Exemplar-based Prediction of Global Object Shape from Local Shape Similarity

Jeannette Bohg[1], Daniel Kappler[1] and Stefan Schaal[1,2]

***Abstract*— We propose a novel method that enables a robot to identify a graspable object part of an unknown object given only noisy and partial information that is obtained from an RGB-D camera. Our method combines the benefits of local with the advantages of global methods. It learns a classifier that takes a *local* shape representation as input and outputs the probability that a grasp applied at this location will be successful. Given a query data point that is classified in this way, we can retrieve all the locally similar training data points and use them to predict latent *global* object shape. This information may help to further prune positively labeled grasp hypotheses based on, e.g. relation to the predicted average global shape or suitability for a specific task. This prediction can also guide scene exploration to prune object shape hypotheses.**

To learn the function that maps local shape to grasp stability we use a Random Forest Classifier. We show that our method reaches the same classification performance as the current state-of-the-art on this dataset which uses a Convolutional Neural Network. Additionally, we exploit the natural ability of the Random Forest to cluster similar data. For a positively predicted query data point, we retrieve all the locally similar training data points that are associated with the same leaf nodes of the Random Forest. The main insight from this work is that local object shape that affords a grasp is also a good predictor of global object shape. We empirically support this claim with quantitative experiments. Additionally, we demonstrate the predictive capability of the method on some real data examples.

I. Introduction

Autonomous grasping of any kind of object in arbitrarily complex environments is still unattainable for today's robots. There have been great advances in robust grasping and even manipulation of *known* objects in environments of moderate complexity, e.g. by Righetti et al. [1], Hudson et al. [2], Kazemi et al. [3]. However, the higher the uncertainty about crucial aspects of a manipulation task, the harder it becomes for the robot to successfully plan and execute its actions.

For example, there are theoretically well-founded metrics to evaluate the performance of a grasp given complete information about the object, the hand and their relative poses [4] that are implemented in all the major simulators such as GraspIt! [5] or OpenRave [6]. But how to let a robot grasp an object of uncertain global shape is an active area of research. There is little agreement in the community on how to best represent partial object information and infer a grasp given this. This is however a very common problem in real-world scenarios. Especially in cluttered scenes, large parts of an object may be occluded and segmentation of the visible parts from its surroundings becomes more difficult. A comprehensive overview of the different approaches towards this problem is given by Bohg et al. [7].

There are a few methods that try to estimate global object shape from partial information. The inferred information provides the basis for grasp planning methods that assume knowledge of a full object model [8], for further guiding tactile exploration [9, 10] to learn object models or for informing a grasp controller [10].

Other methods do not attempt to predict global object shape but rather to predict graspability directly from the partial and local information. These methods often employ supervised learning techniques on annotated grasp experience databases to predict where and how to grasp an object in a scene [11, 12, 13, 14, 15, 16]. Local methods have several advantages over method representing global object shape. They allow to generalize learned models across different objects that may have a very different global shape but are locally similar. Because they only rely on local information, they are also less sensitive to segmentation errors or occlusions. Furthermore, no prior semantic knowledge on e.g. object identity or category is necessary. All these factors reduce the complexity of information extraction from raw sensory data. However, global object shape has a large influence on whether a grasp will succeed or not. This is naturally not captured by local information. If two objects share two similar parts but have otherwise vastly different global shapes, different grasps may be required. In this paper, we propose a method that predicts (i) graspability and (ii) global object shape given only local information. Thereby, we inherit the advantages of local methods and still yield a prediction of global object shape. This can form the input to subsequent grasp planners and controllers or guide further interactive exploration of the environment.

In detail, we aim to infer a grasp pre-shape for an object of unknown identity, category or shape given a point cloud obtained from an RGB-D camera. In line with some of the aforementioned related work [11, 13, 14, 15, 16], we formulate this as a classification problem that takes in a local shape representation and outputs the probability of a grasp applied at this location to be successful. We learn the function that maps our local shape feature to grasp stability based on a recently proposed large-scale synthetic database Kappler et al. [17]. In total, it contains around 500k data points of annotated local shape representation, referred to as *templates* throughout the remainder of this paper.

As a classifier, we use a Random Forest. In the experiment section, we will show that our trained model achieves the

[1] Autonomous Motion Department at the Max-Planck-Institute for Intelligent Systems, Tübingen, Germany Email: first.lastname@tue.mpg.de

[2] Computational Learning and Motor Control lab at the University of Southern California, Los Angeles, CA, USA

same performance as the current state-of-the-art on this data set which uses a Convolutional Neural Net. For our second aim of predicting global object shape from local information we exploit the ability of the Random Forest to cluster the dataset into locally similar templates. Given a classified query template, obtained from the trained Random Forest, we can extract all the exemplars of the training data set that ended up at the same leaf nodes. Since we have access to complete information about these exemplars, we can use it to make predictions about some latent properties of the target object. In this paper, we will focus on the global object shape as this constitutes important information for grasp planners and robot controllers or interactive exploration. We model the global shape of the target object as a non-parametric distribution that is populated with the polygonal mesh models of the retrieved training data points. In the experimental section, we quantitatively show that this yields coherent information about the ground truth shape of the query object although only using local shape information. Furthermore, we show qualitative examples of grasp retrieval and object shape prediction on real data.

II. Related Work

In this section, we will review some of the methods that use a supervised learning technique for deciding whether a grasp is stable or not. We place special focus on those that use local 3D shape information as they are closest to the approach proposed in this paper. Detry et al. [12], Lenz et al. [13], Herzog et al. [14] and Kroemer et al. [15] extract local shape information and either represent it relative to the hand coordinate system [12, 13] or relative to some object-related coordinate frame [14, 15]. All infer hand pre-shapes where Herzog et al. [14] also varies the hand configuration. Kroemer et al. [15] infers a full movement primitive. The extracted local 3D data is differently represented. We adopt a variation of the local shape representation by Herzog et al. [14] that has shown to work well for exemplar based approaches. In particular it holds additional information compared to the raw point clouds as it explicitly encodes occlusion and free space. However, any other local shape representation may also be used with the proposed approach. Lenz et al. [13], Herzog et al. [14] and Kroemer et al. [15] use supervised learning to infer a graspable sub part of the environment. From those, Herzog et al. [14], Kroemer et al. [15] use very little data and incrementally improve their model based on trial and error learning. Lenz et al. [13] learn a deep network on an annotated dataset to classify their local shape representation as either graspable or not. Recently, Redmon and Angelova [18] trained a deep network on the same data and outperformed [13]. However, the authors use global image information to predict a grasp. The resulting model should suffer in performance when objects are not as well segmented as in this particular dataset. We train our model on significantly more data from a new large-scale synthetic database in which each grasp is automatically annotated using physics simulation [17]. The suitability of this metric is verified through crowd-sourcing. It is also the largest dataset available in the community and probably has the least noisy labels of all the synthetic datasets. Furthermore, it provides the aforementioned local shape representation. We use a Random Forest that predicts stable grasps at similar accuracy as a Convolutional Neural Net (CNN) and has the additional ability to cluster locally similar data in a supervised manner. We will show that we can predict the global object shape based on the locally similar exemplars. This is similar to retrieval approaches where given a query object, we retrieve additional object information from the locally similar exemplars. Goldfeder et al. [19] use a similar database to ours. It is also constructed in simulation. However, it is based on the *Princeton Shape Benchmark*(PCB) [20] which contains many objects that are rather uncommon in a household such as trees, insects or planes. Furthermore, the labeling is based on a classic metric which we have shown to be an inferior predictor of grasp success compared to a physics-based metric [17]. The retrieval method by Goldfeder et al. [19] is based on feature matching extracted from synthetic depth maps of the objects as seen from the robotic hand. In our database, there exist multiple local shape templates per grasp that are generated by varying the independent viewpoint of the camera. This reflects sensing conditions on a real robot that seldom have a camera in the palm of the hand. Detry et al. [12] also propose a method for grasping by retrieval. The authors focus on finding a lower dimensional space in which the data samples can be clustered. The information in each cluster is then compressed and represented by a prototypical grasp that can be retrieved by projecting a query grasp into the lower-dimensional space. None of these retrieval methods has investigated whether global object shape can be predicted from local information only.

Summarizing, the contributions of the method proposed in this paper are: (i) a discriminative model for predicting whether local shape affords a stable grasp, (ii) applying the same model for retrieving global object properties. In this paper, we focus on global object shape.

III. Problem Formulation

Given the observation of a scene from an RGB-D camera, we want to infer the best grasp location. Specifically, our aim is to infer a pre-grasp pose (position and orientation of the hand) which forms a natural pre-cursor to reactive grasping approaches such as [21, 22, 23] that can robustly acquire a grasp under uncertainty. More formally, given an observation $\mathcal{O}$ of the scene, we want to learn a function f that outputs the best grasp pose $\mathbf{g}$:

$$\mathbf{g} = f(\mathcal{O}) \tag{1}$$

where $\mathbf{g} = (\mathbf{x}, \mathbf{q})$ with $\mathbf{x}$ as the 3D position and $\mathbf{q}$ as the rotation represented by a quaternion. We formulate this as a classification problem which determines whether grasping at a specific location in the environment will be successful or not. Let $\Phi(\mathbf{g}, \mathcal{O})$ refer to the local shape of the environment $\mathcal{O}$ relative to the coordinate frame of the grasp $\mathbf{g}$. To simplify notation, we will use $\Phi_{\mathbf{g}}$. Furthermore, we define the class

label $l(\Phi_{\mathbf{g}}) = 1$ if this particular local shape affords a stable grasp. We train a discriminative model that given $\Phi_{\mathbf{g}}$ predicts whether $\mathbf{g}$ will result in a stable grasp:

$$y = p(l(\Phi_{\mathbf{g}}) = 1|\mathbf{g}, \mathcal{O}) \tag{2}$$

with $y \in [0, 1]$. In particular, we are interested in the best pre-grasp pose from the set $\mathcal{G}$ of all possible grasps:

$$\mathbf{g}^* = f(\mathcal{O}) = \underset{\mathbf{g} \in \mathcal{G}}{\operatorname{argmax}}\, p(l(\Phi_{\mathbf{g}}) = 1|\mathbf{g}, \mathcal{O}) \tag{3}$$

Some of the related approaches, discussed in Section II, can be exactly cast in this formulation.

Additionally to classifying a grasp candidate as either stable or unstable, we also aim to predict additional latent information about the object that has generated this candidate. We formulate this as a retrieval problem in which the resulting set of most similar data points forms a non-parametric distribution over the associated variables that are latent for the query object. In our case, this distribution may be over the object's global shape, its category, contact points or final grasp pose. In this paper, we focus on the global shape. More formally, given a query grasp pose $\mathbf{g}_q$, we want to retrieve a subset $\mathcal{S}$ from the total set $\mathcal{G}$ of grasps whose members are perceptually similar to the query grasp pose.

$$\mathcal{S} = \{\mathbf{g}_d \in \mathcal{G} | d(\Phi_{\mathbf{g}_q}, \Phi_{\mathbf{g}_d}) < \tau\} \tag{4}$$

where d is a distance function between grasp poses and τ is a threshold on this distance. This is essentially a nearest neighbor problem. In terms of the prediction of target variables, the connection between Random Forests and adaptive k-nearest neighbor has previously been pointed out by [24]. We will show, how a Random Forest classifier can be used to learn the function f in Eq. 3 and how at the same time it defines the variable number of nearest neighbors in the set $\mathcal{S}$ as defined in Eq. 4.

IV. Approach

In this paper, we simultaneously address grasp prediction and retrieval of latent global object properties. We employ a random forest classifier as the discriminative model and use its natural ability to cluster similar data points at the leaf nodes for the retrieval task. This clustering is supervised in the sense that it is not only driven by the similarity of the data points themselves but also by their ground truth grasp stability label. Fig. 1 gives an overview of out full pipeline during both training and testing.

A. Grasp and Local Shape Representation

We use a variation of the local shape representation by Herzog et al. [14] as described in Kappler et al. [17]. There are several advantages of this local representation of the shape at which a grasp is applied. First of all, it is not as dependent on accurate object segmentation as a global representation. It can be extracted easily and efficiently from partial point clouds. An unoptimized version for template extraction requires on around 80 milliseconds for every template. Additionally, it explicitly represents occluded parts as being different from free space. In the following, we will refer to this local shape representation as *template*.

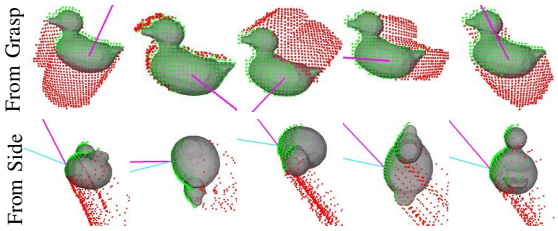


Fig. 2: Variation of the local shape representation given different viewpoints. The grasp for each of these templates is the same, i.e. approach direction along the cyan line and fixed wrist roll. The viewpoint is indicated by the pink line. Each column shows the same template from two different directions. (Top) Template viewed from the approach direction. (Bottom) Template viewed from the side. The occlusion area is the most affected by the varying viewpoint. Figure adopted from our previous work [17].

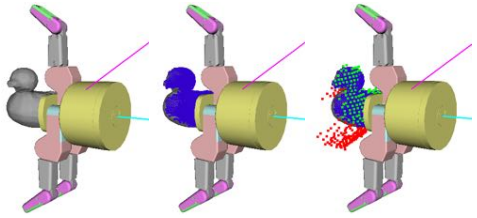


Fig. 3: Template for a small example object. It is linked to the grasp visualized in (a). Cyan line labels the surface normal and approach direction. Pink line labels the viewpoint from which the point cloud (blue dots in (b)) got generated. The associated template is visualized in (c). Green dots label surface points. Red dots label the boundary to the occluded space. Figure adopted from our previous work [17].

Given a partial point cloud and a normal for each point, a template is a circular height map of a specified diameter. Its origin is positioned at the intersection of hand approach and object model. The height map has the same normal vector as this point. The diameter is chosen according to the size of the robotic hand. A template has three channels each storing information about the surface, free or occluded space. The surface channel measures the distance from the height map plane to the point cloud. Given the viewpoint from which the point cloud was recorded, we can compute the occluded space when viewing the point cloud along the template normal. The occlusion channel measures the distance from the height map plane to this occluded space. The free space channel stores a fixed value for all positions on the template that are neither occluded nor occupied by a surface. See Fig. 2 for a visualization of such a template.

A grasp is linked to this template by equating its approach direction with the normal of the height map. Its pre-grasp position is at a fixed distance from the template origin to which we will refer as *stand-off* throughout the remainder of this paper. Each template has a fixed coordinate system. The roll of the hand is determined by aligning the axis along the fingers with the x-axis of the template. For the experiments in this paper, we only consider one finger configuration as depicted in Fig. 3.

B. Feature Vector

We discretize the height map into a grid of 48×48, for all 3 channels. The dimensionality of the template is very high when considering it as the input to the Random Forest

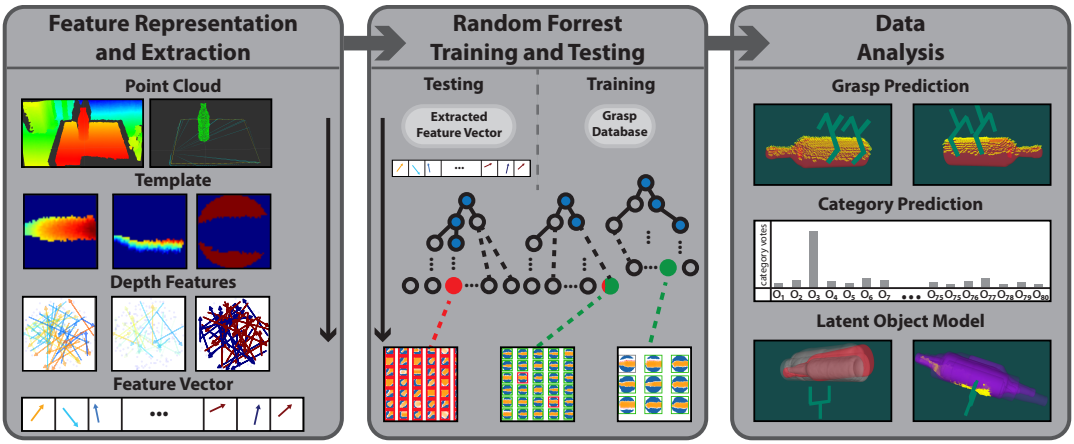


Fig. 1: Overview of the proposed system from extracting the point cloud to suggesting a grasp. (Left) Point cloud is recorded with an RGB-D camera. The point cloud needs to be sampled for grasp candidates. We use a simple table-plane based segmentation to restrict the sampling to the remaining clusters. Each grasp candidate is represented by a template as described in Sec. IV-A. Here we show the three channels: surface, occlusion and free space. We extract a depth feature from each channel of this template as described in Sec. IV-B and stack it into one feature vector. This feature vector is then used for training and at test time. (Middle) The feature vector serves as an input to a Random Forest Classifier which has been trained offline on a database. By averaging over the response of each tree in the forest, the input feature vector is classified as either stable or not. Additionally, each leaf node at which a the query data point ends up is associated with a subset of the training data. This is shown in the 'mosaics' of the bottom row in which each square represents a template of a training data point. (Right) This locally similar data can then not only be used for predicting the set of stable grasps per point cloud, but also for predicting the object category and global object shape. Regarding the latter, the left shape distribution is based on synthetic data and can be compared to the ground truth object model. On the right, you see an example for the prediction of global object shape for real point clouds.

Classifier. Therefore, we extract a feature vector very similar to the one used in [25, 26].

Essentially, we uniformly sample once n pairs of probes within a 48 pixels diameter. Let $\mathbf{u}_i = (x_u, y_u)$ and $\mathbf{v}_i = (x_v, y_v)$ refer to the i^{th} pair of two probes and let $\mathbf{C}_j$ refer to the height map for channel $j \in \{\text{sur}, \text{occ}, \text{free}\}$, then one feature component ϕ_{ij} is computed as:

$$\phi_{ij} = \phi_i(\mathbf{C}_j) = \mathbf{C}_j(\mathbf{u}_i) - \mathbf{C}_j(\mathbf{v}_i) \tag{5}$$

Per channel, we extract n features and stack them into the feature vector

$$\phi_j = [\phi_{1j}, \ \cdots, \ \phi_{nj}]^T \tag{6}$$

where we choose $n = 300$ as in our previous work [25]. The full feature vector is then

$$\Phi = [\phi_{\text{sur}}, \phi_{\text{occ}}, \phi_{\text{free}}]^T \tag{7}$$

We use the same randomly sampled pairs of probes for extracting a full feature vector for all the training and test data. An example visualization of this, can be seen on the left side of Fig. 1.

C. Random Forest Classification

We will give a brief summary of the random forest classifier. For a more in-depth review of this model, we refer to Breiman [27].

1) Classifying a Data Point: A random forest classifier consists of a set of binary decision trees. Each tree consists of a set of split nodes that each have two child nodes and a set of leaf nodes with no further child nodes. Each split node in this tree is associated with a specific feature component ϕ_{ij} and a threshold θ. Let Φ^d be the feature vector of some query data point d, then based on the value of ϕ^d_{ij}, it will either be handed further down the tree along the left branch ($\phi^d_{ij} < \theta$) or along the right branch ($\phi^d_{ij} \geq \theta$). Once the query point arrives at a leaf node, the majority ground truth label of all training data points at this leaf node forms the predicted label for this decision tree. The final label is determined by simply averaging over the predictions of each tree. If the forest has T trees, then

$$y = \frac{1}{T} \sum_{t=1}^{T} y_t(\Phi) \tag{8}$$

2) Training a Random Forest: During training of the forest, the optimization variables are the pairs of feature component ϕ_{ij} and threshold θ per split node. For each tree, a random subset of the total training data is selected that may be overlapping with the subsets for the other trees. At each node, the optimal split of the data is found by randomly sampling among all feature component candidates ϕ_{ij} and finding the best θ such that the split of the data minimizes some impurity criteria. Here, we use the Gini impurity. The open parameters for the forest training are the minimum cardinality of the set of training points at a leaf node, the maximum number of feature components to sample at each split node and the number of trees in the forest.

D. Retrieval

After training the random forest classifier as above, there is a minimum number of training data points at each leaf node. Especially in our case where the input forms a local shape representation, these reduced data sets are clusters of locally similar data. Previously, they have been used to learn local models of e.g. joint positions in human or robot pose tracking [26, 25].

In our case, each training data point is generated in simulation using a complete object mesh model of a known category (see Sec. V for more detail on the database). Given a query template that is classified by the Random Forest, we can not only predict its probability to afford a successful

grasp but also make predictions about latent variables based on the training examples at the corresponding leaf nodes. These variables can recover the global shape of the associated object. More formally, let T denote the number of trees in the Random Forest classifier and let M_t denote the number of leaf nodes in tree t after it has been trained. Per leaf node $b_{m,t}$, we have a set $\mathcal{L}_{m,t}$ containing a minimum number of training data examples that ended up at this node during training. As mentioned before, this minimum number is an open parameter of the training procedure. Given the query data point $\Phi_{\mathbf{g}}$, which is classified to afford a stable grasp, it will have ended up in one specific leaf node per tree: $\{b_{m^*,t} \text{ with } t \in \{1 \cdots T\}\}$. It is therefore associated with T leaf nodes. The retrieved set $\mathcal{S}$ of similar training examples (Eq. 4) consists of the union of all corresponding sets:

$$\mathcal{S} = \bigcup_{t \in \{1 \cdots T\}} \mathcal{L}_{m^*,t}. \tag{9}$$

V. Experimental Evaluation

In this section, we will analyze how well the Random Forest Classifier can (a) predict whether a template affords a stable grasp and (b) whether the retrieval of locally similar training data points helps to predict the global object shape.

A. Dataset

As a dataset we use our recently proposed large-scale database of grasps [17]. It contains approximately 300k different grasps that are applied to more than 700 different object instances of more than 80 different object categories. The grasp varies in terms of the roll around the approach vector towards the object (8 options) and in terms of the stand-off from the object surface (2 options). The ground truth labels of whether a grasp succeeded or not are automatically generated through physics simulation of the grasps and their validity is confirmed through crowd-sourcing. Per grasp and object, the database contains several local shape representations that are extracted from synthetic partial point clouds. These are generated by recording data from different viewpoints using a realistic RGB-D sensor model that closely resembles the characteristics of a real sensor, e.g. quantization, occlusion boundaries, and Perlin noise [25]. In total, the database contains 500k labeled data points.

B. Training

As proposed in [17], we split the objects into four different sets: (i) toy dataset of bottles, (ii) set of small objects, (iii) set of medium-sized objects and (iv) set of large objects. Additionally we consider the union of the small, medium-sized and large objects in the fifth set denoted by *all*. Each of these sets is split into a train, test and validation set. Each object instance is exclusive to one of these test sets. Using this validation set will help us much better to prevent overfitting than using cross-validation.

In the following section, we will learn one classifier for each of the five sets of objects and individually for each stand-off (since this does not influence the appearance of the

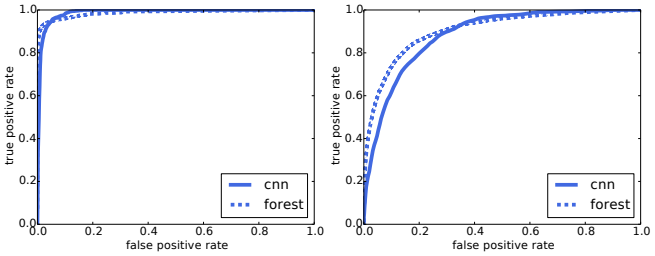


Fig. 4: ROC curves for the datasets (a) Bottles (b) Small. Dashed curves refer to the Random Forest based classifiers. Solid lines show the performance of the CNN-based model. While there is little difference in performance for the bottle dataset, for small objects the Random Forest produces significantly more true positives at a lower false positive rate.

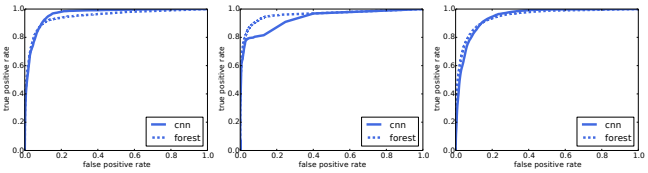


Fig. 5: ROC curves for the datasets (a) Medium (b) Large (c) All. Dashed curves refer to the Random Forest based classifiers. Solid lines show the performance of the CNN-based model. For all three datasets, there is little difference in performance especially at low false positive rates.

template). We use scikit-learn [28] as the implementation of the Random Forest Classifier.

Using the validation dataset, we performed a grid search over the three aforementioned open forest parameters: (i) number of trees in the forest, (ii) minimum number of data points at one leaf node and (iii) the maximum number of feature component to sample as split candidates at each node. We found that the optimal parameters were different per object set and gripper stand-off. Due to space constraints we omit the grid search results here. We finally used the following parameters: 15 trees, 5 minimum data points at a leaf node and 60 feature component candidates. These are essentially the average parameters for the best classifiers for each subset of data.

C. Classifying a Grasp

In this section, we show how well our Random Forests can classify whether a grasp is stable or not. As mentioned above, we split this analysis according to the five different datasets: (i) bottles, (ii) small objects, (iii) medium-sized objects, (iv) large objects and (v) the union of all these datasets denoted by *all*. We compare to the Convolutional Neural Network (CNN) from Kappler et al. [17]. Figs. 4 and 5 show the ROC curves for all five datasets. For most of them, the Random forest based classifiers perform similar to CNN-based classifiers, especially for low false positive rates. For the data set of small objects, the Random Forest outperforms the CNN. For large objects, it performs significantly better at higher false positive rates. At the most important dataset of all objects, they both perform very similarly.

Additionally, we report the average classification accuracy (ACC) for all the classifiers in Table 6. As all the datasets are

		Bottles		Small		Medium		Large		All	
		s-off 0	s-off 1	s-off 0	s-off 1	s-off 0	s-off 1	s-off 0	s-off 1	s-off 0	s-off 1
Forest	Acc	0.977	0.944	0.894	0.935	0.95	0.919	0.982	0.976	0.946	0.941
	MCC	0.895	0.748	0.524	0.488	0.629	0.549	0.612	0.364	0.585	0.486

Fig. 6: Mean of the accuracy (Acc) and Matthew's correlation coefficient (MCC) on the test set over 10 trials of learning the Random Forest Classifier. Standard deviation is not reported as it was very close to 0 for all results. Results are reported per object group (bottles, small, medium, large and all) and gripper stand-off (s-off 0 and s-off 1) from the object surface before closing the fingers. In the case of an extremely biased data set (as it is in our case), classification accuracy does not well reflect misclassifications of the minority class. MCC is a more balanced measure that is defined in $[-1, 1]$ where 1 indicates a perfect prediction and 0 a random prediction.

strongly biased towards negative data, we also included the Matthews Correlation Coefficient (MCC) that better reflects classification performance in these biased cases.

This shows that the proposed method can compete with the current state of the art on this dataset and sometimes even outperforms it.

D. Predicting Global Shape

In this section, we show how our Random Forest classifiers can be used to predict global object shape from local shape information. We are specifically considering templates that are classified to be graspable. As mentioned in Sec. IV-D, given a query shape template $\Phi_{\mathbf{g}_q}$, we can retrieve a set $\mathcal{S}$ of locally similar exemplars from the training data. Members of this set are all the training data points that during training ended up in the same leaf nodes of the trees as $\Phi_{\mathbf{g}_q}$. We use the ground truth object models that are available for each training data point in $\mathcal{S}$ to build a non-parametric distribution over the global object shape. Please note that the object models in the test set are not contained in the training data set. Therefore, a perfect shape prediction is not possible.

1) Evaluation Metric: We quantitatively evaluate how well this non-parametric distribution predicts the ground truth shape of the object from which $\Phi_{\mathbf{g}_q}$ is extracted. For each object model $\mathbf{o}_k$ associated to a locally similar template in $\mathcal{S}$, we sample a set of points $\mathbf{p}_i \in \mathcal{P}^k$ from their surface. Let $\mathcal{P}$ denote the set containing all $\mathcal{P}^k$. Furthermore, we sample a set of points $\mathbf{q}_j \in \mathcal{P}^q$ from the ground truth object model $\mathbf{o}_q$. Please note, that all the object models as well as the ground truth model are all transformed into the same coordinate frame relative to the grasp that is related to the associated template. Then for each object model $\mathbf{o}_k$ and corresponding points in $\mathcal{P}^k$ we compute the Euclidean distance d to the nearest neighbor in $\mathcal{P}^q$

$$d_{\mathbf{p}_i^k} = \underset{\mathbf{q}_j \in \mathcal{P}^q}{\operatorname{argmin}} ||\mathbf{p}_i^k - \mathbf{q}_j||_2^2 \tag{10}$$

such that we have a set $\mathcal{D}$ with

$$\mathcal{D} = \{d_{\mathbf{p}_i^k} \mid \mathbf{p}_i^k \in \mathcal{P}^k \; \forall \mathcal{P}^k \in \mathcal{P}\}. \tag{11}$$

This set represents the distribution of surface errors between the ground truth and the retrieved object models.

2) Baselines:

a) Random Leafs: For each positively classified data point, we randomly sample a different test data point that may also be classified as negative. We retrieve object models from the tree leafs that are associated to this randomly sampled data point. By comparing to the resulting surface error distribution, we want to emphasize that there is no unfair bias in the data set.

b) Unsupervised Clustering: We use *Random Forest Embedding* (RFE) [29] to perform a very similar clustering as done for Random Forest Classification, however in an *unsupervised* manner. That means that during training, the split node parameters are *not* chosen to optimize some impurity measure of classification labels. However, we perform the same retrieval of training data points for some test point as for the supervised model. By comparing to the resulting surface error distribution, we want to emphasize that the information on grasp suitability provides cues on global object shape.

RFE produces a very high-dimensional, sparse and binary feature vector. We train a Bernoulli model for grasp stability classification that is well suited for this type of input data. This is only used to select positively classified test points. The classification accuracy of this model is lower than that of the CNN and Random Forest. However, in this baseline, we are only interested in the performance of the unsupervised clustering and use the classification for some filtering.

3) Results: Fig. 7 shows four example predictions of global object shape from the retrieved object models (2nd row) associated to templates in matching leaf nodes (1st row). The ground truth object model is shown in red. We can observe that all the templates within a leaf node are locally similar. Furthermore, they predict well the global shape of the object as we will quantitatively shown further below. The third row shows object models that are retrieved from leaf nodes which are matching a randomly selected test data point (*Random Leafs* baseline). Especially for leaf nodes which contain a majority of negative templates, the shapes and poses of the retrieved objects vary a lot.

Fig. 8 visualizes the mean and standard deviation of this surface error (red) with increasing distance from the centroid of the template. The *Random Leafs* baseline is shown in blue and the *Unsupervised Clustering* baseline in green. We can observe that the proposed method can well predict global object shape. Even at a distance of 20cm from the grasp position, the mean surface error between ground truth model and shape distribution is approximately 2cm for the 'all' object set. As expected, the error is also significantly lower than for the *Random Leafs* baseline. The proposed method also outperforms the *Unsupervised Clustering* baseline where the retrieved locally similar exemplars are not predicting global object shape nearly as well. Also the standard deviation of the error is much lower for the proposed method than for the two baselines on all data sets. Furthermore, the different range of object sizes in the datasets is reflected in the error distribution. For example, the maximum mean error for small objects does not exceed approximately 1.9cm.

Apart from the object mesh model per retrieved training data point, we also know their ground truth label in terms of grasp stability. Therefore, we can additionally filter the set $\mathcal{S}$ such that it only includes object models that are associated with positive grasp templates. In Fig. 9, we visualize the mean and standard deviation of the surface error (red) when computed over this filtered set. Also for the retrieved object models using the two baselines (blue and green), we only

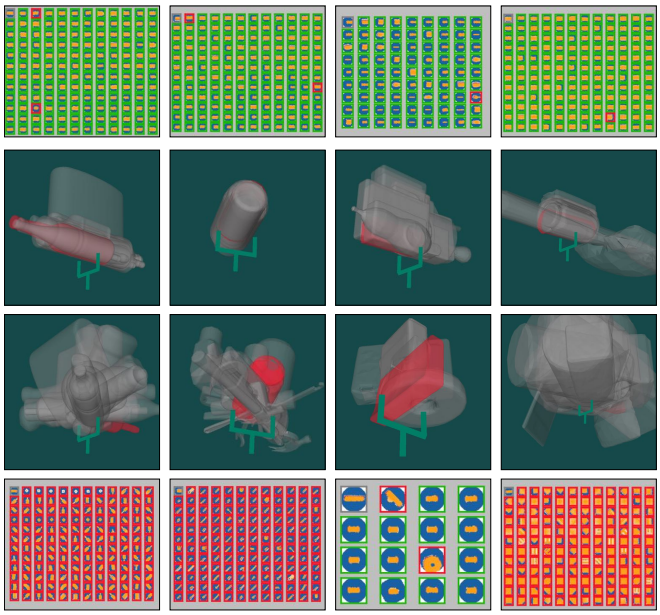


Fig. 7: Comparison of shape distribution for four test templates formed of retrieved objects from matching (second row) or non-matching leaf nodes (third row). (First row) Visualization of the local shape templates at a leaf node in which the query template ended up in. The query template is always shown in the top left and is framed grey. Templates with a positive ground truth label are framed green; the others red. (Second row) Visualization of the shape distribution given the object models associated with the templates in the *matching* leaf node (grey). Ground truth object in red. (Third row) Visualization of the shape distribution given the object models associated with the templates in a *non-matching* leaf node. Ground truth object in red. (Fourth row) Visualization of the local shape templates at a *non-matching* leaf node. Color coding as in the First row. When comparing the second to the third row, it becomes apparent that objects retrieved from matching leaves resemble the ground truth object shape much better than objects retrieved from non-matching leafs.

include those that have a positive grasp label. As a first observation, we can see that the mean surface error and its standard deviation indeed decreased for all objects sets and all retrieval methods. Especially for the *Unsupervised Clustering* baseline it went down significantly. The results for the proposed method changed only insignificantly. The *Random Leafs* baseline performs consistently worse except for the medium object set.

4) Discussion: On the one hand, the comparison to the *Random Leafs* baseline (especially when only retrieving positive training examples) reveals that local similarity has a significant contribution to predict global object shape. Graspability information alone does not provide enough discrimination to recover this. On the other hand, the comparison to the *Unsupervised Clustering* baseline reveals that local similarity alone does not provide enough information to infer global object shape. We have shown that a combination of local similarity and information of graspability allows to best recover global object shape from local information.

If a robot can predict global object shape from partial information only, it can leverage this to improve grasp and manipulation planning. This is especially useful, if the object is unknown and stands within a cluttered environment. We expect that accumulating these predictions from all local shape templates that are extracted from a point cloud observation, will help to reinforce some hypotheses and prune others. Furthermore, hypotheses can also be pruned by verifying them against the observation of the entire environment. It could be used in a hypothesis verification system similar to Aldoma et al. [30] but without being restricted to known objects.

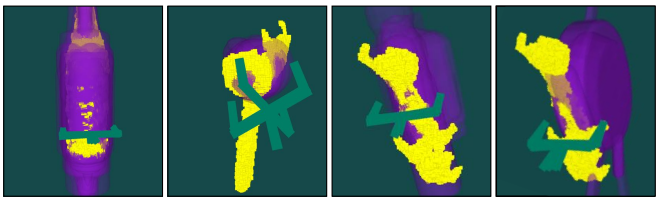


Fig. 10: Global object shape prediction from a real point cloud segment in yellow that is extracted as shown in Fig. 1. In purple are all the retrieved object models. The green grippers show the associated grasps. (Left) Bottle (Middle Left) Hammer (Middle Right) Impact Wrench (Right) Impact Wrench

E. Example Results on Real Data

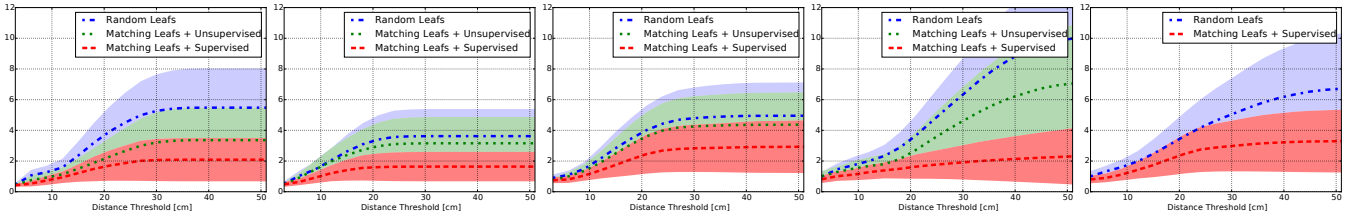
We present qualitative results of predicting global object shape on real point cloud data in Fig. 10. As a first observation, we can see that the suggested grasps are good although they may not be task-relevant as for example for the hammer. This confirms that grasps are generalized across objects that share similar parts which afford grasps. In these examples, we can also observe a remaining ambiguity when inferring global shape from local shape. For example it is not clear which way the bottle is rotated. Furthermore, some local shapes are not good at predicting global shape as for example the fruit fit to the head of the hammer. Regarding the example of the impact wrench on the right, rather large objects are fit to it. These could however be filtered out based on back projecting them into the depth image and cross checking with the visual evidence for the whole scene.

VI. Conclusions

We proposed a novel method that enables a robot to autonomously decide how to grasp an unknown object given only noisy and partial information that is obtained from an RGB-D camera. Our method combines a discriminative model to classify local shape information as either graspable or not with a retrieval method for predicting global object properties. We showed how our discriminative model has a similar performance as the state-of-the-art on this dataset. Furthermore, we showed how the retrieved training data points can provide a coherent prediction of global object shape. Lastly, we demonstrated how the model works on a few real-world data points although it has been trained on simulated data.

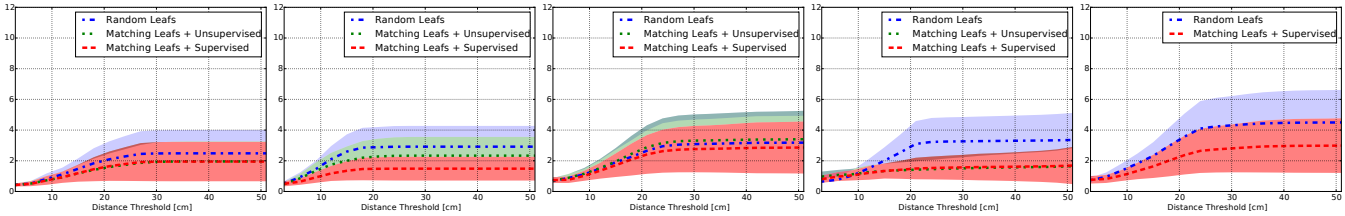
There are many interesting directions that this idea of combining discriminative methods with retrieval. For example, we can study whether other global object properties can be predicted from locally similar information. Preliminary results are promising for predicting the object category. Furthermore, the information at the leaf nodes can also be compressed by for example using Gaussian process regression to predict the global object shape.

Also promising is the accumulation of global shape prediction from all local shape templates. They could be used to populate the environment and then pruned by checking with the visual evidence. High-quality proposals for object segmentation in cluttered environments may be another outcome of this method.



(a) Bottles (b) Small (c) Medium (d) Large (e) All

Fig. 8: Mean and standard deviation over surface error (see Eq. 10) between retrieved object models and ground truth over the different object sets (bottles, small, medium, large and all). Red: Object models retrieved from leafs in which the query template ended up. Blue: *Random Leafs* baseline where object models are retrieved from random leafs. Green: *Unsupervised Clustering* with object models retrieved using an unsupervised random tree embedding. The plot on the 'all' dataset does not contain results on the unsupervised baseline. However, given the results on the subsets, we expect similar results.



(a) Bottles (b) Small (c) Medium (d) Large (e) All

Fig. 9: Mean and standard deviation over surface error (see Eq. 10) between retrieved object models and ground truth over the different object sets (bottles, small, medium, large and all). Different from Fig. 8 only positive grasp templates are selected from the leafs. Red: Object models retrieved from leafs in which the query template ended up. Blue: *Random Leafs* baseline where object models are retrieved from random leafs. Green: *Unsupervised Clustering* with object models retrieved using an unsupervised random tree embedding. The plot on the 'all' dataset does not contain results on the unsupervised baseline. However, given the results on the subsets, we expect similar results.

Finally, we want to investigate how the predicted information can be exploited by a grasp planner and controller to finally execute the resulting grasp on a real robot.

REFERENCES

[1] L. Righetti, M. Kalakrishnan, P. Pastor, J. Binney, J. Kelly, R. Voorhies, G. S. Sukhatme, and S. Schaal, "An autonomous manipulation system based on force control and optimization," *Auton. Robots*, vol. 36, no. 1-2, pp. 11–30, 2014.

[2] N. Hudson, T. Howard, J. Ma, A. Jain, M. Bajracharya, S. Myint, C. Kuo, L. Matthies, P. Backes, P. Hebert, T. J. Fuchs, and J. W. Burdick, "End-to-end dexterous manipulation with deliberate interactive estimation," in *IEEE Int. Conf. on Robotics and Automation, ICRA 2012, 14-18 May, 2012, St. Paul, Minnesota, USA*, 2012, pp. 2371–2378.

[3] M. Kazemi, J. Valois, J. A. Bagnell, and N. S. Pollard, "Robust object grasping using force compliant motion primitives," in *Robotics: Science and Systems VIII, University of Sydney, Sydney, NSW, Australia, July 9-13, 2012*, 2012.

[4] C. Ferrari and J. Canny, "Planning optimal grasps," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 1992.

[5] A. T. Miller and P. K. Allen, "Graspit! a versatile simulator for robotic grasping," *Robotics & Automation Magazine, IEEE*, 2004.

[6] R. Diankov, "Automated construction of robotic manipulation programs," Ph.D. dissertation, CMU, Robotics Institute, Aug 2010.

[7] J. Bohg, A. Morales, T. Asfour, and D. Kragic, "Data-driven grasp synthesis: A survey," *IEEE Transactions on Robotics*, 2014.

[8] J. Bohg, M. Johnson-Roberson, B. León, J. Felip, X. Gratal, N. Bergström, D. Kragic, and A. Morales, "Mind the Gap - Robotic Grasping under Incomplete Observation," in *IEEE Int. Conf. on Robotics and Automation*, May 2011.

[9] M. Bjorkman, Y. Bekiroglu, V. Hogman, and D. Kragic, "Enhancing visual perception of shape through tactile glances," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ Int. Conf. on*, Nov 2013, pp. 3180–3186.

[10] S. Dragiev, M. Toussaint, and M. Gienger, "Uncertainty aware grasping and tactile exploration," in *IEEE Int. Conf. on Robotics and Automation*, 2013.

[11] A. Saxena, J. Driemeyer, and A. Y. Ng, "Robotic grasping of novel objects using vision," *The Int. Jour. of Robotics Research (IJRR)*, vol. 27, no. 2, pp. 157–173, Feb. 2008.

[12] R. Detry, C. H. Ek, M. Madry, and D. Kragic, "Learning a dictionary of prototypical grasp-predicting parts from grasping experience," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2013.

[13] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," *The Int. Jour. of Robotics Research (IJRR)*, 2014.

[14] A. Herzog, P. Pastor, M. Kalakrishnan, L. Righetti, J. Bohg, T. Asfour, and S. Schaal, "Learning of grasp selection based on shape-templates," *Autonomous Robots*, 2013.

[15] O. Kroemer, E. Ugur, E. Oztop, and J. Peters, "A kernel-based approach to direct action perception," in *Int. Conf. on Robotics and Automation (ICRA)*, 2012.

[16] Q. V. Le, D. Kamm, A. F. Kara, and A. Y. Ng, "Learning to grasp objects with multiple contact points." in *ICRA*, 2010, pp. 5062–5069.

[17] D. Kappler, J. Bohg, and S. Schaal, "Leveraging big data for grasp planning," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2015.

[18] J. Redmon and A. Angelova, "Real-time grasp detection using convolutional neural networks," in *IEEE International Conference on Robotics and Automation, ICRA 2015, Seattle, WA, USA, 26-30 May, 2015*, 2015, pp. 1316–1322.

[19] C. Goldfeder, M. Ciocarlie, H. Dang, and P. K. Allen, "The columbia grasp database," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2009.

[20] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser, "The Princeton shape benchmark," in *Shape Modeling International*, Jun. 2004.

[21] K. Hsiao, S. Chitta, M. Ciocarlie, and E. G. Jones, "Contact-reactive grasping of objects with partial shape information," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, Taipei, Taiwan, Oct 2010, pp. 1228 – 1235.

[22] P. Pastor, L. Righetti, M. Kalakrishnan, and S. Schaal, "Online Movement Adaptation based on Previous Sensor Experiences," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, San Francisco,USA, Sep 2011, pp. 365 – 371.

[23] J. Romano, K. Hsiao, G. Niemeyer, S. Chitta, and K. Kuchenbecker, "Human-inspired robotic grasp control with tactile sensing," *IEEE Trans. on Robotics*, vol. 27, no. 6, pp. 1067 – 1079, Dec 2011.

[24] Y. Lin and Y. Jeon, "Random forests and adaptive nearest neighbors," *Jour. of the American Statistical Association*, vol. 101, no. 474, pp. 578–590, 2006.

[25] J. Bohg, J. Romero, A. Herzog, and S. Schaal, "Robot arm pose estimation through pixel-wise part classification," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, Hongkong, China, 2014.

[26] J. Shotton, R. Girshick, A. Fitzgibbon, T. Sharp, M. Cook, M. Finocchio, R. Moore, P. Kohli, A. Criminisi, A. Kipman, and A. Blake, "Efficient human pose estimation from single depth images," *IEEE Trans on Pattern Analysis and Machine Intelligence*, vol. 99, p. 1, 2012.

[27] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[28] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Jour. of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[29] F. Moosmann, B. Triggs, and F. Jurie, "Fast discriminative visual codebooks using randomized clustering forests," in *Advances in Neural Information Processing Systems (NIPS 19)*, B. Schölkopf, J. Platt, and T. Hoffman, Eds. Cambridge, MA: MIT Press, 2006, pp. 985–992.

[30] A. Aldoma, F. Tombari, L. di Stefano, and M. Vincze, "A global hypotheses verification method for 3d object recognition," in *12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part III*, 2012, pp. 511–524.