Selection of Robot Pre-Grasps using Box-Based Shape Approximation

Kai Huebner and Danica Kragic

***Abstract*— Grasping is a central issue of various robot applications, especially when unknown objects have to be manipulated by the system. In earlier work, we have shown the efficiency of 3D object shape approximation by box primitives for the purpose of grasping. A point cloud was approximated by box primitives [1]. In this paper, we present a continuation of these ideas and focus on the box representation itself. On the number of grasp hypotheses from box face normals, we apply heuristic selection integrating task, orientation and shape issues. Finally, an off-line trained neural network is applied to chose a final best hypothesis as the final grasp. We motivate how boxes as one of the simplest representations can be applied in a more sophisticated manner to generate task-dependent grasps.**

I. INTRODUCTION

In a service robot scenario, robot grasping capabilities are necessary to actively execute tasks, interact with the environment and thereby reach versatile goals. These capabilities also include the generation of stable grasps to safely handle even objects unknown to a robot. In earlier work [1], we motivated the idea that the key to this ability is not primarily to select a grasp depending on the identification of a selected object, but rather on its shape. We presented an algorithm that efficiently wraps given 3D data points of an object into primitive box shapes by a fit-and-split algorithm based on Minimum Volume Bounding Boxes. Though box shapes are not able to approximate arbitrary data in a precise manner, it was shown that they give efficient clues for planning grasps on arbitrary objects or object parts. This seems reasonable, since it should not be necessary to find the most stable grasp, but sufficient to find one of those that are stable. Additionally, the part-describing box concept allows for grasp semantics mapped to boxes in the set, e.g. "*approach the biggest part to stably move the object*" or "*approach the smallest part to show a most unoccluded object to a viewer.*" The description of an object by a shape-based part representation, which is claimed to be necessary for this kind of task-dependent grasping, is thereby made available, and also needed as a criterion what grasp is the "best" in terms of a given task.

In this context, we present our novel approach for connecting shape, boxes, tasks and grasping in this paper. We briefly introduce our basic work as also other related work in Section II. While we refer to [1] for the description of the box decomposition algorithm, we focus on taking advantage of the box representation. We develop a sequence of steps, including heuristics and learning of grasp qualities to select one final, task-dependent grasp for an object. We will discuss the simple ideas that are used to reach this goal in Section III. Section IV practically shows an experiment, where we connect to 3D data from a real, though convenient scene for the first time. We finally conclude our work in Section V.

The authors are with the KTH – Royal Institute of Technology, Stockholm, Sweden, as members of the Computer Vision & Active Perception Lab., Centre for Autonomous Systems, www home page: `http://www.csc.kth.se/cvap`, e-mail addresses: {`khubner,danik`}`@kth.se`.

II. RELATED WORK

When talking about a robot grasping unknown objects, one has to think about a representation that not only eases grasping, but which can also be efficiently delivered from the sensor data. Though there is interesting work on producing grasp hypotheses by visual features from 2D images only, e.g. [2], most techniques rely on 3D data. 3D data, which in its simplest form may be a set of 3D points belonging to an object's surface, can be produced by several kinds of sensors and techniques, e.g. distance imaging cameras, laser scanners or stereo camera systems. Since the last solution is cheap, easy to integrate and close to the human sensory system, a multitude of concepts in the area use 3D point cloud data from stereo disparity. These point clouds are usually afflicted with sensor noise and uncertainties, which has to be taken into account for precise shape approximation of such data. In [1], we have referenced and stated our claim that precise shape approximation, e.g. using superquadrics, might not be necessary for extracting grasp hypotheses. The work of Lopez-Damian *et al*. [3], [4] is related to ours in terms of object decomposition and grasping. Additionally, they propose a grasp planner to find a stable grasp. However, their concept uses polygonal structures instead of 3D points. Though one could produce polygonal surfaces from 3D point data, for example by the Power Crust algorithm [5], this introduces another step causing additional effort both in processing time and noise handling. In this paper, we have also used the Power Crust, but only to visualize the 3D data.

It has to be mentioned that our approach is not explicitely handling contact-level grasp planning. A grounded theory on stable contact-level grasps has been developed in the literature. Conclusions of the ideas and outcome can be found in [6], [7]. In this theory of grasp planning, finger contact locations, forces and grasp wrench spaces can be simulated. Different criterions can be defined to rate grasp configurations, e.g. force closure, dexterity, equilibrium, stability and dynamic behavior [6]. However, the dependency on a-priori known or dense and detailed object models is apparent. Miller *et al*. [8] therefore proposed grasp planning on simple shape primitives, like spheres, cylinders and cones, clearly demanding a pre-classification of object shape. Dependent on the primitive shape, one can test several grasp configurations on this shape. This work was continued by Goldfeder *et*

al. [9], using more sophisticated shape primitives, known as superquadrics.

In our work, we also work with shape primitives. We chose the box shape as one of the most simple ones and integrate an efficient bounding box algorithm for 3D point data [10]. However, while the classical contact-level solution includes a merge of both transport (leading the hand to the grasp position) and grip (closing the fingers to perform the grasp), we see a benefit in loosely decoupling these two components. The psychophysical shortcomings of completely decoupling the grip from the transport component have been discussed in [11], even if this is described as the classical approach. It is also hardly questioned that the transport component refers to extrinsic object properties only (e.g. position, orientation) while the grip component depends on intrinsic properties (e.g. size, shape, weight). Derbyshire *et al*. [12] even motivate action to be an intrinsic property.

The work presented here does neither separate nor combine these two components. It is more a connecting module inbetween them. First, the transport component is just seen as a predecessor. It would demand grasp planning and collision detection in a definition of successful robot hand transport, being a research topic for itself. However, the final location of a grasp is also clearly dependent on the task at hand, making the task another extrinsic property.

Second, the grip component is a successor of our grasp hypotheses generation. The final grip is not handled in a comparable way to classical contact-level grasp planning, as this connects directly to all perceptually sensed intrinsic properties. Thus, we classify our idea as a pre-grip component that is both dependent on selected extrinsic (orientation, task) and intrinsic (size, shape) properties (see bold in Tab. I). We see precise shape, weight or surface texture properties as being part of an adjacent fine-controller based on tactile feedback and corrective movements, like included in [13].

III. FEATURES OF THE BOX REPRESENTATION

The result of our box decomposition technique is the following: given a set of 3D points, we can find a compact box set $\mathcal{B} = \{B_1, \ldots, B_n\}$ that encloses the points and thereby offers a primitive shape approximation. For each box B_i in the set, we focus on its six rectangular faces $\{F_{(i,1)}, \ldots, F_{(i,6)}\}$. In [1], each face spawned up to four grasp hypotheses by using the face normal as approach vector and the four edges as orientation vectors, using a pre-defined grasp. Fig. 1 shows some of the models that were used, a model of a 5-finger hand, as also an exemplary box decomposition of the duck model. Finally, we showed that even if we drastically reduce the grasp hypotheses,

TABLE I

GRASP COMPONENTS AND OBJECT PROPERTIES

Grasp component	extrinsic properties	intrinsic properties
Transport	position	–
Pre-Grip	**orientation, task**	**size, rough shape**
Grip	–	precise shape, weight, surface texture

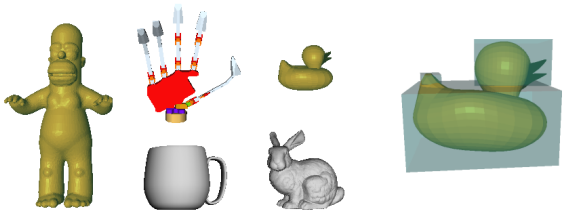

Fig. 1. Left: Some objects and a robot hand model, simulated in GraspIt! [14]. Right: A result of the box approximation for the duck model [1].

this concept does not significantly reduce the grasp quality, but opens up new possibilities like task-oriented grasping or object part description. We will now present some of these issues, which have been integrated in a grasp selection mechanism, starting with task-dependencies.

A. Task Dependencies

Task dependency of grasps is an important issue, which shows that "best" grasps do not have to be the most stable ones. Picking up a cup from the "open side" will be unsuitable for the task of filling the cup, as a very stable full-enclosing grasp (power-grasp) will be unsuitable for handing over or presenting the cup to someone. Application of such re-usability semantics by defined keep-out zones has been proposed in [15]. Object properties like hollowness are hard to detect for today's systems, as also are high-level properties like *filled* or *empty*. Our box set method allows intuitive mapping of less complex actions to simple box properties.

Given a box set $\mathcal{B}$, one can easily compute criteria like the overall mass center (assuming uniformly distributed mass density), each volume and dimension of a box, or the relations between boxes. For example, one can define the *outermost* or *innermost*, the *largest* or *smallest*, the *top* or the *bottom*, etc. Given a task, we can easily map an action like *pick-up*, *push*, *show*, *rotate*, etc., to a selected box. In fact, we can even order the boxes according to the above criteria. For example, in order to *pick-up* something to place it somewhere else, it may intuitively be a good choice to grasp the *largest* box. When showing the same object to a viewer, it may be better to grasp the *outermost* box.

Similarly, different grasp configurations can be linked to tasks when using a simple representation like a box. We apply another simple mapping from an action to a pre-defined movement here. We already introduced two of these in [1]: the backup power-grasp, which approaches a box until contact, retreats a bit and than closes fingers simultaneously, and the pincher-grasp, which approaches the box until it is in position to closing fingers and contact the box most centrally. One might extend this idea towards the selection of different grasp pre-shapes [16], or even the selection of controllers for different tasks. In fact, Prats *et al*. [17] also use box representations for task-oriented grasping with hand pre-shapes and task frames. However, they assume geometrical knowledge about each object (using a database of 3D models) and structural and mechanical knowledge about a task (e.g. "turning" a door handle).

B. Box Face Visibility

From the box level, we now continue to the face level. Each box provides six rectangular faces in 3D space. Here, we have to consider that incomplete data is produced by a single sensor view of an object, as the back of the object is not visible. Thus, box decompositions are clearly view-dependent and do only envelope visible data points. For this reason, it may be helpful to only take those box faces into account that are visible from the viewpoint. Note that here, "visibility" is understood as the face being oriented towards the viewpoint only, not being visible in sense of occlusion by other objects. We see another motivation for a face visibility check considering the relation between an end-effector, i.e. the robot hand, and the object. Intuitively, humans tend to use grasping movements that involve minimum activity effort. A short experiment at least showed evidence for this:

Test persons had to grasp various objects on a table to describe their appearance, thus the task of grasping was implicit. It showed up that in case of cups, the handle was pinch-grasped when it was orientated towards the human hand, while otherwise the cup body was power-grasped.

Though this experiment is not compelling in terms of a psychophysical evaluation and will therefore not be described any further, it is intuitive in the same way as the viewpoint face check. Valid faces can thereby be selected by being accessible from a given end-effector viewpoint, even if one end-effector might be busy, e.g. holding another object.

In opposition to these observations that a visibility check keeps a large potential, the technical computation if a 3D plane is oriented towards or away from a 3D point is trivial and easy to use. In this way, we also integrate orientation properties into our concept.

C. Box Face Occlusion and Blocking

While the visibility criterion is a check for orientation of faces towards a camera's or an end-effector's viewpoint, occlusions and blockings between faces in the box set are also considered. As an example, grasping the head of the duck (Fig. 1) towards the bottom face is not profitable, as this face is "occluded" by a face of the body box. In another way, one may also classify other duck head grasps as being unprofitable. Imagine the 5-finger hand grasping the duck's head box B_1 from one of the side faces and have in mind that the fingers will not contact the approached face $F_{(1,a)}$, but two of its neighbors, $F_{(1,b)}$ and $F_{(1,c)}$, depending on the grasp orientation. We then define $F_{(1,a)}$ as "blocked" in this grasp orientation, if $F_{(1,b)}$ or $F_{(1,c)}$ is occluded, and remove these grasp hypotheses from the set.

This technique has proven to be very useful in further reducing the number of hypotheses. Technically, the detection of opposing faces is more complex than the visibility check and therefore forms the end of the heuristical selection sequence. Each face of a box has to be compared to each face of all other boxes. The handling of such situations demands an additional computational effort. For this reason, and as it reduces the number of hypotheses drastically, we currently strictly remove all occluded and blocked hypotheses from our selection.

It may be mentioned that the calculations necessary are purely geometrical problems on faces and points. Like the whole grasp selection process, visibility, occlusion and blocking are currently computed in software (C/C++), one might think about taking advantage of graphical processors to speed up and optimize the geometrical operations.

D. Projection Grids and Learning

The previous steps have been heuristical, aiming at reducing the number of grasp hypotheses according to an object's task, orientation and shape. Even if it was not named, also the size, i.e. the dimensions of a face, is considered. A face that exceeds the maximum grasp opening in one dimension cannot be grasped. However, there is usually a set of remaining hypotheses from which we would like to select one final grasp. Our current approach to this issue is learning of grasp qualities from 2.5D shape projections.

Considering a box and the points that it envelopes, each face produces a projection of the points onto the face plane. In fact, these projections were already computed for best cut detection [1]. Discretization was made by dividing the face into equally sized cells, thus projections were represented as dynamically sized binary grids. To adapt this representation and enrich it, we now compute linear information, i.e. minimum distance information to the face plane, in a normalized, fixed-sized grid. Fig. 3 shows 18 of such projection grids with size 15×15 for the faces produced by the duck decomposition in Fig. 1. This representation both allows analyzing the 2.5D depth map of each face and fulfills the input space conditions of a classical neural network learner like the one we will use here (see Fig. 2).

In the following experiment three models (homer, mug and duck, see Fig. 1) have been processed by the algorithm and the projections been grasped in the grasp simulator GraspIt! [14]. By providing the two quality measures *eps*, a worst-case epsilon measure for force-closure grasps, and *vol*, an average case volume measure, GraspIt! is automatically used as a teacher for the supervised network, estimating the stability of a grasp on a given face *F* and its 2.5D projection grid *proj(F)*, respectively. Since due to the normalization in width, height and depth, information about the dimension of *F* is lost, the box dimensions *dim(F)* are added in terms of three additional neural network inputs.

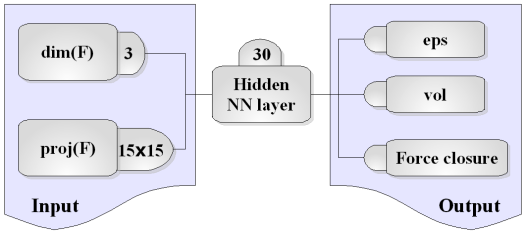

Fig. 2. The neural network structure for off-line learning of grasp qualities from face representations. It holds 228 input, 30 hidden and 3 output neurons. *eps* and *vol* are grasp quality measures that GraspIt! delivers [14]. The force closure is also learned separately even if it equals (*eps* > 0).

IV. EXPERIMENT

We will now present an experiment on the determination of one final grasp hypothesis from a real 3D point cloud. The 3D data is produced from disparity using a stereo vision system, consisting of a Yorick [18] head equipped with two Allied Vision Marlin cameras. The scene is shown in Fig. 4a. As earlier experiments have been performed in simulation only, one focus of the experiment is to test the box decomposition on real 3D data which is influenced by natural dense stereo noise and incompleteness. The second focus is the practical processing of the proposed heuristical and learning selection mechanism, including the considered decisions on task, view-point, shape and size properties.

A. Producing 3D Data

In Fig. 4b, the disparity image produced by the stereo image pair can be seen. It is clearly influenced by incompleteness, both observable by some holes and by the backside which is not visible. Additionally, and though we have cared for a uniform background, there is little noise at the bottom left of the image. The effects of these uncertainties become clearer in Fig. 4c, representing the 3D model of the object.

B. Box Decomposition

We use the box decomposition algorithm [1] to deliver a box approximation of the point cloud. The decomposition steps can be seen in Fig. 6. The fit-and-split algorithm iteratively fits and splits minimum volume bounding boxes, initially starting with the root box enclosing all points (Fig. 6b). The first split, chosen due to maximum volume gain, nicely cuts the outliers from the main shape. The gain parameter Θ^* of 0.41 relates to the new overall box volume being 41% of the box volume before the cut. Out of the two new boxes, the one including the noise keeps to few points and thus is automatically removed. During the following cuts (Fig. 6c-d), the volume gain value increases continuously, since the more the boxes approximate the shape, the less volume can be gained by a cut. After three cuts, the algorithm stops, as a gain threshold below 0.93 will not be reached by any new split. The gain threshold is a parameter of the algorithm and manually set. In practice, threshold values

(a) Stereo images (b) Disparity image (c) 3D points

Fig. 4. 3D points from disparity. (a) shows the two images taken from the stereo vision system. From those, disparity values are calculated (b). These can be used to produce 3D points (c). Note that polygonal structures have just been artificially computed with PowerCrust [5] to visualize results. The box decomposition algorithm uses pure 3D point data only.

between 0.9 and 0.95 have led to good results. The higher the threshold, the more cuts will be applied and the more precise the shape will be approximated by boxes.

C. Heuristical Box and Face Selection

With three boxes in the final set, 18 faces and their projections can be accessed. As the decomposition of the real duck is different from the model duck, the box constellations and the projection faces are different. Due to noise and resolution, they are even hard to recognize for the human eye (compare Fig. 5 with Fig. 3 to its left).

As we restrict to grasp orientations parallel to a face's edge, each of the faces theoretically produces four grasps of different orientation. On all 18 faces, this would make 72 theoretical grasp hypotheses available. If we chose a selection of a box by giving an initial task (see Section III-A), as we will do in the following two examples, we could reduce this set to one box with 6 faces, according to 24 grasp hypotheses. The face check selection according to occluded and blocked grasp hypotheses (Section III-C) is presented exemplary for the duck's head box in Fig. 7. As stated, six faces yield four grasp hypotheses each. These rotations are easy to process from one source projection, as the transformation only includes coordinate switching. One face has completely been rejected by occlusion check. It corresponds to the bottom face of the head box. Other faces have been blocked with respect to grasp directions. Intuitively, these are exactly those grasp hypotheses that would cause finger contact on the bottom face, which is

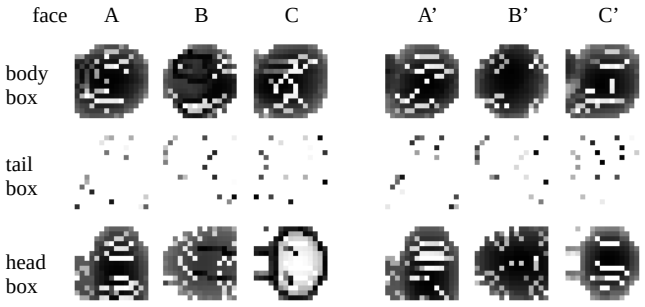

Fig. 3. The set of projection grids for the decomposition in Fig. 1. Three boxes result in 18 faces, where 15×15 grid resolution was chosen. Note that the tail projection is noisy as there are very few points in a very small box. Also note the difference between Head C and C'. C is from below, showing the hollow head, while C' is the projection of the head top.

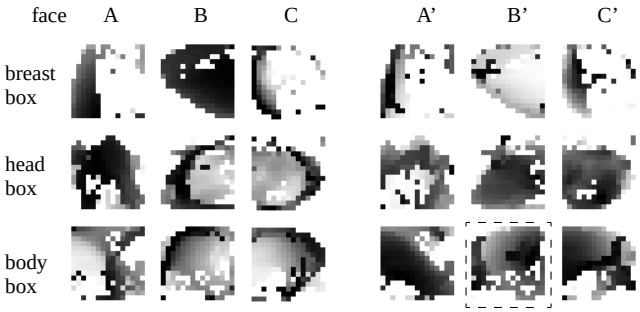

Fig. 5. The set of projection grids for the decomposition in Fig. 6e. Three boxes result in 18 faces, where 15×15 grid resolution was chosen. As the grids are low-resolution, shape is hard to recognize for the eye. One might see the duck pecker facing upwards in head B'. Head B, its opposing face, is visibly a hollow shape. The *pick* grasp is on Body B' (Fig. 8 left).

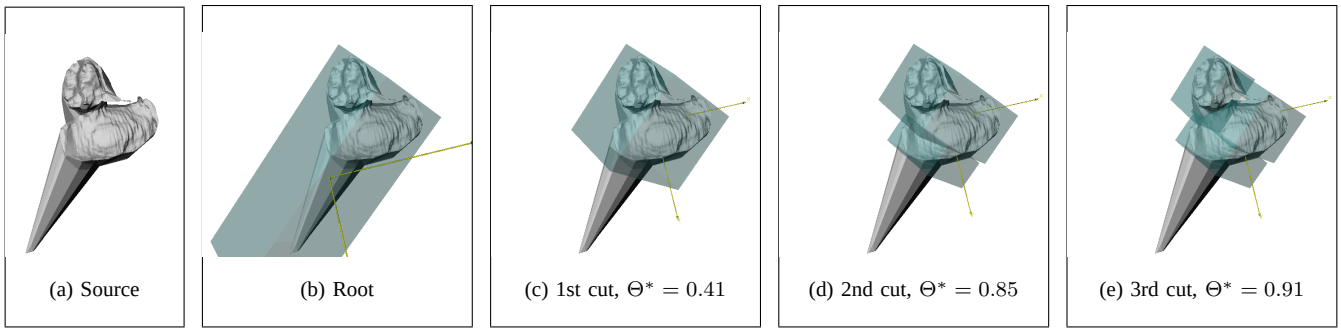

Fig. 6. Decomposition on the source data (a) with a gain threshold of 0.93: (b) The first approximation produces the root box of all points. (c) The first cut separates the noise from the shape. Noise are very few points, so these are not treated further. (d) shows the 2nd cut which still has a good volume gain of 85%. (e) presents the final cut, as further steps did not reach a gain smaller than 0.93.

occluded. For this example, the set of grasp hypotheses is thereby reduced from 24 to 12 hypotheses. Having in mind the option of a viewpoint check as discussed in Section III-B, these could further be reduced to 8, as the head top (C') and only two of the opposing faces (A or A' and B or B') are oriented towards the camera. Note that all the heuristics are optional and not dependent on each other. Using all of them, 72 initial hypotheses were reduced to 8 in this example.

D. Final Grasp Decision and Learning

After having reduced the hypotheses to a small set, we have to finally decide where and how to grasp. The "where" component equals a decision on grasping one of the faces with one orientation. To do this, we apply the neural network structure presented in Section III-D. The face projections of the remaining hypotheses are fed into the net that has been previously off-line trained with artificial examples. After sorting out those hypotheses that do not result in good force-closure response larger than 0.5 (third output), we decide for the one hypothesis with optimal vol grasp quality.

Until here, we have not explicitely mentioned the task-dependent decisions (Section III-A). Assume these two tasks and have a look on the corresponding results in Fig. 8:

(T1) $task : pick \quad \rightarrow box : largest, grasp : backup,$
(T2) $task : show \quad \rightarrow box : outermost, grasp : pincher.$

The derivation of the final grasp has been performed as presented, where depending on the task, a box selection has been applied. On the final set of hypotheses grids, the one that the trained neural network votes best for is selected. For both examples, these final hypotheses are also marked in Fig. 5 and 7, respectively. Note that in Fig. 5 the selected projection that keeps the best hypotheses is marked for (T1), body box face B', while in 7 the best hypothesis, head box face C' 90^o, is shown for (T2). Additionally, the different choice of grasp type is visible in Fig. 8. In the pick task, the backup grasp focusses on enclosing the whole box, while for the show task the pincher grasp focusses on placing fingers centrally to the contact faces.

In this example, the 3D point cloud had 86310 points, the decomposition algorithm tried 6 fit-and-split iterations, whereof 3 were successful. The decomposition is still the main effort in computation time, it took 22 seconds. The computation of projections as also the heuristical and neural network decisions are neglectable, taking altogether less than half a second. The experiments were performed on a Double Intel Core2 Quad CPU with 2.66 GHz.

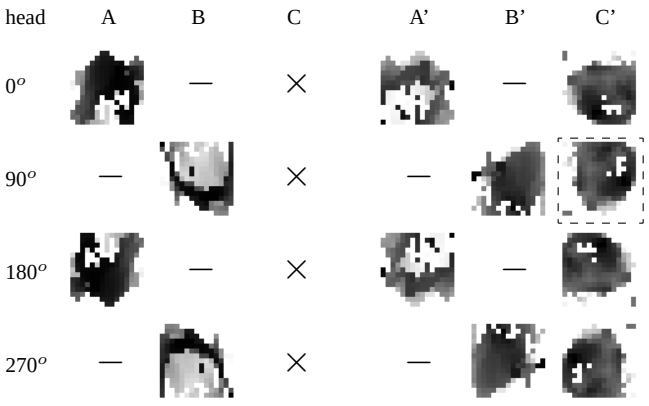

Fig. 7. Face check selection for the duck's head box only. Each head face (columns; see Fig. 5) gives access to four different grasp orientations (rows). Note Head C being completely occluded ($\times$), as it is the face that connects to the Body box. Some grasp directions are blocked ($-$) from the side. The *show* grasp is on Head C' 90^o (Fig. 8 right).

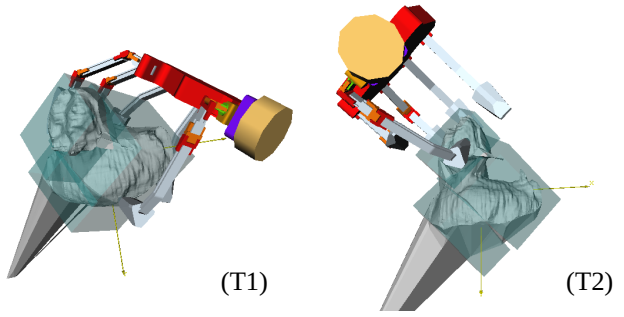

Fig. 8. Left: The final decision for the *pick* grasp (T1). Gripper configuration has been chosen to be the *backup* grasp for enclosing the object. Only faces of the *largest* box (body) have been taken into account. The algorithm finally decided for the top face of the body box, as the neural net forecasted the best grasp qualitiy measure on its projection.
Right: The final decision for the *show* grasp (T2). Gripper configuration has been chosen to be the *pincher* grasp for putting fingertips on face center points. Only faces of the *outermost* box (head) have been taken into account. The algorithm finally decided for the top face of the head box, as the neural net forecasted the best grasp quality measure on its projection.

V. CONCLUSIONS

We presented the continuation of box approximation for the purpose of robot grasping. While we specified the core algorithm of box approximation in earlier work, we now concentrated on subsequent steps that all take advantage of the very simple shape representation of boxes. Starting from boxes and their faces that the core algorithm produces, we extended the idea of "grasping on boxes" towards an applicable grasping strategy. This strategy only includes heuristical selection based on efficient geometrical calculations, as also learning from off-line simulation. Basic task-dependencies have been included in this process easily. We see the strength of our approach in its simplicity and its modularity. The simplicity is clear by using boxes and faces in 3D space. Geometric calculations are much more easy to do in contrast to more sophisticated shape primitives like superquadrics. As presented, boxes and faces can additionally take advantage of linear shape projections. The modularity is established by mostly independent criterions and heuristics that complement each other and even leave space for extensions.

There are many possibilities to extend and optimize the current framework both in theory and practice. In theory, we have to evaluate and optimize the current algorithm. Considerations have to be made for the neural net structure, e.g. if it might be better to extend the learning to grasp qualities dependent on the chosen grasp pre-shape, i.e. setting three quality outputs for *each* available grasp pre-shape. Additionally, the simulation part for learning is currently done using static simulation. Thus, contact will stay static between gripper and object, while in dynamics, and reality, the object pose will change dependent on the force applied to it. We are working on this issue also with regard to what we called the grip component. For the sake of efficiency and intuitive motivation, we are aware that our approach is a pre-grip component on very robust shape information. The grip component, as an additional module, would contribute in terms of fine correction based on haptic feedback [13]. In practice, we are still missing some necessary parts to physically perform a grasp with a real robot manipulator. Our current work is also on putting these parts together and connect them to the work proposed here.

The box representation of an object is simple. However, the projection of an object onto the box faces ignores the real 3D shape of the object in the box, not considering the correct surface normals of the object in the grasp planning. Thus, there is a possibility that planned grasps are infeasible, which addresses the limitation of the proposed planning. In future work, we will examine finger positioning estimations on the projections, connected to the work of Morales *et al*. [19]. The effectiveness of the approach in real applications has also to be evaluated through experiments.

As future work, one could also imagine higher-level part classification. Given all three projections of a box, one could try to learn and classify the enclosed shape, which with high probability corresponds to an object part. This relates to work on view-based object (part) representation. Classification of shape is a beneficial, but also complex task, as additionally, the box constellation might be very different as influenced by noise, perspective view and uncertainties (e.g. compare the different box constellations of the two ducks in Fig. 1 and Fig. 6e). For the purpose of grasping on faces, this is not a very severe problem, while in part and object classification, it probably will be. Therefore, evaluations of these high-level ideas are not a topic of our short-term goal.

VI. ACKNOWLEDGMENTS

This work was supported by EU through the project PACO-PLUS, IST-FP6-IP-027657.

REFERENCES

[1] K. Huebner, S. Ruthotto, and D. Kragic, "Minimum Volume Bounding Box Decomposition for Shape Approximation in Robot Grasping," in *Proceedings of the 2008 IEEE International Conference on Robotics and Automation*, 2008, pp. 1628–1633.

[2] A. Saxena, J. Driemeyer, and A. Y. Ng, "Robotic Grasping of Novel Objects using Vision," *Journal of Robotics Research*, 2008.

[3] E. Lopez-Damian, D. Sidobre, and R. Alami, "Grasp Planning for Non-Convex Objects," in *International Symposium on Robotics*, 2005.

[4] E. Lopez-Damian, "Grasp Planning for Object Manipulation by an Autonomous Robot," Ph.D. dissertation, Laboratoire d'Analyse et d'Architecture des Systèmes du CNRS, 2006.

[5] N. Amenta, S. Choi, and R. Kolluri, "The Power Crust," in *6th ACM Symposium on Solid Modeling and Applications*, 2001, pp. 249–260.

[6] K. Shimoga, "Robot Grasp Synthesis Algorithms: A Survey," *Journal of Robotic Research*, vol. 15, no. 3, pp. 230–266, 1996.

[7] A. M. Okamura, N. Smaby, and M. R. Cutkosky, "An Overview of Dexterous Manipulation," in *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*, 2000, pp. 255–262.

[8] A. T. Miller, S. Knoop, H. I. Christensen, and P. K. Allen, "Automatic Grasp Planning Using Shape Primitives," in *Proceedings of the 2003 IEEE International Conference on Robotics and Automation*, 2003, pp. 1824–1829.

[9] C. Goldfeder, P. K. Allen, C. Lackner, and R. Pelossof, "Grasp Planning Via Decomposition Trees," in *Proceedings of the 2007 IEEE International Conference on Robotics and Automation*, 2007.

[10] G. Barequet and S. Har-Peled, "Efficiently Approximating the Minimum-Volume Bounding Box of a Point Set in Three Dimensions," *Journal of Algorithms*, vol. 38, pp. 91–109, 2001.

[11] J. B. J. Smeets and E. Brenner, "A New View on Grasping," *Motor Control*, vol. 3, pp. 237–271, 1999.

[12] N. Derbyshire, R. Ellis, and M. Tucker, "The potentiation of two components of the reach-to-grasp action during object categorisation in visual memory," *Acta Psychologica*, vol. 122, pp. 74–98, 2006.

[13] J. Tegin, S. Ekvall, D. Kragic, B. Iliev, and J. Wikander, "Demonstration based Learning and Control for Automatic Grasping," in *Proc. of the International Conference on Advanced Robotics*, 2007.

[14] A. T. Miller and P. K. Allen, "Graspit! A Versatile Simulator for Robotic Grasping," *Robotics & Automation Magazine, IEEE*, vol. 11, no. 4, pp. 110–122, 2004.

[15] T. Baier and J. Zhang, "Reusability-based Semantics for Grasp Evaluation in Context of Service Robotics," in *Proc. of the International Conference on Robotics and Biomimetics*, 2006, pp. 703–708.

[16] M. Cutkosky, "On Grasp Choice, Grasp Models and the Design of Hands for Manufacturing Tasks," *IEEE Transactions on Robotics and Automation*, vol. 5, pp. 269–279, 1989.

[17] M. Prats, P. J. Sanz, and A. P. Del Pobil, "Task-Oriented Grasping using Hand Preshapes and Task Frames," in *Proceedings of the 2007 IEEE International Conference on Robotics and Automation*, 2007, pp. 1794–1799.

[18] Interactive Systems Research Group (ISRG), University of Reading, "Yorick robot head series." [Online]. Available: http://www.isrg.reading.ac.uk/yorick/index.htm

[19] A. Morales, E. Chinellato, A. H. Fagg, and A. P. del Pobil, "Experimental Prediction of the Performance of Grasp Tasks from Visual Features," in *Proceedings of the 2003 IEEE/RSJ International Conference on Robots and Systems*, 2003, pp. 3423–3428.