus uniform distribu-

tions on object categories for two inference problems. In the first problem, we want to infer the most suitable grasping task to be performed on a given query object. The second problem is to predict the most suitable pre-grasp, i.e. graspable region for a given grasping task. This setting is depicted in the bottom row of Figure 1 for the given task *pass*. A more detailed evaluation of the whole probabilistic logical pipeline goes beyond the scope of this paper. In summary, the results for both grasping tasks are significantly better when using the graph kernel distribution as opposed to the uniform one. For example we achieve an improvement of about 20% accuracy for the task selection problem. This highlights the importance of good category distributions for task-dependent robot grasping and encourages further research on graph kernels for object category prediction.

5. CONCLUSIONS

We have presented a novel application of graph kernels, specifically, object category prediction as a prerequisite for task-dependent robotic grasping. Given an object to grasp, we convert its 3D point cloud into a graph and use a part detector to predict part labels for each node. This graph representation of the object is then compared to graphs of known objects in a database, and the most similar ones are used to predict the category of the unknown object. A previously proposed graph kernel, the diffusion graph kernel, was chosen for this task, and we showed in a series of experiments that it is able to quickly and accurately predict an object's category. Finally, our graph-kernel based object category predictor has measurable impact on the overall quality of the task-dependent robotic grasping pipeline it is part of.

6. ACKNOWLEDGMENTS

This work was supported by the European Commission under contract "FP7-248258-First-MM", by the Fraunhofer ATTRACT fellowship STREAM, and by the German Science Foundation (DFG) under reference number "GA 1615/1-1".

7. REFERENCES

[1] L. Antanas, P. Frasconi, F. Costa, T. Tuytelaars, and L. D. Raedt. A Relational Kernel-Based Framework for Hierarchical Image Understanding. In *Structural, Syntactic, and Statistical Pattern Recognition - Joint IAPR International Workshop (SSPR/SPR-2012)*, pages 171–180, 2012.

[2] L. Antanas, P. Frasconi, T. Tuytelaars, and L. De Raedt. Employing Logical Languages for Image Understanding. In *IEEE Workshop on Kernels and Distances for Computer Vision (ICCV-2011)*, 2011.

[3] F. R. Bach. Graph Kernels between Point Clouds. In *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML-2008)*, pages 25–32, 2008.

[4] J. Bohg and D. Kragic. Learning Grasping Points with Shape Context. *Robotics and Autonomous Systems*, 58(4):362–377, 2010.

[5] K. M. Borgwardt, C. S. Ong, S. Schönauer, S. V. N. Vishwanathan, A. J. Smola, and H. P. Kriegel. Protein Function Prediction via Graph Kernels. In *Intelligent Systems for Molecular Biology (Supplement of Bioinformatics) (ISMB-2005)*, pages 47–56, 2005.

[6] M. Datar and P. Indyk. Locality-sensitive Hashing Scheme Based on p-stable Distributions. In *Proceedings of the 20th Annual Symposium on Computational Geometry (SCG-2004)*, pages 253–262, 2004.

[7] O. Duchenne, A. Joulin, and J. Ponce. A Graph-Matching Kernel for Object Categorization. In *IEEE International Conference on Computer Vision (ICCV-2011)*, pages 1792–1799, 2011.

[8] R. Farid and C. Sammut. Plane-based object categorization using relational learning. *Machine Learning*, pages 1–21, 2013.

[9] Z. Harchaoui and F. R. Bach. Image Classification with Segmentation Graph Kernels. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR-2007)*, pages 1–8, 2007.

[10] H. Kashima, K. Tsuda, and A. Inokuchi. Marginalized kernels between labeled graphs. In *Proc. of the 20th International Conference on Machine Learning (ICML-2003)*, pages 321–328, 2003.

[11] I. Lenz, H. Lee, and A. Saxena. Deep Learning for Detecting Robotic Grasps. *CoRR*, abs/1301.3592, 2013.

[12] M. Madry, C. H. Ek, R. Detry, K. Hang, and D. Kragic. Improving Generalization for 3D Object Categorization with Global Structure Histograms. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS-2012)*, pages 1379–1386, 2012.

[13] A. Mahboubi, L. Brun, and F. Dupé. Object Classification Based on Graph Kernels. In *Proceedings of the 2010 International Conference on High Performance Computing & Simulation (HPCS-2010)*, pages 385–389, 2010.

[14] L. Montesano and M. Lopes. Active Learning of Visual Descriptors for Grasping using Non-parametric Smoothed Beta Distributions. *RAS*, 60(3):452–462, 2012.

[15] M. Neumann, R. Garnett, P. Moreno, N. Patricia, and K. Kersting. Propagation Kernels for Partially Labeled Graphs. In *ICML–2012 Workshop on Mining and Learning with Graphs (MLG–2012)*, Edinburgh, UK, 2012.

[16] M. Neumann, N. Patricia, R. Garnett, and K. Kersting. Efficient Graph Kernels by Randomization. In *European Conference on Machine Learning and Knowledge Discovery in Databases (ECML/PKDD-12)*, pages 378–393, 2012.

[17] A. Passerini, M. Lippi, and P. Frasconi. Predicting metal-binding sites from protein sequence. *IEEE/ACM Trans. Comput. Biology Bioinform.*, 9(1):203–213, 2012.

[18] S. Savarese and F.-F. Li. 3D Generic Object Categorization, Localization and Pose Estimation. In *IEEE 11th International Conference on Computer Vision (ICCV-2007)*, pages 1–8, 2007.

[19] A. Saxena, L. L. S. Wong, and A. Y. Ng. Learning Grasp Strategies with Partial Shape Information. In *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence (AAAI-2008)*, pages 1491–1494, 2008.

[20] D. Semenovich and A. Sowmya. Geometry Aware Local Kernels for Object Recognition. In *10th Asian Conference on Computer Vision (ACCV-2010)*, pages 490–503, 2010.

[21] N. Shervashidze, P. Schweitzer, E. van Leeuwen, K. Mehlhorn, and K. Borgwardt. Weisfeiler–Lehman Graph Kernels. *Journal of Machine Learning Research*, 12:2539–2561, 2011.

[22] A. Smola and R. I. Kondor. Kernels and Regularization on Graphs. In *Computational Learning Theory and Kernel Machines, 16th Annual Conference on Computational Learning Theory and 7th Kernel Workshop (COLT/Kernel-03)*, pages 144–158, 2003.

[23] L. Zhang, M. Song, X. Liu, J. Bu, and C. Chen. Fast Multi-View Segment Graph Kernel for Object Classification. *Signal Processing*, 93(6):1597–1607, 2013.