2011 IEEE International Conference on Robotics and Automation
Shanghai International Conference Center
May 9-13, 2011, Shanghai, China

Part-Based Robot Grasp Planning from Human Demonstration

Jacopo Aleotti, Stefano Caselli
RIMLab - Robotics and Intelligent Machines Laboratory
Dipartimento di Ingegneria dell'Informazione, University of Parma, Italy
E-mail {aleotti,caselli}@ce.unipr.it

Abstract— **In this work we introduce a novel approach for robot grasp planning. The proposed method combines the benefits of programming by human demonstration for teaching appropriate grasps with those of automatic 3D shape segmentation for object recognition and semantic modeling. The work is motivated by important studies on human manipulation suggesting that when an object is perceived for grasping it is first parsed in its constituent parts. Following these findings we present a manipulation planning system capable of grasping objects by their parts which learns new tasks from human demonstration. The central advantage over previous approaches is the use of a topological method for shape segmentation enabling both object retrieval and part-based grasp planning according to the affordances of an object. Manipulation tasks are demonstrated in a virtual reality environment using a data glove. After the learning phase, each task is planned and executed in a robot environment that is able to generalize to similar, but previously unknown, objects.**

I. INTRODUCTION

The complexity of grasp planning derives from the need of modeling and recognizing the objects, as well as from the dimension of the space of possible grasps, which is too large to be searched exhaustively. A fundamental issue which, however, is not considered in most approaches, is that manipulation planning should be task-oriented. Task-oriented grasp planning refers to the problem of finding a grasp on an object which is suitable for a particular action or affordance. An affordance is a property of an object that provides an indication of how to interact with that object. The idea of task-oriented grasping is strictly related to neuro-psychology studies that have evidenced the influence of shape decomposition for human perception of objects. Hoffman et al. [13] suggested that shapes are perceived as collections of simple components. Biederman [6] elaborated the Recognition-By-Components theory (RBC), according to which the preferred mode of human object recognition is a bottom-up process where objects are separated into interrelated geometries. Gibson's theory of affordances [8] states that objects are perceived not only in terms of shapes and spatial relationships but also in terms of possibilities for action.

This paper presents a novel approach for manipulation planning from human demonstration where objects are grasped by their parts. We adopt a method for shape segmentation, based on Reeb graphs, that generates a topological representation of the objects. We show the benefits of using such representation as it enables object recognition, part-based planning for different but similarly shaped objects and detection of the grasped parts in the user demonstration phase of the task (performed in virtual reality). Algorithms for topological decomposition are indeed capable of identifying semantic object's parts that are natural candidates for being grasped (for example the handle of a bag). Hence, shape segmentation applied to robot grasping provides meaningful information about grasp affordances, which are shared among similar objects. In summary, the most important contribution of this paper compared to previous works is that we consider a more general semantic representation of a grasping action which is specified not only by the object but also by the target part of the object that should be grasped by the robot hand in order to perform a task. A further advantage is that a grasping by parts approach narrows down the search space of possible grasps.

Figure 1 shows the architecture of the part-based manipulation planning system by human demonstration. The proposed system consists of an interactive virtual reality environment used for task demonstration and a robot environment used for planning and testing the learned tasks. The virtual reality environment comprises a data glove and supports virtual grasping. The two environments can be set up with different but similarly shaped objects thus enabling the system to plan manipulation tasks at a more general level of abstraction. Each object is automatically segmented and recognized as a member of a specific class (by means of a 3D shape retrieval algorithm) and its parts are automatically annotated according to an ontological description of the class. The user can provide multiple demonstrations of the same task. We focus on manipulation tasks that consist of a sequence of actions of grasping and placing objects. After the demonstration phase the system automatically learns a task precedence graph (TPG) which is augmented with information about the parts of the objects that should be grasped. In the final stage a grasping by parts manipulation planner is invoked in the robot environment which comprises a robot arm with a three finger hand. The manipulation planner selects candidate grasps from an object grasp map that contains sampled stable grasps for each object part of interest.

The paper is organized as follows. Section II reviews previous works related to robot grasp synthesis and manipulation planning. Section III presents the method for topological object segmentation and recognition while section IV provides a detailed description of the proposed approach for part-based grasp planning from human demonstration.

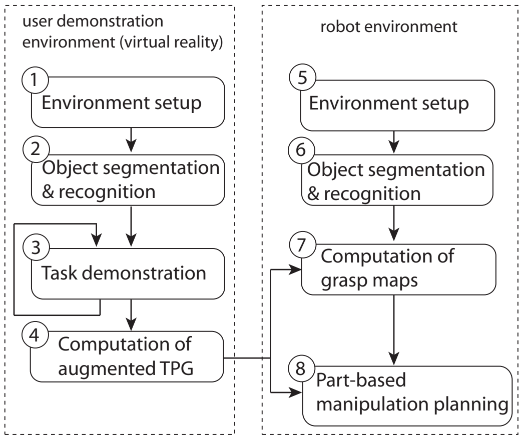

Fig. 1. Architecture of the part-based manipulation planning system by human demonstration.

Experiments in a simulated robot environment are reported in section V.

II. STATE OF THE ART

Topological shape decomposition has not been considered in related works on robot grasping. Indeed we investigate a novel approach for semantic robot grasping by demonstration based on shape segmentation. Hsiao et al. [14] proposed a system for learning whole body grasps from imitation. Grasp planning is performed by adapting demonstrated grasps to the target object via morphing transformations. Objects are assumed to be composed of up to three primitive shapes (boxes, cylinders and spheres) with aligned axes of symmetry. Kyota et al. [17] presented a method for learning force-closure grasps based on neural networks. An object is considered as a whole entity. Candidate grasping portions are selected according to their cylindrical likeness. Ekvall et al. [7] adopted an approach for learning and evaluating the grasp approach vector by demonstration. Grasp demonstrations are used to find candidate hand preshapes with respect to the objects. Objects are manually segmented into elementary shape primitives before planning. Sahbani et al. [22] investigated a method for learning to grasp unknown objects by imitation and part decomposition with superquadrics. A single natural grasping component is predicted for each target object. The method permits to find a grasp in accordance with the object usage but does not perform object recognition.

In [10], [15] 3D object segmentation has been applied to robot grasping as a way to reduce the planning time without extracting semantic information from the objects. Such methods were based on superquadric decomposition or minimum volume bounding boxes and were not aimed at generating grasps across similar objects. The use of superquadrics or bounding boxes has some limitations for objects that can not be easily segmented into convex shapes, thus leading to a coarse approximation. Moreover, superquadrics provide a model fitting approach while Reeb graphs generate an actual topological segmentation of the object.

In [11] a large database of 3D shapes has been generated and a grasp planner for novel objects is proposed using pre-computed grasps. The method is not based on grasping by parts. Geometric similarity between the objects is computed by using shape descriptors. Further research works have focused on learning grasp affordances without explicitly detecting object parts for semantic grasping. Montesano et al. [19] introduced a bayesian algorithm for learning visual grasp descriptors from local features through robot self-experimentation. The method generates a probabilistic representation of the object affordances taking into account the morphology of the robot hand. Li et al. [18] proposed a method for matching the hand shape to the local shape of the object. The algorithm synthesizes human-like enveloping grasps. Sweeney et al. [25] proposed a vision-based system that identifies shared grasp affordances from teleoperation. Affordances are represented as statistical distributions that consider the visual appearance of the object as well as the hand position and orientation during grasping.

Grasp planners are usually integrated in more complex manipulation planning systems. Hereafter we briefly review a selection of research works focused on manipulation planning which, however, do not address the problem of part-based grasping. In [21] a human-inspired system is presented for manipulation planning of real objects. In [24] a manipulation planner is described for simulated pick and place operations that handles continuous placements. Other advanced applications deal with mobile manipulation tasks [27], [16]. Many authors have also focused on manipulation planning problems in complex cluttered scenes by modeling the reachable workspace of the robot [28], [2], [12], [20], [9], [3]. In particular, Morales et al. [20] presented a method that combines visual object recognition and grasping. Berenson et al. [2] proposed a general framework for manipulation planning with a humanoid robot.

III. 3D OBJECT SEGMENTATION AND RECOGNITION

The proposed solution for shape segmentation and recognition is based upon the Reeb graph, a data structure that depicts the skeleton of a geometric model. A Reeb graph [5], [23], [4] represents the topology of a surface described by the evolution of the level-sets of a scalar function over the mesh. More formally, given a surface S and a real, continuous function $f : S \rightarrow R$ defined on it, the Reeb graph of S with respect to the mapping function f is the quotient space of f in $S \times R$ by the equivalence relation $(v_1, f(v_1)) \sim (v_2, f(v_2))$ which holds if and only if $f(v_1) = f(v_2)$ and if the two points v_1 and v_2 are in the same connected component of $f^{-1}(f(v_1))$.

Given a 3D mesh, its Reeb graph is defined by the choice of the scalar function f and by a fixed number of quantization levels. The choice of f affects the topology of the resulting Reeb graph and the computational time (the theoretical lower bound is $O(n \log n)$, where n is the number of triangles). In this work we have adopted the integral geodesic distance as it is a scalar function that ensures invariance to rotation. The integral geodesic distance is defined as $f(u) = \int_{v \in S} g(u, v) dS$ where $g(u, v)$ is the geodesic distance between the two points

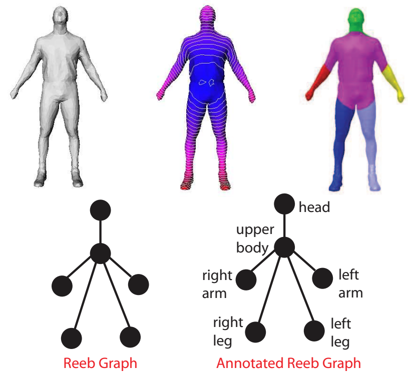

Fig. 2. Reeb graph of 3D human model. In the top row (from left to right) the original mesh (25060 triangles), the level-sets of the integral geodesic function and the segmented shape (each part has a different color). In the bottom row the Reeb graph and the annotated Reeb graph.

(u, v) on S. Invariance to rotation is a valuable property in the context of robot manipulation where objects may have arbitrary poses. The complexity of the integral geodesic distance is $O(n^2 \log n)$. For complex 3D shapes like the ones in the chosen database (figure 3) the computation of the Reeb graph takes some minutes (about 5 minutes for a model with $20K$ vertices). To make the computation more tractable the original 3D shapes in figure 3 have been simplified (by a factor of about 20%) using a quadric edge collapse decimation procedure that preserves the topology of the shapes (the computational time after re-meshing for the shapes in figure 3 ranges from a minimum of $11.8s$ for a 2011 vertices model to a maximum of $243s$ for a 8653 vertices model). After the computation of the Reeb graph, to avoid over-decomposition of the mesh a post-processing filter [4] is applied which iteratively merges segmented regions based on adjacency and cardinality of level sets.

Figure 2 shows the Reeb graph computed for a 3D mesh of a human model and the resulting object decomposed into six parts (upper body, lower body, two arms and two legs). The figure also shows the annotated Reeb graph with tags that are automatically associated to the object's parts, after recognition, given the ontological description of the object. The annotated Reeb graph is used as input for both the task demonstration environment and the manipulation planner. The automatic annotation phase is supervised by the user to resolve potential ambiguities arising from internal object symmetries.

A. Object recognition

After segmentation, an algorithm for automatic object recognition is executed to classify each object to be manipulated into one of the classes of the database (in both the demonstration and robot environments). Reeb graphs are represented by a symmetric adjacency matrix A (element $A_{i,j} = 1$ if edge (i, j) belongs to the graph). A prototype Reeb graph has been predefined, with annotated parts, for each class of objects in the database. The object to be grasped is classified as belonging to the class of the closest prototype. We have adopted the efficient PATH algorithm [29] that supports approximate graph matching between graphs of different sizes. The distance between the Reeb graph of the object A and the adjacency matrix of a prototype Reeb graph P_i is computed by minimizing the Frobenius matrix norm $\|A - \hat{P}_i\|_F^2$ over the set of permutation matrices, where $\hat{P}_i$ is a permutation of the adjacency matrix P_i.

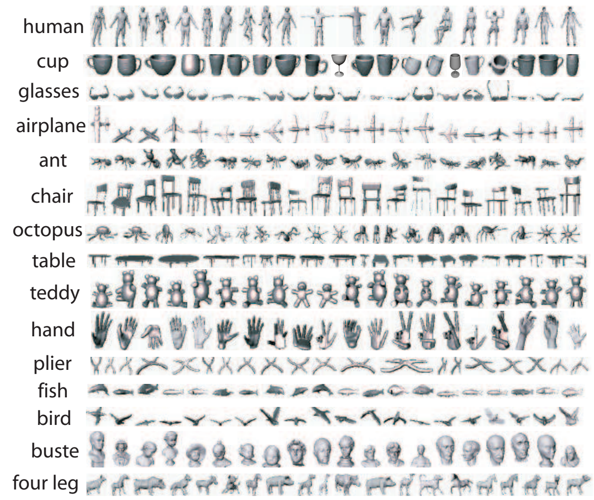

Fig. 3. The database of objects used in this work. The database is taken from the 2007 shape retrieval contest (SHREC, http://www.aimatshape.net/).

Figure 3 shows the 3D models used in the experiments. The dataset consists of 15 classes including daily life objects. Each class contains 20 similar models. The models are water-tight, i.e. there are no holes in the surface of the object. The dataset has been taken from the 2007 shape retrieval contest (SHREC). We evaluated the object recognition algorithm for all the objects in the dataset. The percentage of recognition is 88.4%. It must be noted that the selected dataset is challenging because it contains ambiguous shapes, i.e. there are classes of objects with the same Reeb graphs (e.g. plier-table). These shapes can not be distinguished by using a purely graph-based approach. More advanced classification algorithms [26], that are beyond the scope of this work, can achieve better results by extending the standard Reeb graph with geometrical attributes.

IV. MANIPULATION PLANNING BY DEMONSTRATION

This section describes the main components of the manipulation planning system, that are illustrated in figure 1. Namely the interactive environment for task demonstration, the computation phase of the objects grasp maps and the part-based robot manipulation planner.

A. Task demonstration

The manipulation planning system presented in this paper is grounded on robot programming by demonstration. Task demonstration is performed in an interactive 3D virtual environment. The virtual environment supports physics-based animation for realistic simulation of rigid bodies with arbitrary shape (based on the Open Dynamics Engine). User interaction is allowed by driving in real-time a virtual anthropomorphic hand (with 22 dofs) while wearing a CyberTouch glove and a motion tracker (FasTrak, by Polhemus Inc.). The devices enable full hand pose estimation and tracking. A virtual grasping algorithm has been developed for object picking and for computing the part of the object being grasped. We focus on grasps where the hand touches a single part of the object, which is the case for most common grasps. A standard grasp quality metric based on the grasp wrench space is adopted to plan force-closure grasps.

Given an object o the algorithm used for detection of the grasped part $k^* \in K$ (K being the set of parts of o) takes advantage of the shape segmentation phase and is described as follows. Shape segmentation provides a direct mapping between each vertex on the mesh M and which part of the object the vertex belongs to. Let $p : M \rightarrow K$ the function that returns for each vertex its part. Let also be X a large set of uniformly sampled vertices in M (covering all the parts of the object). Let C be the set of contact points of a grasp g between the virtual hand and the object o, provided by collision detection. Since demonstration is performed by a human demonstrator the contact points of the virtual hand may not all belong to the same part of the shape. We define the set of contact points Z_k belonging to the part k as:

$$Z_k = \left\{ c \in C |\ x^* = \operatorname*{argmin}_{x \in X} \|c - x\|, p(x^*) = k \right\} \quad (1)$$

that is, a contact point c belongs to Z_k if its closest point $x^* \in X$ belongs to k. Then, the grasped part k^* is given by the set Z_k with the largest cardinality:

$$k^* = \operatorname*{argmax}_{k \in K} Card(Z_k) \quad (2)$$

i.e. the grasped part is the one that contains most of the contact points. Images of virtual grasping actions are reported in figures 5,7.

Hereafter, we describe the approach for task learning from user demonstration. We have extended a standard method for learning discrete actions to the problem of grasping by parts. The method is based on the incremental learning of a task precedence graph (TPG) from multiple demonstrations. A TPG is a directed, acyclic graph $G = \{V, E\}$ where V is the finite set of feasible elementary actions $V = \{t_1, t_2, \ldots, t_n\}$ and $E \subset V \times V$ is a set of arcs of precedence relations (t_a, t_b) stating that action t_a must be completed before the execution of action t_b can start. A TPG can be learned incrementally from a sequence of m demonstrations of the same task. After the first demonstration $D_1 = \{t_{1_1}, t_{1_2}, \ldots, t_{1_n}\}$ the set of constraints $E_1 = E^{D_1} = \{(t_{1_i}, t_{1_j}) | i < j\}$ is initialized with all the precedence relations contained in D_1. For each additional demonstration D_h that solve the same task the TPG is updated by keeping only the precedence relations introduced by the latest demonstration that comply with the past knowledge about the task ($E_{h+1} = E_h \cap E^{D_{h+1}}$). New demonstrations, that show alternative solutions for the task, have the effect of removing previous constraints in the TPG. The elementary actions that are recognizable by the system are similar to the ones described in [1], [30] which comprise pick and place operations, tool handling and peg in hole tasks. Elementary actions also support inter-object geometrical relations such as placing an object on top of, or to the right of another object. However, in previous approaches [1], [30] grasping actions were specified only generically. This work is aimed at learning both elementary actions and grasped parts. For the purpose of learning which part to grasp we augment the definition of elementary actions by including information about the part of the object to be grasped. Augmented elementary actions are defined as $\tilde{t} = \{t, k\}$ where t is an elementary action like the ones described above and k is the part of the object to be grasped. This definition provides a more detailed description of the learned grasping actions that is exploited by the manipulation planner in the robot environment.

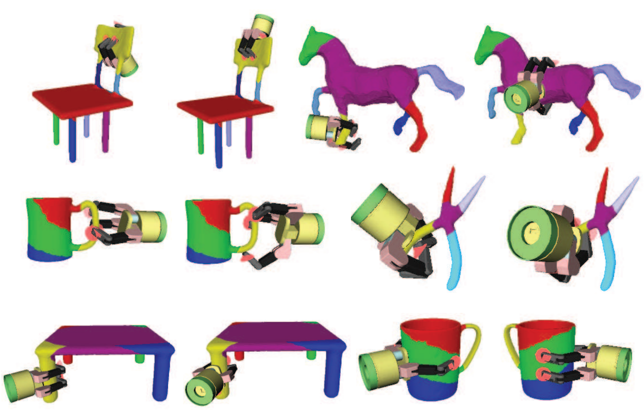

Fig. 4. Example grasps extracted from the grasp maps of the objects used in the experiments in section V.

Algorithm 1 computation of object grasp map

Require: A segmented object o
for all k parts of interest of o **do**
  Compute the centroid of the sub-mesh representing k
  Compute the three principal axes of inertia of k
  for all trial grasps **do**
    Sample a random start point near the principal axis
    Sample a random approach vector
    Sample a random pregrasp config. of the robot hand
    Approach target part k along the normal of the palm
    Close the fingers until collision is detected
    Evaluate force-closure of the generated grasp g
    if g is force-closure **then**
      update grasp map $[\Omega_o^k] = [\Omega_o^k | g]$
    end if
  end for
end for

B. Computation of grasp maps

The robot manipulation planner synthesizes grasping actions from the learned human demonstrations. Candidate part-based grasps are selected from a robot grasp map that is computed for each object in the robot environment. A grasp map Ω_o for object o is a dataset that contains a large number

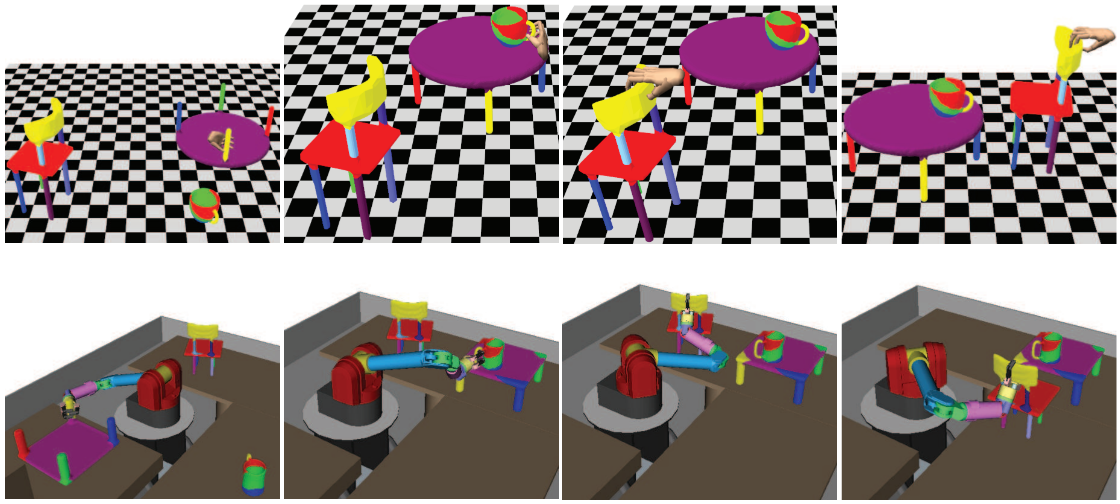

Fig. 5. Part-based manipulation planning: experiment 1. Demonstration phase (top row) and robot simulation phase (bottom row).

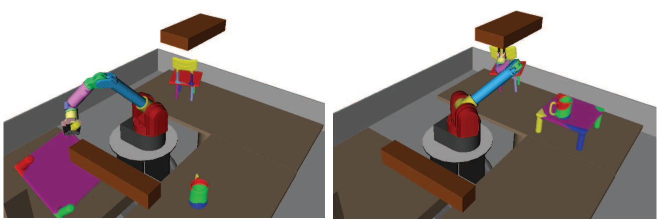

Fig. 6. Part-based manipulation planning: experiment 1 with additional static obstacles.

of force-closure grasps for each object part of interest. In particular, the grasp map is organized as a block matrix as $\Omega_o = [\ \Omega_o^1 \mid \Omega_o^2 \mid \cdots \mid \Omega_o^{|K|}\ ]$, where Ω_o^k is a block that contains as many column vectors as there are force-closure grasps for the kth part of object o. Each column vector $g \in \mathbb{R}^{12}$ uniquely identifies a grasp by specifying the pregrasp end-effector pose (6dof), the four joint angles of the robot hand (we have performed experiments with the Barrett hand, which has three fingers and four degrees of freedom), the initial roll angle of the wrist, and an offset that specifies the final distance of the palm from the object. The planner has been designed upon the OpenRAVE engine. Figure 4 shows some of the grasps extracted from the grasp maps of the objects used in the experiments reported in section V.

The procedure for sampling the part-based grasp map is reported in Algorithm 1. For each part k of interest the centroid of the mesh of vertices of that part is initially computed as well as the three principal axes of inertia using Principal Component Analysis. Then, the algorithm samples a pre-grasp configuration of the end-effector within a certain distance from the centroid of k (including random joint angles). The orientation of the robot hand is chosen so as to point approximately towards the principal axis. The simulated hand is then translated towards the object along the approach vector until a random small offset is reached from the object (often set to zero). In the final stage fingers are closed until contact is detected with the object and force-closure is evaluated. If the resulting grasp is force-closure it is included in the grasp map. To improve efficiency, given a task learned by demonstration, the system is programmed to compute grasp maps only for the object parts of interest (i.e. all the parts to be grasped and their equivalent parts, see section IV-C). The proposed approach for part-based grasp synthesis has the further benefit of making the planning phase efficient as it naturally narrows down the search space of possible grasps in the neighborhood of the object parts, where it is more likely to generate force-closed grasps.

C. Part-based manipulation planning

The part-based manipulation planner is invoked after the demonstration phase. Given a learned task precedence graph and the objects grasp maps, the manipulation planner is responsible for finding a sequence of augmented elementary actions that comply with the TPG and, for each action, a collision-free trajectory for the robot. The manipulation planner is based on the OpenRAVE engine. OpenRAVE uses a Bi-directional Rapidly-exploring Random Tree (RRT) algorithm for motion planning. Experiments have been performed in a simulated robot environment with a WAM arm and a Barrett Hand. The simulation environment comprises the robot manipulator, the graspable objects and static rigid obstacles. For each augmented grasping action $\tilde{t} = \{t, k\}$ of an object o, candidate grasps are chosen among the grasp block Ω_o^k of the grasp map Ω_o. We recall that a grasp block contains all the computed grasps of a part of the object. If the grasp planning phase fails for all the trial grasps contained in Ω_o^k then the manipulation planner tries to extend the search for feasible grasps to equivalent parts of the object. Indeed, in the object annotation phase, the user is able to specify equivalence between parts (e.g. all the legs of a table are equivalent). The class of equivalence of part k is defined as $[k] = \{\chi \in K \mid k \sim \chi\}$, where χ are all the parts in K that are equivalent to k. If the extended grasp planning phase

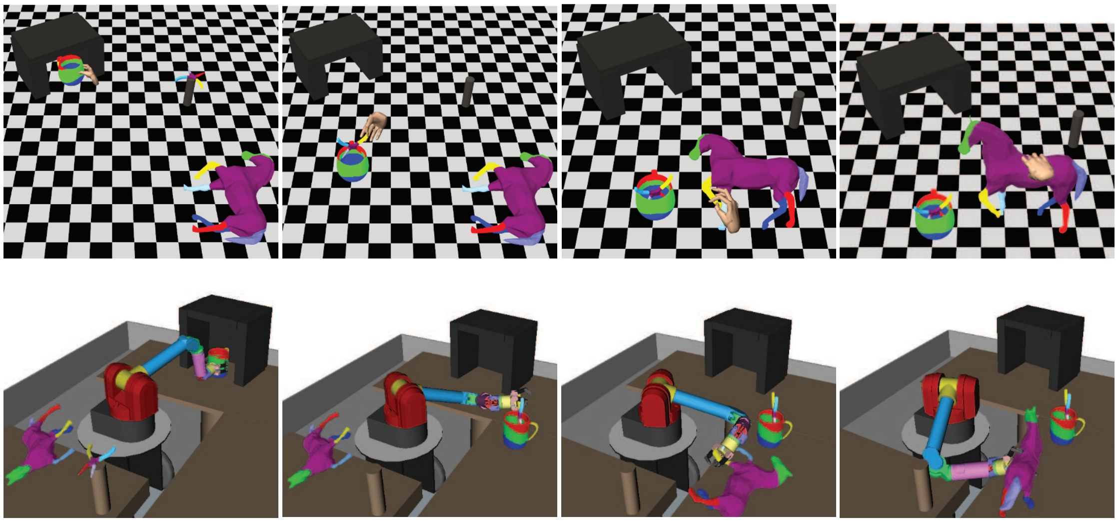

Fig. 7. Part-based manipulation planning: experiment 2 (see the accompanying video). Demonstration phase (top row) and robot simulation phase (bottom row).

does not find a feasible solution the manipulation planner returns a failure and explores another compliant sequence of actions.

V. EXPERIMENTAL EVALUATION

Experiments have been performed in order to assess the capabilities of the proposed method for part-based manipulation planning. In particular, we show that the simulation phase generalizes grasping tasks to objects that are different, but similarly shaped, from those used in the demonstration phase. The grasp map of each object contains about twenty force-closure grasps per part. The first experiment comprises an environment with three objects (a cup, a table and a chair) as well as three recognizable augmented actions: picking the table from a leg, turning it upside-down and placing it on the ground (action A_1); grasping the cup from the handle and laying it on the table (action B_1); grasping the back of the chair and putting the chair on the ground (action C_1). The chair and the table are side-by-side. The user provides two demonstrations of the task. In the first demonstration (reported in figure 5) the user performs the sequence of actions $A_1B_1C_1$, while in the second demonstration the sequence is $C_1A_1B_1$. The system learns that the precedence relation (A_1, B_1) holds, i.e. the cup should be put on top of the table only after turning the table, while the chair can be moved regardless of them. Figure 5 also reports images of the planned collision-free trajectories from the robot simulation phase. Figure 6 shows the same manipulation experiment planned in an environment with additional static obstacles. Table I reports the average time of each phase (Intel @$2.16Ghz$) of the part-based approach (first experiment). The most time consuming phase is object segmentation ($120s$) which, however, could be reduced by undersampling the original meshes. The grasp maps contain 20 grasps for each part. We have also compared the performance of the part-based approach with the standard algorithm provided by OpenRAVE which generates the grasp maps by uniformly sampling the bounding box of the object. The standard algorithm does not guarantee that the objects are grasped by the intended parts, according to the user demonstration, and did not find a solution for the manipulation planning problem unless the number of sampled grasps per object is larger than 270. The reason is that for a lower number of samples the standard algorithm does not synthesize grasps on the legs of the table (which are required in order to perform action A_1).

The second experiment includes three objects (a cup, a plier and a small horse) and four recognizable augmented actions: grasping the caged cup from the body and laying it on the ground (action A_2), grasping the plier from one handle and peg it into the cup (action B_2), grasping the horse from a leg and laying it on the ground (action C_2), grasping the horse from the body, turning it and placing it on the ground (action D_2). The cup and the horse are side-by-side. The user provides two demonstrations of the task. In the first demonstration (reported in figure 7) the user performs the sequence of actions $A_2B_2C_2D_2$, while in the second demonstration the sequence is $C_2D_2A_2B_2$. The system learns two precedence relations: (A_2, B_2), i.e. the plier should be put in the cup after laying the cup on the table, and (C_2, D_2). Table II reports the average time of each phase. The second experiment confirms the results of the first experiment as the non part-based algorithm is not able to find a solution to the planning problem for a low number of sampled grasps (i.e. action C_2 can not be planned since the bounding box sampling algorithm does not generate grasps on the legs of the horse).

VI. CONCLUSIONS

In this paper a novel method for robot manipulation planning has been presented that enables semantic grasping of previously unseen objects. The approach is based on programming by demonstration in virtual reality and 3D shape segmentation. We have adopted Reeb graphs for shape segmentation as they provide a topological representation

TABLE I
EXPERIMENT 1: AVERAGE TIME (SEC.) FOR EACH PHASE

	Part-based	Non part-based	Non part-based
grasps/part	(20)		
grasps/object		(90)	(270)
Object segmentation	120	-	-
Object recognition	0.8	-	-
Task demonstration (1 run)	45	45	45
Grasp map computation	39	87	237
Manipulation planning	68	dnf	81

TABLE II
EXPERIMENT 2: AVERAGE TIME (SEC.) FOR EACH PHASE

	Part-based	Non part-based	Non part-based
grasps/part	(20)		
grasps/object		(90)	(270)
Object segmentation	95	-	-
Object recognition	0.8	-	-
Task demonstration (1 run)	48	48	48
Grasp map computation	51	90	246
Manipulation planning	73	dnf	95

of the objects. We have also described how topological shape decomposition can be exploited for object recognition and grasping-by-parts. Part-based grasping is an important feature for task oriented manipulation. However, it has been considered only to a limited extent in previous works. The proposed manipulation planning system learns from human demonstration and it is capable of performing intelligent grasps of objects by their parts. In our future work we are interested in evaluating the manipulation planning system in a real robot workspace with 3D range scan sensors coping with incomplete sensory data.

ACKNOWLEDGMENT

We thank the AIM@SHAPE network of excellence and the FOCUS K3D Coordination Action of the European Union for providing the 3D object database.

REFERENCES

[1] J. Aleotti, S. Caselli, and M. Reggiani. Leveraging on a virtual environment for robot programming by demonstration. *Robotics and Autonomous Systems*, 47(2-3):153–161, 2004.

[2] D. Berenson, R. Diankov, K. Nishiwaki, S. Kagami, and J. Kuffner. Grasp planning in complex scenes. In *7th IEEE-RAS Int'l Conference on Humanoid Robots*, pages 42–48, nov. 2007.

[3] D. Berenson and S.S. Srinivasa. Grasp synthesis in cluttered environments for dexterous hands. In *8th IEEE-RAS Int'l Conference on Humanoid Robots*, pages 189–196, dec. 2008.

[4] S. Berretti, A. Del Bimbo, and P. Pala. 3D Mesh decomposition using Reeb graphs. *Image and Vision Computing*, 27(10):1540–1554, 2009.

[5] S. Biasotti, D. Giorgi, M. Spagnuolo, and B. Falcidieno. Reeb graphs for shape analysis and applications. *Theoretical Computer Science*, 392(1-3):5–22, 2008.

[6] I. Biederman. Recognition-by-Components: A Theory of Human Image Understanding. *Psychological Review*, 94:115–147, 1987.

[7] S. Ekvall and D. Kragic. Learning and Evaluation of the Approach Vector for Automatic Grasp Generation and Planning. In *IEEE Intl Conference on Robotics and Automation, (ICRA)*, pages 4715–4720, Rome, Italy, 2007.

[8] J.J. Gibson. *The Ecological Approach to Visual Perception*. Houghton Mifflin, 1979.

[9] M. Gienger, M. Toussaint, and C. Goerick. Task maps in humanoid robot manipulation. In *IEEE/RSJ Intl Conference on Intelligent Robots and Systems, (IROS)*, pages 2758–2764, sep. 2008.

[10] C. Goldfeder, P. K. Allen, C. Lackner, and R. Pelossof. Grasp Planning via Decomposition Trees. In *IEEE Intl Conference on Robotics and Automation, (ICRA)*, pages 4679–4684, Roma, Italy, 2007.

[11] C. Goldfeder, M. Ciocarlie, Hao Dang, and P.K. Allen. The Columbia Grasp Database. In *IEEE Intl Conference on Robotics and Automation, (ICRA)*, pages 1710–1716, Kobe, Japan, May 2009.

[12] Y. Hirano, K. Kitahama, and S. Yoshizawa. Image-based object recognition and dexterous hand/arm motion planning using RRTs for grasping in cluttered scene. In *IEEE/RSJ Intl Conference on Intelligent Robots and Systems, (IROS)*, pages 2041–2046, aug. 2005.

[13] D. D. Hoffman and W. A. Richards. Parts of Recognition. *Cognition*, 18(1-3):65–96, 1984.

[14] K. Hsiao and T. Lozano-Perez. Imitation Learning of Whole-Body Grasps. In *IEEE/RSJ Intl Conference on Intelligent Robots and Systems, (IROS)*, pages 5657–5662, Oct. 2006.

[15] K. Huebner, S. Ruthotto, and D. Kragic. Minimum Volume Bounding Box Decomposition for Shape Approximation in Robot Grasping. In *IEEE Intl Conference on Robotics and Automation, (ICRA)*, pages 1628–1633, Pasadena, USA, May 2008.

[16] U. Klank, D. Pangercic, R.B. Rusu, and M. Beetz. Real-time CAD model matching for mobile manipulation and grasping. In *9th IEEE-RAS Int'l Conference on Humanoid Robots*, pages 290–296, dec. 2009.

[17] F. Kyota, T. Watabe, S. Saito, and M. Nakajima. Detection and Evaluation of Grasping Positions for Autonomous Agents. In *International Conference on Cyberworlds*, Nov. 2005.

[18] Y. Li, J. Fu, and N. Pollard. Data Driven Grasp Synthesis using Shape Matching and Task-Based Pruning. *IEEE Transactions on Visualization and Computer Graphics*, 13(4):732–747, 2007.

[19] L. Montesano and M. Lopes. Learning affordance visual descriptors for grasping. In *IEEE Intl Conference on Development and Learning, (ICDL)*, Shanghai, China, 2009.

[20] A. Morales, T. Asfour, P. Azad, S. Knoop, and R. Dillmann. Integrated grasp planning and visual object localization for a humanoid robot with five-fingered hands. In *IEEE/RSJ Intl Conference on Intelligent Robots and Systems, (IROS)*, pages 5663–5668, oct. 2006.

[21] F. Rothling, R. Haschke, J.J. Steil, and H. Ritter. Platform portable anthropomorphic grasping with the bielefeld 20-DOF shadow and 9-DOF TUM hand. In *IEEE/RSJ Intl Conference on Intelligent Robots and Systems, (IROS)*, pages 2951 –2956, oct. 2007.

[22] A. Sahbani and S. El-Khoury. A Hybrid Approach for Grasping 3D Objects. In *IEEE/RSJ Intl Conference on Intelligent Robots and Systems, (IROS)*, pages 1272–1277, St. Louis, MO, 2009.

[23] S. Shalom, L. Shapira, A. Shamir, and D. Cohen-Or. Part Analogies in Sets of Objects. In *Eurographics Workshop on 3D Object Retrieval*, 2008.

[24] T. Simeon, J. Cortes, A. Sahbani, and J.P. Laumond. A manipulation planner for pick and place operations under continuous grasps and placements. In *IEEE Intl Conference on Robotics and Automation, (ICRA)*, pages 2022–2027, 2002.

[25] J.D. Sweeney and R. Grupen. A Model of Shared Grasp Affordances from Demonstration. In *IEEE-RAS Intl Conference on Humanoid Robots*, pages 27–35, Dec. 2007.

[26] T. Tung and F. Schmitt. Augmented Reeb Graphs for Content-Based Retrieval of 3D Mesh Models. In *Proceedings of the Shape Modeling International, (SMI)*, pages 157–166, Washington, DC, USA, 2004.

[27] K. Yamazaki, M. Tomono, T. Tsubouchi, and S. Yuta. A grasp planning for picking up an unknown object for a mobile manipulator. In *IEEE Intl Conference on Robotics and Automation, (ICRA)*, pages 2143–2149, may. 2006.

[28] F. Zacharias, C. Borst, and G. Hirzinger. Bridging the gap between task planning and path planning. In *IEEE/RSJ Intl Conference on Intelligent Robots and Systems, (IROS)*, pages 4490–4495, oct. 2006.

[29] M. Zaslavskiy, F. Bach, and J.-P. Vert. A Path Following Algorithm for the Graph Matching Problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(12):2227 –2242, dec. 2009.

[30] R. Zöllner, M. Pardowitz, S. Knoop, and R. Dillmann. Towards Cognitive Robots: Building Hierarchical Task Representations of Manipulations from Human Demonstration. In *IEEE Intl Conference on Robotics and Automation*, pages 1535–1540, 2005.