Appearing in the *IEEE International Conference on Robotics and Automation*, 2013.
This copy is for personal use only.
978-1-4673-5643-5/13/$31.00 © 2013 IEEE
Prints are available at http://dx.doi.org/10.1109/ICRA.2013.6630635.

Learning a Dictionary of Prototypical Grasp-predicting Parts from Grasping Experience

Renaud Detry Carl Henrik Ek Marianna Madry Danica Kragic

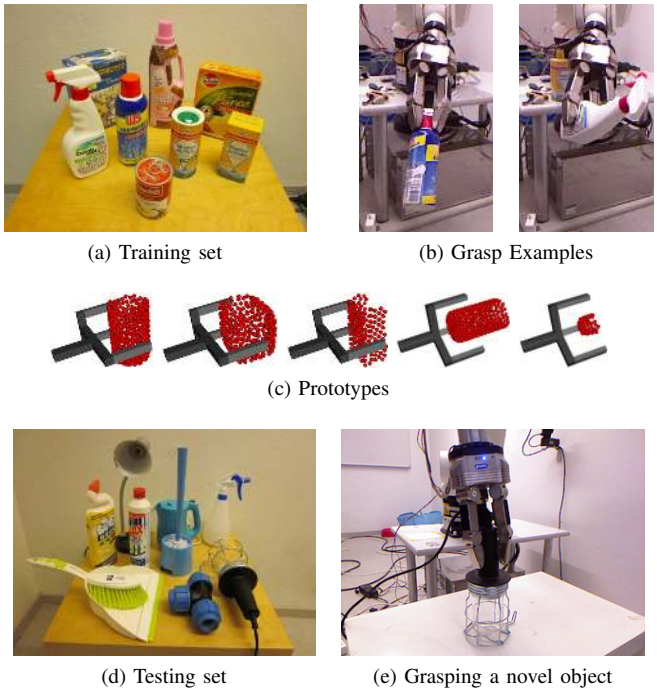
***Abstract*— We present a real-world robotic agent that is capable of transferring grasping strategies across objects that share similar parts. The agent transfers grasps across objects by identifying, from examples provided by a teacher, parts by which objects are often grasped in a similar fashion. It then uses these parts to identify grasping points onto novel objects. We focus our report on the definition of a similarity measure that reflects whether the shapes of two parts resemble each other, *and* whether their associated grasps are applied near one another. We present an experiment in which our agent extracts five prototypical parts from thirty-two real-world grasp examples, and we demonstrate the applicability of the prototypical parts for grasping novel objects.**

I. Introduction

This paper addresses the problem of robotic grasp planning. We present a method that allows a robot to compute, from a single object snapshot produced by a Kinect camera, the position, orientation, and preshape to which it needs to bring its manipulator in order to grasp the object. A substantial challenge in grasp planning is to generate workable finger placements while one finger or more must unavoidably be applied onto surfaces that are behind the object, and thus not perceived by the robot. To address this problem, planning algorithms usually exploit prior object knowledge in order to postulate the shape of occluded regions and devise a workable strategy. For instance, when working in controlled environments, we can provide robots with 3D shape models and grasp parameters for every object. From a single snapshot, the robot can recognize and estimate object poses, which leads to a reconstruction of occluded faces and the generation of accurate grasps. However, when robots are deployed in uncontrolled environments such as houses or hospitals, hard-coding grasping strategies for every object that the robot may encounter quickly becomes unpractical. In order to work with unknown objects, assumptions on shape regularity, such as symmetry [7], [22], [39], may be used to fill occluded regions and properly formulate finger placements. Unfortunately, there is no guarantee on the extent to which such assumptions apply.

In order to overcome the limitations associated to hard-coded means of predicting 3D shapes, authors have increasingly looked for means of extracting from experimental data a mapping that links visual cues to grasp parameters. This way, a robot can acquire experience and progressively learn to grasp new kinds of objects [10], [23], [31], [33].

In this paper, we present a method that allows a robot to learn to formulate grasp plans from visual data obtained from a 3D sensor. Our method relies on the identification of prototypical parts by which objects are often grasped. To this end, we provide the robot with means of identifying, from a set of grasp examples, the 3D shape of parts that are recurrently observed within the manipulator during the grasps. Our approach effectively compresses the training data, generating a dictionary of prototypical parts that is an order of magnitude smaller than the training dataset. As prototypical parts are extracted from grasp examples,



(a) Training set (b) Grasp Examples (c) Prototypes (d) Testing set (e) Grasping a novel object

Fig. 1: Transferring grasps to novel objects. From grasps demonstrated on a set of training objects (Figures (a) and (b)), the agent extracts a dictionary of prototypes (Figure (c)). These prototypes allow the agent to grasp novel objects that are partly similar to the training objects, such as those of Figure (d). Figure (e) shows an example of the application of the fifth prototype to an object whose global shape is unlike any of the training objects, but that present a part that resembles the fifth prototype.

R. Detry, C. H. Ek, M. Madry, and D. Kragic are with the Centre for Autonomous Systems and the Computer Vision and Active Perception Lab, CSC, KTH Royal Institute of Technology, Stockholm, Sweden. Email: {detryr,chek,madry,danik}@csc.kth.se

This work was supported by the Swedish Foundation for Strategic Research, the Swedish Research Council, the Belgian National Fund for Scientific Research (FNRS), and the EU projects CogX (FP7-IP-027657) and TOMSY (IST-FP7-Collaborative Project-270436).

each of them automatically inherits a grasping strategy that parametrizes (1) the position and orientation of the manipulator with respect to the part, and (2) the finger preshape, i.e., the configuration in which fingers should be set prior to grasping. When a novel object appears, the robot tries to fit the prototypical parts to a 3D snapshot (e.g., from a Kinect) that partially captures the object. The grasp associated to the part that best fits the snapshot can be executed to manipulate the object. In effect, fitting prototypes implicitly postulates the object shape in occluded regions, allowing the robot to formulate finger placements that match all sides of the object.

Through this work, we argue that it is critical to take both visual data and grasping experience into account in the process of defining grasp-predicting shapes. Parsing objects into parts has been a topic of interest in computer vision for decades [8], [18], [19]. It thus appears tempting to tap into that literature in order to define parts to which we may later-on associate grasps. Contrary to this line of thought, we argue that a part decomposition that is suitable for object recognition is not necessarily suitable for grasp prediction. Instead, we aim to let prototypical parts emerge from both object shape and grasp examples. A key effect is that the shape and the spatial extent (or size) of the prototypes generated by our method directly result from the available grasp data. Our approach involves an explicit search for recurrent patterns within the agent's visuomotor experience, which leads to the identification of parts that directly predict grasp applicability.

Figure 1 illustrates the concept of our method. The robot is first taught, via teleoperation, to grasp the objects of Figure 1a, as shown in Figure 1b. From these data, the robot generates the dictionary of prototypes shown in Figure 1c. It can then devise grasping strategies for novel objects that partly resemble those of the training set, such as the objects in Figure 1d. Figure 1e shows a grasp suggested by fitting the fifth prototype onto a lamp.

The main contribution of this paper is the application of our method to real-world data. Our previous work [14] included a proof-of-concept experiment on synthetic data, where only the learning of parts was demonstrated. In this paper, we present an experiment where the agent computes a set of prototypes from real-world grasps demonstrated by a teacher, and we test the resulting model by executing 55 grasps on a real-world robot platform. In order to manage the variability of real-world data, we developed a new method for computing prototypical parts. The second contribution of this paper is a new part similarity measure that allows the robot to identify parts that have a similar shape, while being robust to a certain amount of variation in the absolute pose of each part, as grasps demonstrated on similar parts will often present slightly different approach vectors. Finally, this paper details the procedure that allows us to align prototypes to partial 3D snapshots in order to grasp new objects.

II. Related Work

In robotics, mainstream grasp planning has traditionally relied on force analysis [6], [37]. Force analysis exploits object shape models, possibly augmented with frictional and inertial parameters, to compute through the application of the laws of mechanics whether the net force applied by a manipulator onto an object is sufficient to bind the object to the manipulator. It has been shown that force analysis can be integrated into action-perception loops to implement useful behaviors [2], [27]. Unfortunately, this approach suffers from a number of shortcomings. As it requires 3D models of the objects that need to be grasped, it cannot directly handle novel objects. Moreover, finding an optimal grasp in the force-analysis sense is computationally expensive.

Methods that provide a more direct link from visual perception to grasp parameters have soon emerged and become increasingly popular. Several groups have developed algorithms that compute grasp parameters from a single view of an object [26], [32]. Instead of hard-coding the function that computes grasp parameters from vision, other authors have looked into vision-action policies whose parameters are learned from experimental data [10], [23], [31]. This way, grasp-related object properties can be captured implicitly through interaction.

Authors have studied the association of grasping strategies to various kinds of visual cues. Grasps associated to local visual features [33], [30] have the advantage of being easily transferable across objects, as many objects share similar components. However, local features suffer from a poor geometric resolution, which makes it difficult to accurately compute the 6D pose of a gripper, let alone finger preshape parameters. Conversely, grasps associated to a model of a whole object [11], [15], [20] benefit from increased geometric robustness, but the resulting models are less likely to apply to novel objects. Authors have explored this trade-off between transferability and robustness by associating grasps to object parts of varying size [1], [4], [13], [14], [17], [21], [24], [29], [38], [40]. Miller et al. [29] have manually constructed a set of shape primitives (cone, sphere, cube, etc.) and associated grasping parameters, and demonstrated in their *GraspIt!* simulator that fitting the shapes to novel objects allows the robot to quickly generate workable grasping strategies. Sweeney et al. [38] have defined grasp-predicting parts by parsing objects into sets of ellipsoidal primitives. With the advent of cheap 3D sensors, methods based on 3D data have started to flourish [14], [21], [28], [40]. Kroemer et al. [28] presented a part-based model of grasp and task parameters, where both the shape of a 3D part, and the trajectory the robot needs to follow, are encoded. Herzog et al. [21] and Zhang et al. [40] presented two data-driven approaches where a part describes the object shape in a fixed-size region around a grasping point, and each grasp example yields a part. By contrast, in our work, a grasp example only "votes" for the potential inclusion of a prototype into the dictionary, which provides us with a means of controlling the size of the dictionary in order to keep the computational cost of planning a grasp onto a novel object reasonably low.

An important distinctive point of our work is that, as in our previous work [13], [14], [17], we provide the agent with means of optimizing the transferability-robustness trade-off

Fig. 2: Grasp preshapes. In this work, grasps are executed by setting the manipulator to either one of a parallel (first image) or cylindric (third image) preshape, then closing the fingers towards a goal configuration (second and fourth images) until resistance is encountered.

mentioned above, *by allowing it to select prototypical parts of varying size*, depending on their occurrence statistics in the training database. The result is a compact dictionary of parts that lend themselves to grasping. This paper goes beyond our previous work [14] by providing: (1) a novel shape similarity measure that is robust to local pose variations, (2) a clustering approach that works directly in the space induced by the similarity measure, and (3) an experiment that shows the applicability of our work to real-world data.

III. Method

The concept of our approach is to identify, within the agent's visuomotor experience, recurrent associations of object *parts* and successfully executed grasps. We proceed as follows. We first collect a number of grasp examples. From each example, we segment parts of varying sizes in the vicinity of the point at which the grasp is applied. We then compute pairwise similarities between all the resulting part *candidates*. Pairwise similarities allow us to identify dense clusters of similar parts, i.e., parts that are shared by multiple objects. We select the central part of each cluster. These parts, along with their corresponding grasp parameters, altogether form a dictionary of prototypes that will later be used to suggest grasps onto novel objects. As only cluster centers are selected, we are able to limit the dictionary to a size that is an order of magnitude smaller than the number of initial grasp examples.

A grasp prototype is composed of a shape model, in the form of a point cloud, along with the corresponding wrist pose and hand preshape. A novel object is grasped by aligning the prototypes to a 3D snapshot of the object. The grasp parameters of the prototype that best fits the snapshot are used to execute a grasp. A grasp is executed by bringing the manipulator to the correct pose and preshape, then closing the fingers to apply a fixed force onto the object. In this paper, we consider two different preshapes, that we refer to as *parallel* and *cylindric* (see Figure 2). We emphasize that although grasps begin in either of these two preshapes, the final finger positioning is not limited to two configurations. Instead, the compliance of the hand to an object's shape yields a different finger configuration for each object.

A. Grasp Examples

Our method is trained with a set of grasp examples. Each example is composed of the 3D shape model of the object being grasped, the 6D pose of the object, the 6D pose of the manipulator's wrist at the time of the grasp, and the preshape to which the manipulator was set prior to the grasp. We assume that the objects used for training are known to the robot, therefore full 3D shape models are available for these. We note that while our method requires full 3D surface models for training, it is applicable to grasping new object for which only a partial 3D snapshot is available.

B. Part Candidates

The first step of our approach is to generate a set of part candidates, by segmenting shapes of various size near grasping points. Generating object segments amounts to sampling the space of possible parts. Sampling this space is necessary, as analyzing the space of parts continuously would be computationally prohibitive.

Part candidates are generated by segmenting objects along a set of predefined box-like regions of interest (ROIs). These ROIs are predefined by hand. Each grasp preshape constrains the shape of the object being grasped. For instance, a cylindric preshape would not be applicable to an object that is elongated in a direction parallel to the wrist of the hand. As a result, we define ROIs separately for each preshape. In this work, we consider three different ROIs for both the parallel and cylindric preshapes.

Our next step is to search for recurring shapes within the part candidates. To this end, we first define a measure of dissimilarity between parts (Section III-C). This measure then allows us to identify groups of similar parts, by clustering part candidates in the space induced by our dissimilarity measure (Section III-D).

C. Similarity Measure

This section presents a measure of the similarity between pairs of part candidates. Intuitively, the measure reflects whether the shapes of two parts resemble each other, *and* whether their associated grasps are applied at the same place. In this sense, a candidate composed of a cylindric part associated to a sideways grasp would not be similar to another candidate composed of the same cylindric part associated to a top grasp. We note that grasp preshapes also play an important role in prototyping. Preshapes could be taken into account in the similarity measure, marking grasps that have different preshapes as clearly dissimilar. This is however not the idea followed in this paper. Instead, as explained in the next sections, we separate the data that correspond to different preshapes and cluster them separately.

The similarity between two part candidates is computed by expressing the point clouds of the two parts in the reference frames defined by the associated grasps, and computing how closely their surfaces match. In this way, we simultaneously compute whether the part shapes *and* their poses relative to the manipulator are similar.

Let $P = \{a_i\}_{i\in[0,n]}$ and $Q = \{b_i\}_{i\in[0,m]}$ denote the point-cloud representations of two parts, with all a_i's and b_i's belonging to $\mathbb{R}^3$. To the end of defining the similarity between P and Q, we define *object-surface distributions*

$\phi_P(x)$ and $\phi_Q(x)$ from P and Q [16]. The value of ϕ_P (resp. ϕ_Q) at a given point $x \in \mathbb{R}^3$ is inversely proportional to the distance between x and its closest neighbor in P (resp. Q). Object-surface distributions are essentially computed by centering a Gaussian function onto each input datapoint, and summing the Gaussians. The similarity between two parts is then expressed by

$$s^*(P,Q) = \int_{\mathbb{R}^3} \phi_P(x)\phi_Q(x)\mathrm{d}x, \quad (1)$$

which produces values in $[0,1]$. This expression is solved by Monte Carlo integration [9], averaging values of $P(x)$ at points drawn randomly from $Q(x)$

$$s^*(P,Q) \simeq \frac{1}{M}\sum_{i=1}^{M}\phi_P(x_i) \quad \text{with} \quad x_i \sim \phi_Q(x). \quad (2)$$

For further details on Monte Carlo integration, we refer the reader to our previous work [13]. Additionally, an implementation of s^* is publicly available online [12].

The similarity s^* defined above could potentially be used for our purpose. However, this measure is highly sensitive to the relative pose of the parts with respect to their grasp. Although we wish for two cylinders grasped from the top and side to appear as dissimilar prototypes, our measure does need to account for *some* variation in the part-grasp relative pose. For example, although a sideways grasp onto a cylinder should ideally approach along a vector that is perfectly perpendicular to the cylinder's surface, in practice, there will often be some deviation. Consequently, our measure needs to allow two part candidates with a slight relative part-grasp deviation to still appear similar. To this end, we define a second similarity measure that is computed by considering the similarities s^* between P and a number of parts Q_i, where Q_i is generated by applying a small random pose transformation to Q. Small pose perturbations are computed by applying a rotation and translation drawn from zero-mean isotropic distributions on $SO(3)$ and $\mathbb{R}^3$. Random rotations $\hat{r}$ are drawn from the Von Mises-Fisher distribution centered on the identity rotation. The Von Mises-Fisher distribution is the $SO(3)$ equivalent of a Gaussian. Its expression is proportional to

$$v(r) = e^{\sigma_r\, q^T r} + e^{-\sigma_r\, q^T r}, \quad (3)$$

where q is a unit quaternion that corresponds to the identity rotation, and σ_r is a bandwidth parameter that is set in our experiments to produce rotations mainly distributed in the range $[0°, 10°]$. Random translations are drawn from an isotropic trivariate Gaussian of zero mean and standard deviation σ_t. In our experiments, σ_t is set to five centimeters. These parameters have been chosen by inspection of our data (Section IV). Their value needs to be set proportionally to the variance in the poses of the grasps demonstrated to the robot.

As the next section requires a set of pairwise *dis*similarities, we define the dissimilarity between P and Q with

$$d(P,Q) = \left(1 - \max\left\{s^*\left(P, T_{\hat{t}_i,\hat{r}_i}(Q)\right)\right\}_{i=1}^{\ell}\right)^p, \quad (4)$$

where $\hat{t}_i$'s and $\hat{r}_i$'s correspond to random translations and rotations generated as explained above, and T operates a rigid transformation by first rotating then translating Q by $\hat{r}_i$ and $\hat{t}_i$. The parameter p is a positive exponent which allows us to globally influence the transferability-robustness trade-off. Values smaller than 1 will only allow very similar parts to be near each other in the space induced by d. Values greater than 1 will encourage generalization, by allowing slightly different parts to be near each other in the space induced by d. In our experiments, p is set to 2.

We note that s^* is similar in spirit to the dissimilarity measure presented in our previous work [14]. The contribution of this paper lies in Eq. 4, which is explicitly robust to pose variations. Applying either s^* or the dissimilarity measure presented in our previous work [14] directly to the real-world problems studied below made similar parts appear dissimilar, due to small differences in the placement of the robot's hand with respect to the objects. This approach led to no meaningful results. By contrast, the measure of Eq. 4 lead to positive results, as demonstrated in Section IV.

By contrast to traditional surface-alignment methods such as ICP [5], the method above penalizes transformations proportionally to their amplitude (as defined by the Euclidean distance to $(0,0,0)$ and the geodesic distance on the unit-quaternion sphere to the identity rotation). In other worlds, we limit the pose transformation search space to a region centered on the identity transformation. An implementation of the measure presented in this section is publicly available online [12].

D. Prototypes

Our aim of is to construct a compact dictionary of prototypical graspable parts. We wish to find prototypes that generalize over objects but still span a representation with sufficient expressive power. We proceed similarly to our previous work [14].

Given a set of parts $\{P_i\}_{i\in[1,p]}$, we can compute the dissimilarity between each pair of parts $\mathbf{D}_{ij} = d(P_i, P_j)$. In order to proceed, we assume that the measure is close to metric for the subspace spanned by the training data. This allows us to, in a two stage process, recover a geometric representation. First, we convert the dissimilarity matrix $\mathbf{D}$ to an inner-product matrix $\mathbf{K}$, through application of the kernel trick [34], [14]. Second, we compute the best positive semi definite approximation of $\mathbf{K}$ under the Frobenius norm,

$$\hat{\mathbf{C}} = \mathrm{argmin}_{\mathbf{C}} ||\mathbf{K} - \mathbf{C}||_{\mathrm{F}}^2. \quad (5)$$

The above can be solved in closed form through a simple eigenvalue decomposition. We note that this procedure is almost identical to our previous work [14], with the difference that $\mathbf{C}$ is not constrained to rank two anymore.

Having resolved a geometrical representation of the data we wish to partition the space in such a manner that we

can discover atomic classes of grasps independent of object type. We cluster the data with a graph-based/normalized-cuts approach [36] applied to the matrix $\hat{\mathbf{C}}$, which allows us to compute partitions without making assumptions about the structure of each cluster. We refer the reader to our previous work for more details on this method [14].

E. Grasping Novel Objects

In order to grasp a novel object, the agent captures a 3D snapshot of the object and compares is to the prototypes it has acquired. The prototype that best fits the data is selected to parametrize the position, orientation and preshape of the grasp. Once the manipulator is set to the intended configuration, its fingers are closed until they apply a fixed force onto the object.

Prototypes are aligned to the object snapshot using a sample-based pose estimation method [16]. The method works by exploring the space of possible prototype poses (t, r) in search for the pose that maximizes the surface similarity measure discussed above (1). Additionally, we exploit our knowledge of the placement of fingers onto prototypical parts to prevent the selection of poses that would lead fingers to collide with the surfaces captured by the snapshot (including the table), and we exploit our knowledge of arm kinematics to prevent selecting unreachable grasps.

Let us denote by $T_{t,r}(\cdot)$ a function that transforms its argument by applying to it a rotation r followed by a translation t. Let us denote by S the point cloud obtained from a 3D snapshot, by P the point cloud of a prototype, and by B a collection of 3D boxes that cover the volume occupied by the robot's fingers in the grasp from which the prototype was learned. We define the reachability of a grasp pose (t, r), given the snapshot S, as

$$R_{t,r}\left(B, S\right) = \begin{cases} 1 & \text{if } S \cap T_{t,r}(B) = \emptyset \text{ and } \exists\, \mathbf{IK_{t,r}}, \\ 0 & \text{else,} \end{cases} \tag{6}$$

where $\exists\, \mathbf{IK_{t,r}}$ denotes whether there exists an inverse kinematics solution for grasp pose (t, r). The collision-free, best-fitting prototype pose is given by

$$\arg\max_{(t,r)} s^*\left(T_{t,r}\left(P\right), S\right) R_{t,r}\left(B, S\right). \tag{7}$$

Its value is computed via simulated annealing [25] on a Markov chain [3] whose invariant distribution is an increasing power of $s^*\left(T_{t,r}\left(P\right), S\right) R_{t,r}\left(B, S\right)$. The chain is defined with a mixture of two local- and global-proposal Metropolis-Hastings transition kernels. Intuitively, the method alternates between a hill-climbing policy that reveals local pose maxima, and random jumps in the pose space that allow multiple local maxima to be discovered.

The prototype that yields the highest surface similarity (7) is used for grasping the object.

IV. Experiment

This section presents an experiment that demonstrates the applicability of our approach on a real robot platform. The results of the experiment are: (1) a dictionary of parts learned from grasps demonstrated to the robot (Section IV-A), and (2) the exploitation of this dictionary to grasp novel objects (Section IV-B).

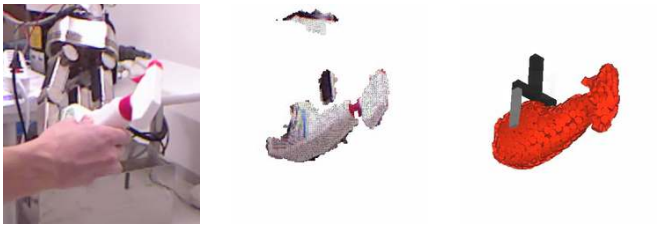


Fig. 3: Teaching a grasp to the robot. The second image shows a 3D snapshot of the grasp, taken with a Kinect camera. The third image shows the object model aligned to the correct pose, and the pose of the gripper.

A. Dictionary of Prototypes

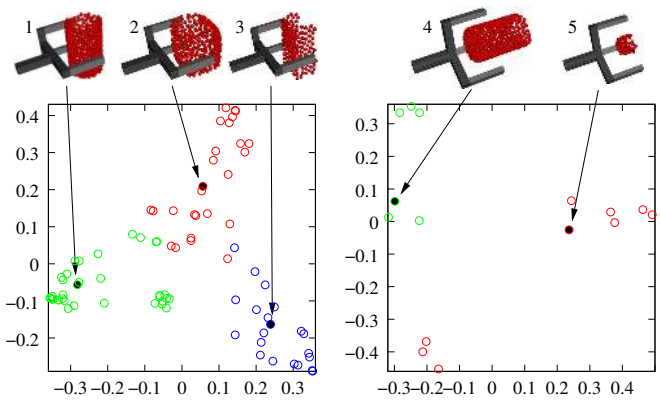
The objects used for training the robot are presented in Figure 1a. In general, our method assumes that the robot knows the *training* objects, which means that it has full 3D shape models of all of them. (We naturally do not make the same assumption for the new objects that the robot subsequently grasps.)

Our robot is composed of a Schunk three-finger hand mounted on a Kuka arm, and a Kinect camera that is fixed on a structure 1.5m away from the robot.

We demonstrated thirty-two grasps to the robot. We proceeded by first instructing the robot to set its hand to either a parallel or cylindric preshape (see Figure 2). We then placed one of the objects in the hand, and instructed the robot to close the hand and take a 3D snapshot of the grasp with the Kinect camera. For each snapshot the model of the object was aligned to the correct pose, using the pose estimation method described above [16]. This process resulted in a representation of each grasp in terms of the gripper pose, obtained from forward kinematics, the gripper preshape before the grasp, and a full 3D shape model of the object being grasped (see Figure 3).

From the 32 grasps, we generated 96 part candidates, following the procedure of Section III-B. We then computed pairwise similarities between the 81 parallel-preshape candidates and between the 15 cylindric-preshape candidates, and we applied the clustering algorithm of Section III-D to these data. Computing part similarities took 25 minutes with a single-threaded C++ implementation on an Intel Core i7 processor. Clustering the data took about a second.

We inspected the results generated with different numbers of clusters. For the parallel-preshape data, the most sensible result was obtained with three clusters. The first cluster contained side grasps on the cylindric objects of Figure 1a. The second cluster contained grasps on the white and pink objects, whose shape lies between a cylinder and a box. The third cluster contained grasps on the box-like objects. For the cylindric preshape data, two clusters best captured the nature of the data, separating it into top grasps onto the two cylindric objects, and grasps applied to the cap of the blue



(a) Parallel-preshape candidates (b) Cylindric-preshape candidates

Fig. 4: Two-dimensional approximation of the geometric configuration of part candidates. The distance between two points in these plots approximates the dissimilarity of the two corresponding parts. Colors correspond to the labels that result from the clustering process performed on the pairwise similarities. The black dots are the cluster centers from which the prototype dictionary is constructed.

Success	Firm	39	(70%)
	Loose	7	(13%)
	Total	**46**	**(84%)**
Failure		**9**	**(16%)**

TABLE I: Grasp success rates. Loose successful grasps correspond to grasps for which the object moved during lift-up. See text for details.

and pink bottles. We note that although we determined the numbers of clusters by inspection, for larger datasets a BIC-like criterion that computes an optimal number of clusters from the data could be used instead [35].

Figure 4 shows a 2D approximation of the space induced by the shape dissimilarities between parts. This approximation is obtained by keeping only the two principal components of the data. We see that the clusters look approximately Gaussian. In order to compress the data into a compact dictionary of prototypes, we model each cluster as a Gaussian and use its center as the prototype. The dictionary of prototypes is shown in the topside of Figure 4.

B. Grasping Novel Objects

To grasp an object, the robot took a 3D snapshot with the Kinect camera, and aligned the five prototypes of Figure 4 to it. Aligning the five prototypes by solving Eq. 7 five times took 90 seconds on average on an Intel Core i7 CPU. The grasp parameters of the prototype that best fitted the snapshot were used to execute the grasp. The grasp was executed by bringing the manipulator to the correct pose and preshape, then closing the fingers to apply a fixed force onto the object.

Results are presented in Table I. The robot planned 55 grasps on the object of Figure 1d. Our main criterion of success is whether the robot manages to apply its fingers on the object in a way that blocks the object between the fingers. Out of the 55 grasps, 46 were successful. We note that for 7 of these, the force applied to the hand by the object while the robot was lifting it up was strong enough to make the object slip in the hand (5 grasps) or even fall (2 grasps). We emphasize that the goal of this work is to plan grasps that *locally* match an object's shape. Taking inertial parameters into account to favor grasps applied, e.g., near the center of mass of an object, is a problem that we plan to address in future work.

Figure 5 shows examples of grasps. Figure 5a shows a typical grasp, where a cylindric prototype is fitted to a bottle. In Figure 5b and Figure 5c, the shapes of the dustpan and kettle and are unlike any of the objects of the training set. The grasps generated by the closest matches did work, which reveals the robustness of the method. A more elegant solution to these two cases would however come from realizing that the two objects present cylinder-like parts, albeit of different radii. The prototype associated to a cylinder could then be scaled to the proper size, and the corresponding grasp would be better adapted to the object. Such a behavior is however beyond the scope of this paper and we leave its discussion to future work.

Figure 5d shows more examples of successful grasps. In the third image, the electric cable prevented the robot from planning a side grasp on the handle. Instead, the robot grasped the work lamp by its top by transferring a bottle-top grasp (see for instance Figure 1b). The same situation occurs in the last image of Figure 5d, where the lampshade was out of reach, and the lamp pole doesn't match any prototype. Again, the robot adapted a cylindric bottle-top grasp to the top of the lamp (we note that the fingers are in a cylindric preshape in the fifth and sixth images). Figures 5e to 5g show examples of failed grasps. In Figure 5e, the dustpan is laying upside-down on the table. This is a difficult configuration, where side grasps would make one of the fingers pass close to the table. Given that the robot rejects all grasps that risk colliding with the table, its only plan hallucinates a cylinder above the dustpan. In the next image (Figure 5f), the robot had correctly aligned a cylindric part to the hose connector. Unfortunately, one of the fingers collided with the back side of the left-ward connector while the hand was closing, making the object fall. This kind of problem would be hard to solve with only one view of the object. Addressing it would require the robot to turn its camera around to view all sides of objects. In Figure 5g, the robot matches the bottom part of a cylindric prototype to the brush container. However, the container is shorter than the prototype, and the fingers miss the object. This is a problem that we can address, in two different ways. The first solution would be to increase the training dataset to include objects of the size of the brush container. The second solution would be to force the robot to match all of the prototype surfaces *that should be visible from the camera viewpoint* to surfaces obtained from the Kinect. This way, the robot would make sure that the prototype completely fits the side of the object that faces the camera.

We note that our method is applicable to environments with multiple objects (see Figure 5a). The only differences with an environment that contains only one object are that (1) there may be occlusions that make grasping some of the objects more difficult, and (2) when two small objects stand next to each other, our method may fit a bigger prototype to the surface formed by both objects. We leave a longer discussion of these issues to future work.

V. Conclusion

We presented a robotic agent that is capable of transferring grasping knowledge across partly similar objects. Our method relies on the identification of recurring parts within sets of part candidates generated by extracting object surface segments in the vicinity of grasps demonstrated by a human. We devised a similarity measure that allows the agent to identify parts that have a similar shape, while being robust to a certain amount of variation in the absolute pose of each part. In a real-world experiment, our agent learned a dictionary of prototypical parts from parallel-finger and cylindric grasps demonstrated on eight different objects. The dictionary allowed the robot to devise workable strategies for real-world novel objects whose global shape differs from that of any of the training objects.

As the dictionary of parts is only formed from cluster centers, it is allowed to be orders of magnitude smaller than the set of grasp examples initially provided to the agent. A grasp example only "votes" for the potential inclusion of a prototype into the dictionary, which provides us with a means of controlling the size of the dictionary in order to keep the computational cost of planning a grasp onto a novel object reasonably low. Finally, not only the shape, but also the spatial extent (or size) of the parts that form the dictionary depend on the available grasp data. Prototypical parts are selected based on their recurrence across experienced grasps, which leads to the identification of parts that strongly predict grasp applicability.

References

[1] J. Aleotti and S. Caselli. Part-based robot grasp planning from human demonstration. In *IEEE International Conference on Robotics and Automation*, 2011.

[2] P. Allen, A. Miller, P. Oh, and B. Leibowitz. Integration of vision, force and tactile sensing for grasping. *Int. J. Intelligent Machines*, 4(1):129–149, 1999.

[3] C. Andrieu, N. de Freitas, A. Doucet, and M. I. Jordan. An introduction to MCMC for machine learning. *Machine Learning*, 50(1):5–43, January 2003.

[4] C. Bard and J. Troccaz. Automatic preshaping for a dextrous hand from a simple description of objects. In *IEEE International Workshop on Intelligent Robots and Systems*, pages 865–872. IEEE, 1990.

[5] P. Besl and N. McKay. A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(2):239–256, 1992.

[6] A. Bicchi and V. Kumar. Robotic grasping and contact: a review. In *IEEE International Conference on Robotics and Automation*, 2000.

[7] J. Bohg, M. Johnson-Roberson, B. León, J. Felip, X. Gratal, N. Bergstrom, D. Kragic, and A. Morales. Mind the gap – robotic grasping under incomplete observation. In *IEEE International Conference on Robotics and Automation*, pages 686–693, 2011.

[8] M. C. Burl, M. Weber, and P. Perona. A probabilistic approach to object recognition using local photometry and global geometry. In *European Conference on Computer Vision*, pages 628–641, 1998.

[9] R. Caflisch. Monte carlo and quasi-monte carlo methods. *Acta Numerica*, 7:1–49, 1998.

[10] J. Coelho, J. Piater, and R. Grupen. Developing haptic and visual perceptual categories for reaching and grasping with a humanoid robot. In *Robotics and Autonomous Systems*, volume 37, pages 7–8, 2000.

[11] C. de Granville, J. Southerland, and A. H. Fagg. Learning grasp affordances through human demonstration. In *IEEE International Conference on Development and Learning*, 2006.

[12] R. Detry. Learning a dictionary of prototypical grasp-predicting parts from grasping experience – dissimilarity measure implementation. URL: http://renaud-detry.net/p/icra2013.

[13] R. Detry. *Learning of Multi-Dimensional, Multi-Modal Features for Robotic Grasping*. PhD thesis, University of Liège, 2010.

[14] R. Detry, C. H. Ek, M. Madry, J. Piater, and D. Kragic. Generalizing grasps across partly similar objects. In *IEEE International Conference on Robotics and Automation*, 2012.

[15] R. Detry, D. Kraft, O. Kroemer, L. Bodenhagen, J. Peters, N. Krüger, and J. Piater. Learning grasp affordance densities. *Paladyn. Journal of Behavioral Robotics*, 2(1):1–17, 2011.

[16] R. Detry and J. Piater. Continuous surface-point distributions for 3D object pose estimation and recognition. In *Asian Conference on Computer Vision*, pages 572–585, 2010.

[17] R. Detry and J. Piater. Grasp generalization via predictive parts. In *Austrian Robotics Workshop*, 2011.

[18] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient matching of pictorial structures. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2066–, 2000.

[19] M. Fischler and R. Elschlager. The representation and matching of pictorial structures. *IEEE Transactions on Computers*, 100(1):67–92, 1973.

[20] C. Goldfeder, M. Ciocarlie, H. Dang, and P. Allen. The Columbia grasp database. In *IEEE International Conference on Robotics and Automation*, 2009.

[21] A. Herzog, P. Pastor, M. Kalakrishnan, L. Righetti, T. Asfour, and S. Schaal. Template-based learning of grasp selection. In *The PR2 Workshop (Workshop at IROS'11)*, 2011.

[22] K. Hsiao, S. Chitta, M. Ciocarlie, and E. Jones. Contact-reactive grasping of objects with partial shape information. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1228–1235, 2010.

[23] I. Kamon, T. Flash, and S. Edelman. Learning to grasp using visual information. In *IEEE International Conference on Robotics and Automation*, volume 3, pages 2470–2476, 1996.

[24] J. Kim. M. eng. Master's thesis, Massachusetts Institute of Technology, 2007.

[25] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[26] E. Klingbeil, D. Rao, B. Carpenter, V. Ganapathi, A. Ng, and O. Khatib. Grasping with application to an autonomous checkout robot. In *IEEE International Conference on Robotics and Automation*, 2011.

[27] D. Kragic, A. T. Miller, and P. K. Allen. Real-time tracking meets online grasp planning. In *IEEE International Conference on Robotics and Automation*, pages 2460–2465, 2001.

[28] O. Kroemer, E. Ugur, E. Oztop, and J. Peters. A kernel-based approach to direct action perception. In *IEEE International Conference on Robotics and Automation*, 2012.

[29] A. T. Miller, S. Knoop, H. Christensen, and P. K. Allen. Automatic grasp planning using shape primitives. In *IEEE International Conference on Robotics and Automation*, volume 2, pages 1824–1829, 2003.

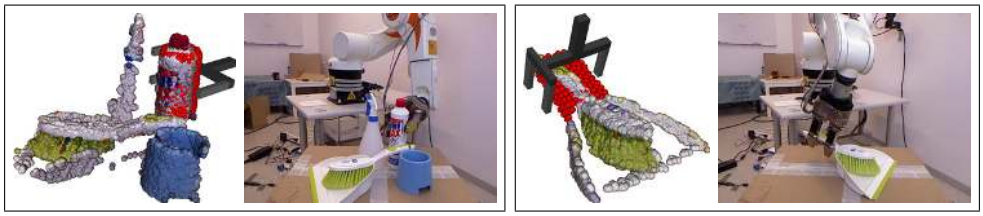
[30] L. Montesano and M. Lopes. Learning grasping affordances from local visual descriptors. In *IEEE International Conference on Development and Learning*, 2009.

[31] A. Morales, E. Chinellato, A. H. Fagg, and A. P. del Pobil. Using experience for assessing grasp reliability. *International Journal of Humanoid Robotics*, 1(4):671–691, 2004.

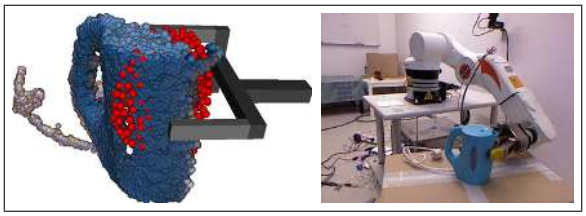
[32] M. Popović, D. Kraft, L. Bodenhagen, E. Başeski, N. Pugeault, D. Kragic, T. Asfour, and N. Krüger. A strategy for grasping unknown objects based on co-planarity and colour information. *Robotics and Autonomous Systems*, 2010.

[33] A. Saxena, J. Driemeyer, and A. Y. Ng. Robotic Grasping of Novel Objects using Vision. *International Journal of Robotics Research*, 27(2):157, 2008.

[34] B. Schölkopf and A. Smola. Kernel principal component analysis. *Artificial Neural Networks—ICANN'97*, 1997.

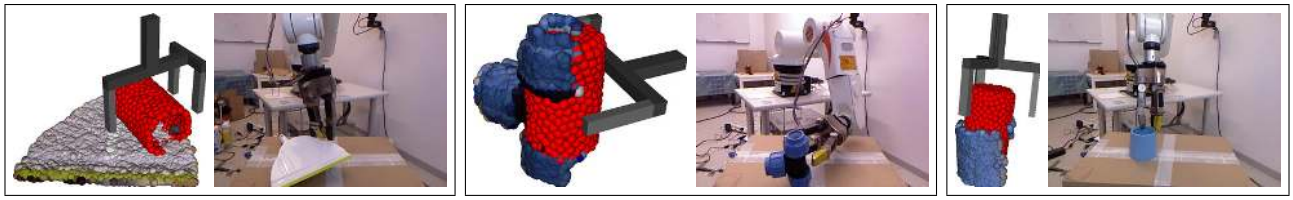


(a) Success (b) Success



(c) Success

(d) Success



(e) Failed grasp (f) Failed grasp (g) Failed grasp

Fig. 5: Grasping the objects shown in Figure 1d using the prototypes from Figure 4. For each object, we show the best-fitting prototype aligned to a 3D snapshot of the object, and the grasp performed by the robot. The robot images are produced by the Kinect camera with which we took 3D snapshots.

[35] G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464, 1978.

[36] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[37] K. B. Shimoga. Robot grasp synthesis algorithms: A survey. *The International Journal of Robotics Research*, 15(3):230, 1996.

[38] J. D. Sweeney and R. Grupen. A model of shared grasp affordances from demonstration. In *International Conference on Humanoid Robots*, 2007.

[39] S. Thrun and B. Wegbreit. Shape from symmetry. In *IEEE International Conference on Computer Vision*, volume 2, pages 1824–1831, 2005.

[40] L. E. Zhang, M. Ciocarlie, and K. Hsiao. Grasp evaluation with graspable feature matching. In *RSS Workshop on Mobile Manipulation: Learning to Manipulate*, 2011.