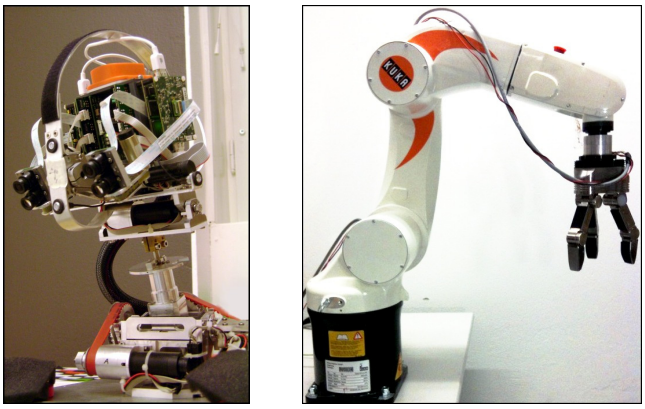
Scene Representation and Object Grasping using Active Vision

Xavi Gratal, Jeannette Bohg, Mårten Björkman and Danica Kragic

***Abstract*— Object grasping and manipulation pose major challenges for perception and control and require rich interaction between these two fields. In this paper, we concentrate on the plethora of perceptual problems that have to be solved before a robot can be moved in a controlled way to pick up an object. A vision system is presented that integrates a number of different computational processes, e.g. attention, segmentation, recognition or reconstruction to incrementally build up a representation of the scene suitable for grasping and manipulation of objects. Our vision system is equipped with an active robotic head and a robot arm. This embodiment enables the robot to perform a number of different actions like saccading, fixating, and grasping. By applying these actions, the robot can incrementally build a scene representation and use it for interaction. We demonstrate our system in a scenario for picking up known objects from a table top. We also show the system's extendibility towards grasping of unknown and familiar objects.**



(a) Armar III Stereo Head. (b) Kuka Arm and Schunk Hand.

Fig. 1. Hardware Components of the Grasp Inference System

I. Introduction

The process of grasping an object both in humans and robots is a research topic that opens up the possibility to study many related sub-problems. One of the open question is how to grasp a specific object. Inspired by the theory of affordances [1], different object centered strategies have been proposed. If the object is known, an already known action can be applied to it [2], [3], [4], [5]. If the object is similar to a known object, experience can be re-used for grasp synthesis [6], [7], [8], [9]. An unknown object needs to be analyzed in terms of its 3D structure and other physical properties from which a suitable grasp can be inferred [10], [11], [12], [13].

Object grasping in realistic scenarios requires more than a pure decision of where to put fingers: it requires a whole set of processing steps whose purpose is to achieve an understanding of the scene that a robot is facing, thus obtaining object hypotheses. Once this is done, the aforementioned approaches can be exploited. The general requirement for a system that implements such a grasping process is robustness in a real world situation without too many assumptions.

In this paper, we present a vision system that integrates different computational processes to incrementally build a scene representation suitable for object grasping and manipulation. The hardware components are the Armar III robotic head [14] and a 6 DoF Kuka arm [15] equipped with a Schunk Dexterous Hand 2.0 (SDH) [16] as shown in Figure 1. This embodiment enables the robot to perform a number of actions like saccading, fixating, and grasping. By applying these actions, the robot can extend its internal scene representation as well as interact with the environment. Compared to similar embodied active vision systems, our main contribution consists of an offline hand-eye calibration procedure whose accuracy enables us to produce dense stereo reconstructed point clouds of segmented objects. We will demonstrate the proposed system in a table-top scenario containing known objects. Furthermore, an outlook will be given on how the same integrated active vision system can be used for a more complex task like picking up unknown objects.

The structure of the paper is as follows. After discussing the related work in the next section, we give a general system overview in Section III. This is followed by a more detailed presentation of the individual modules. Qualitative results for the scenario of picking up known objects are shown in Section IV.

The authors are with the Computer Vision and Active Perception Lab, Centre for Autonomous Systems, Royal Institute of Technology, Stockholm, Sweden, e-mail: javiergm,bohg,celle,danik@kth.se

II. Related Work

Vision based grasping of known objects has been studied a lot. Here, we will focus on the work by Huebner et. al [3] and Ude et. al [17] who are using a similar robotic platform to ours [4] including an active head. In [3], the Armar III humanoid robot is enabled to grasp and manipulate known objects in a kitchen environment. Similar to our system, a number of perceptual modules are at play to fulfill this task. Attention is used for scene search. Objects are recognized and their pose estimated with the approach originally proposed in [5]. Once the object identity is known, a suitable grasp configuration can be selected from an offline constructed database. Here, a box-based approach is used in which the object shape is approximated by a constellation of boxes. Thereby, the number of candidate grasps for one

object is limited to just a few [10]. Visual servoing is applied to bring the robotic hand to the desired grasp position [18]. Different from our approach, absolute 3D data is estimated by fixing the 3 DoF for the eyes to a position for which a stereo calibration exists. The remaining degrees of freedom controlling the neck of the head are used to keep the target and current hand position in view. In our approach, we keep the eyes of the robot in constant fixation on the current object of interest. This ensures that the left and right visual field overlap as much as possible, thereby maximizing e.g. the amount of 3D data that can be reconstructed. It has also been shown that some cues, like shape and motion can be easier derived in fixation [19], [20]. However, the calibration process becomes much more complex.

In the work by Ude et. al [17], fixation plays an integral part of the vision system. Their goal is however somewhat different from ours. Given that an object has been already placed in the hand of the robot, it brings it in front of its eyes and rotates it in a controlled movement. By this it gains several views from the currently unknown object for extracting a view-based representation that is suitable for recognizing it later on. Different to our work, no absolute 3D information is extracted for the purpose of object representation.

In [21], the authors presented a method for calibrating the active stereo head. The correct depth estimation of the system was demonstrated by letting it grasp an object held in front of its eyes. No dense stereo reconstruction has been shown in this work.

Similarly, in [22] a procedure for calibrating the Armar III robotic head was presented. Our calibration procedure is similar to the one described in those papers, with a few differences. We extend it to the calibration of all joints, thus obtaining the whole kinematic chain. Also, the basic calibration method is modified to use an active pattern instead of a fixed checkerboard, which has some advantages that we outline below.

III. System Architecture

In this section, we provide a system overview. Its individual building blocks are described in more detail below. An overview of the system is shown in Figure 2.

First, there are processes for the purpose of incrementally building up a scene representation. This representation contains the detected table plane, the position of the Kuka robot arm relative to the robotic head and a number of detected object hypotheses.

The emergence of these hypotheses is triggered by the visual exploration of the scene with the ARMAR III robotic head [14]. It has 7 DoF and is equipped with two stereo camera pairs, a wide-angle and a narrow-angle one. The former are used for peripheral vision in which scene search can be performed. This is done by computing a saliency map and assuming that maxima in this map are initial object hypotheses, [4]. A saccade is performed to a maxima such that the stereo camera with the narrow-angle lenses center on the potential object. Once the system is in fixation, a disparity map is calculated and segmentation performed [23].

The result of this part of the system is a geometric model of the scene. Assuming a tasks such as to clean the table, different grasping strategies can be applied based on the available knowledge about the object. In this paper, we will demonstrate grasping of known objects. For this purpose, we assume a database containing scale invariant features (SIFT) [24], color co-occurrence histograms (CCH) [25], an approximate shape model and grasps for a number of objects. For deciding whether an object hypothesis is a specific object, SIFT and CCH based recognition is performed. The approximate shape model and pre-defined grasp help to decide the exact arm and hand configuration to pick up the object. Visual servoing is then used to guide the arm and hand to the correct pre-grasp and final grasping position.

A. Offline Calibration

Though the use of visual servoing for grasping allows for reasonable results with limited calibration, it is still desirable to have the cameras accurately calibrated. Stereo calibration is necessary for image rectification prior to calculating the disparity map. Head-eye calibration is necessary for performing the saccade that brings the detected attention point to the center of the narrow-field cameras. And finally, some hand-eye calibration is necessary for the visual servoing method that we are using. This leaves us with three kinds of transformations that we need to determine: (i) the transformation between the left and the right camera coordinate systems for a given configuration of the joints; (ii) the transformation between one camera system in two different joint configurations; and (iii) the transformation between the camera coordinate system and the arm coordinate system.

1) Stereo Calibration: One of the most commonly used methods for finding the transformation between two camera coordinate systems is the use of a checkerboard which is observed by two cameras (or the same camera before and after moving) [26]. The checkerboard defines its own coordinate system. By detecting the intersection between the squares, it is possible to find the transformation from this coordinate system to the left and right camera coordinate system. Once we have this transformation, it becomes easy to obtain the transformation between the left and right camera coordinate systems.

For the purposes of our experiment, we used a modified version of this method. Instead of using a static checkerboard pattern, we used a small LED rigidly attached to the end effector of the robotic arm, which we moved describing a certain pattern. Because of the accuracy and repeatability of the KUKA arm (< 0.5mm) and the sub-pixel precision for the detection of the LED in camera space, we can obtain results that are at least as accurate as those obtained by the use of traditional checkerboard patterns, with several advantages:

- Instead of using an arbitrary checkerboard coordinate system as the intermediate coordinate system, we can use the arm coordinate system. So at the same time that we are performing the stereo calibration, we are obtaining the hand-eye calibration for free.

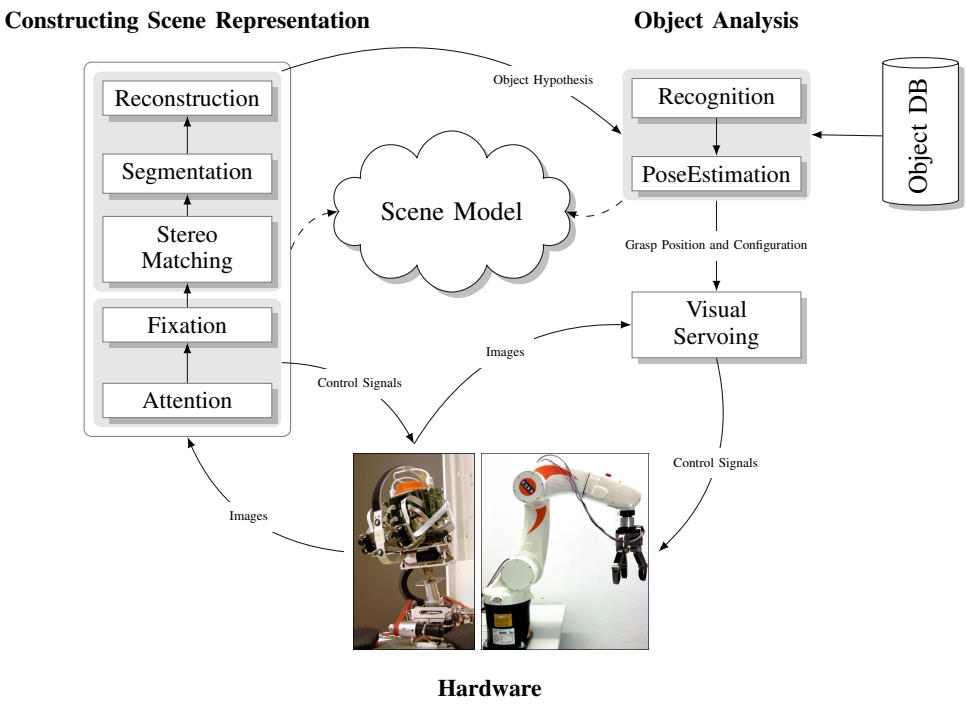


Fig. 2. Overview of the system.

- With a static pattern, it is necessary to use the same checkerboard at the same position for the two camera coordinate frames for which we are trying to obtain the transformation. This means that the pattern must be within the field of view for the two poses of the camera, which may be difficult when the two poses are not similar. With our approach, it is not necessary to use exactly the same end effector positions for the two camera coordinate systems, since any set of points will allow us to obtain the transformation between the arm coordinate system and the camera coordinate system.
- For these same reasons, we found empirically that this approach makes it possible to choose a pattern that offers a better calibration performance. For example, by using a set of calibration points that is uniformly distributed in image space (as opposed to world space, which is the case for checkerboard patterns), it is possible to obtain a better characterization of the distortion parameters of the lenses. To create this kind of pattern, we bring the LED to the center of the foveal image. By moving the LED to a series of points that are on a plane perpendicular to the principal axis of the camera, we form a uniform and comprehensive pattern in image space of the foveal cameras. We repeat this process at different distances from the camera, and we end up with a collection of calibration points that effectively fill the image space and a range of depths. The shape of this calibration pattern forms a truncated pyramid in world space.

2) Head-eye Calibration: For a static camera setup, the calibration process would be completed here. However, our vision system can move to fixate on the objects we manipulate. Therefore it is important to obtain a calibration that remains valid after these movements. The Armar III head has 7 DoF: three for the neck, two common tilt joints, and one for the pan of each camera. Only these last two joints allow for independent movement of one camera with respect to the other, so they are the only ones which will affect the stereo calibration.

Ideally, it would be possible to obtain the exact transformation between the camera coordinate system before and after moving a certain joint just from the known kinematic chain of the robot and the readings from the encoders. However, inaccuracies arise in the manufacturing process influencing the true center and axis of joint rotations and in the discrepancy between motor encoder readings and actual angular joint movement. There are also some repeatability issues that cannot be dealt with by means of offline calibration. The use of visual servoing as described in Section III-D.3 and online calibration as described in Section III-B.2 tries to reduce the impact of these. However, we have found that our method provides an acceptable estimation of the hand-eye and head-eye calibration at all times.

Our method consists of performing the following steps for each of the joints:

1) Choose two different positions of the joint, that are far enough apart to be significant, but with an overlapping viewing area that is still reachable for the robotic arm.
2) For each of these two positions, perform the static calibration process as described above, so that we obtain the transformation between the arm coordinate system and each of the camera coordinate systems.

(a) Left Wide Field Camera. (b) Saliency Map on Left Wide Field Camera.

Fig. 3. Example Output for Attention Process on Wide Field Images.

3) Find the transformation between the camera coordinate systems in the two previously chosen joint configurations. This transformation is the result of rotating the joint around some roughly known axis due to mechanical inaccuracies, with a roughly known angle from the motor encoders. From the computed transformation, we can then more exactly determine this axis, center and angle of rotation.

After performing this process for all the joints, we can use the result to rebuild the kinematic chain of the head, in a way that takes into account the deviations in the axis and centers of rotation resulting from the manufacture process.

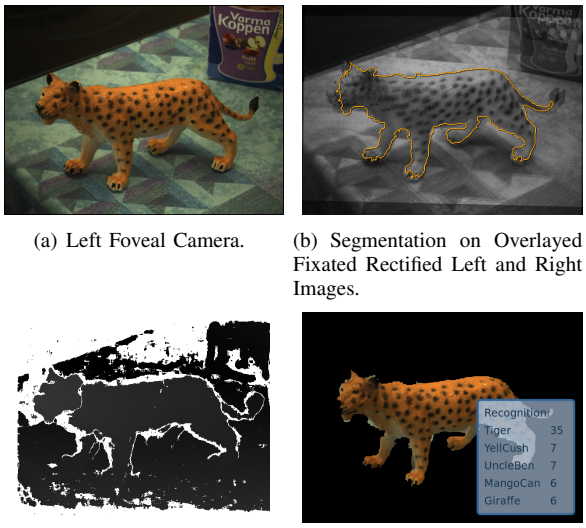
B. Building up the Scene Representation

In the following section, we briefly present the computational modules needed for forming a model of the scene (see left part of Figure 2). For a more detailed description, we refer to our previous work [4], [23], [19].

1) Attention: As mentioned in Section III, our vision system consists of two stereo camera pairs, a peripheral (Figure 3(a)) and a foveal one (Figure 4(a)). Scene search is performed in the wide-field camera by computing a saliency map on it. An example for such a map based on the Itti & Koch Saliency model [27] is given in Figure 3(b). Peaks in this saliency map are used to trigger a saccade of the robot head such that the foveal cameras are centered on this peak.

2) Fixation: When a rapid gaze shift to a salient point in the wide-field is completed, the fixation process is immediately started. The foveal images are initially rectified using the camera parameters obtained from the offline calibration described in Section III-A. This rectification is then refined online by matching Harris' corner features extracted from both views and computing an affine essential matrix. The resulting images are then used for stereo matching [28]. A disparity map on the foveal image in Figure 4(a) is given in Figure 4(c). The vergence angle of the cameras is controlled such that the highest density of points close to the center of the views are placed at zero disparity.

3) Segmentation: For 3D object segmentation, we use a recent approach [23] that relies on three possible hypotheses: figure, ground and a flat surface. The commonly made flat surface assumption simplifies the problem of segregating an object from the surface it stands on, when both are very similar in appearance.



(a) Left Foveal Camera. (b) Segmentation on Overlayed Fixated Rectified Left and Right Images.

(c) Disparity Map. (d) Segmented Image and Recognition Result. Five best matching objects are shown in the table.

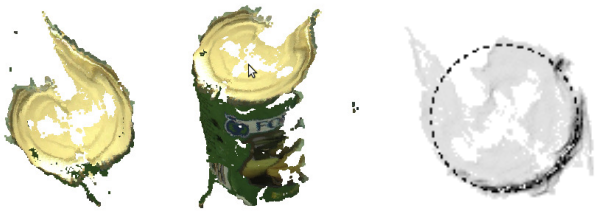
Fig. 4. Example Output for Processes running on Foveal Images.

The segmentation approach is an iterative two-stage method that first performs pixel-wise labeling using a set of model parameters and then updates these parameters in the second stage. In our case, these parameters are color and disparity information. Model evidence is summed up on a per-pixel basis using marginal distributions of labels obtained with belief propagation. For initialization, we place an imaginary 3D ball around the fixation point and label everything within the ball as foreground. RANSAC [29] is used for finding the most dominant plane. Points that belong to it are labeled as table. The remaining points are initially labeled as background.

4) Re-Centering: The attention points coming from the saliency map on wide-field images tend to be on the border of objects rather than on their center. Therefore, when performing a gaze shift, the center of the foveal images does not correspond to the center of the objects. We perform a re-centering operation to account for this. This is done by letting the iterative segmentation process stabilize for a specific gaze direction of the head. Then the center of mass of the segmentation mask is computed. A control signal is sent to the head to correct its gaze direction such that the center of the foveal images is aligned with the center of segmentation. After this small gaze-shift has been performed, the fixation and segmentation process is started again until the center of the segmentation mask is sufficiently aligned with the center of the images.

An example for the resulting segmentation is given in Figure 4(b), in which the object boundaries are drawn on the overlayed left and right rectified images of the foveal cameras. The segmented point cloud calculated from the segmented disparity map is depicted in Figure 5.

(a) Point cloud of tiger and fitted bounding box



(b) Point cloud of mango can and fitted cylinder

Fig. 5. 3D Point Cloud (from two Viewpoints) and Estimated Object Pose generated from Disparity Map and Segmentation in Figure 4(c) and 4(b).

C. Grasping Known Objects

Once a scene model with a number of object hypotheses has been obtained, they can be further analyzed. In this paper, we consider the problem of cleaning a table containing an unknown number of known objects in an unknown configuration. To be able to remove them from the table, we need to determine their identity and pose.

1) Recognition: The identity of an object hypothesis is sought in a database of 25 known objects. Two complementary cues are used for recognition; SIFT and CCH. The cues are complementary in the sense that SIFT features rely on objects being textured, while CCH works best for objects of a few, but distinct, colors. An earlier version of this system, including a study on the benefits of segmentation for recognition, can be found in [19]. Here, the total recognition score is the product of the number of matched SIFT features and the CCH correlation score. An example of the recognition results is shown in Figure 4(d).

2) Registration: If an object is identified and it is known to be either rectangular or cylindrical, the pose is estimated using the cloud of extracted 3D points projected onto the 2D table plane, with object dimensions given by a lookup in the database. For rectangular objects a dominating plane is initially sought using random sampling of 3D points (RANSAC) and least median optimization. The orientation of this plane gives an initial estimate of the orientation and position of the object on the table. Next the pose estimate is improved by minimizing the sum of absolute errors from each individual object point projected onto the table to the closest edge of the rectangular model (see Figure 5(a) for an example). This is efficiently done using a 2D distance map and gradient descent. Similarly, the position of a cylindrical object is determined through random sampling and least median optimization, followed by distance map based fitting, using its known radius and the assumption that the object is standing upright. An example for a fitted cylinder is shown in Figure 5(b).

3) Choosing a Grasp Configuration: This paper focuses on the perceptual problems that arise prior to the actual picking up of an object. We therefore simplified the search for possible grasp configurations to only those ones from the top. Given an object identity and a pose of the object, we can determine the desired pre-grasp shape, position and wrist orientation of the end effector.

The pre-grasp shape is chosen dependent on whether the object is rectangular or cylindrical. In the former case, we choose a pinch grasp with the thumb opposing the two other fingers. For a cylindrical object, the rotation between the two neighboring fingers of the SDH is increased to achieve a spherical pre-grasp shape. The position of the hand is influenced by the known height of the object and its position on the table. This will be clarified further below in Section III-D.2.

The wrist orientation is dependent on the estimated object pose. If it is rectangular, then the vector between the thumb and the two opposing fingers is aligned with the minor axis of the rectangle. If it is spherical, then the wrist orientation does not matter and we use a predefined wrist orientation.

D. Moving the arm

The previous components provide all the information needed to initiate the actual grasping process. This information has all been obtained from visual input, and is thus expressed in the camera coordinate system. This makes it adequate to use a visual servoing approach.

1) Tracking the hand: In a pure visual servoing approach, we would estimate the position and orientation of one or more relevant parts of the hand (e.g. the fingertips) and use the control loop to bring them to the grasping points, which would also be visually detected. This would require a complete tracking of the robotic hand. We simplify this complex problem by choosing an LED near the wrist of the hand as a single point that can be robustly tracked. This is similar to the approach in [18] in which a red marker ball mounted on the wrist of the robotic arm is used as a reference.

2) The grasping point: As mentioned before, we are only considering top grasps at the current iteration of the system, so the trajectory of the arm (not including the wrist orientation joint) can be totally determined by a single point, the point around which the hand closes. For the visual servoing implementation, we need to determine this grasping point in image space. To do that, we take into account that after the iterative fixation, segmentation and re-centering process, the center of the object will be in the center of each image in both wide-field cameras. Having recognized the object, we also know its height from a database look-up. From this information, we can determine the position of the hand above the grasping point. From the kinematic

calibration, we can transform the center of the image to the head coordinate system, add the height correction and transform it back to image coordinates.

3) Visual servoing: Following the same procedure we used to add the height correction, we add the distance from the LED (we use as a reference) to the position above the grasping point, which provides us with the target position for the LED.

We then use an image-based visual servoing control scheme to move the hand in the direction that reduces the distance between the LED and the target position. Our approach is a simplified version of the method presented in [18] in which more than just top grasps are considered. For determining the target position of their reference point in image space also the orientation of the end effector as read from the motor encoders is taken into account.

4) The whole process: The visual servoing stage allows us to increase the accuracy of the process by using visual feedback to position the hand prior to grasping. Thereby the inaccuracies associated to calibration and mechanical error are circumvented. The complete grasping process includes the following steps:

1) Move the arm to a position and orientation that renders the LED visible in both cameras.
2) Use visual servoing to position the hand above the grasping point as shown in Figure 6(a).
3) Rotate the arm to the wrist orientation provided by the pose estimation. See Figure 6(b).
4) Move the hand down in a vertical movement, until it reaches the grasping point.
5) Close the hand using the selected pre-shape as shown in Figure 6(c).
6) Lift the object, using another vertical movement, and move it away to some pre-established destination position. where the hand is opened again and the object dropped. Examples of this are shown in Figure 6(d) to 6(f)

IV. Experiments

In this paper, we are demonstrating the presented system fulfilling the task of grasping known objects from a table top. We will only show qualitative results in form of a video [30] from a whole run. Stills of this run are given in Figure 6.

Quantitative results as for example the evaluation of accuracy of the calibration remain future work.

A. Setup

An example for the experimental setup including robotic arm and head can be seen in Figure 6(a). Here a subset of 4 objects from the 25 known ones has been selected and randomly spread on the table. Since we are not using closed loop grasp execution in this system, we make the assumption that objects are well separated on the table. Thereby we ensure that the hand is not colliding with an object while picking up its neighbor. Furthermore, we assume that cylinders are standing upright.

B. Results

1) Calibration: In Figure 5 we have already shown segmented point clouds that were the results from stereo matching on rectified images. The rectification was achieved by bootstrapping the online calibration with initially rectified images based only on calibration parameters from the offline calibration.

In Figure 7, we show two typical example for point clouds comprising a whole scene. They are merged from separate point clouds that were reconstructed from different view points (five in these cases). This is an illustration of the accuracy of the offline head-eye calibration process.

2) Grasping: As we can see in the video, the system is consistently able to perform its task in the relatively controlled environment we are using. There are some circumstances under which the grasp process may fail. For example, it is difficult to fixate on cylinders, because there is not a dominant plane in disparity space. This can cause the grasping position to be slightly off.

Our segmentation approach has been shown to be robust even if the object is moving and kept in fixation during that movement, [20]. However, the most critical phase for any iterative scheme for figure-ground segmentation is initialization. In our case, this was dependent on the fixation point. If the gaze shifts to a point that is too close to the border of the object, parts of the background will be initialized as foreground. Then it might happen that the system tries to segment the background instead of the object, which makes the segmentation wander away from the object.

This has not proven to be a significant problem for the normal operation of the system. However, as future work, we aim at performing a quantitative evaluation of each of the components of the system and thereby of the whole grasping process.

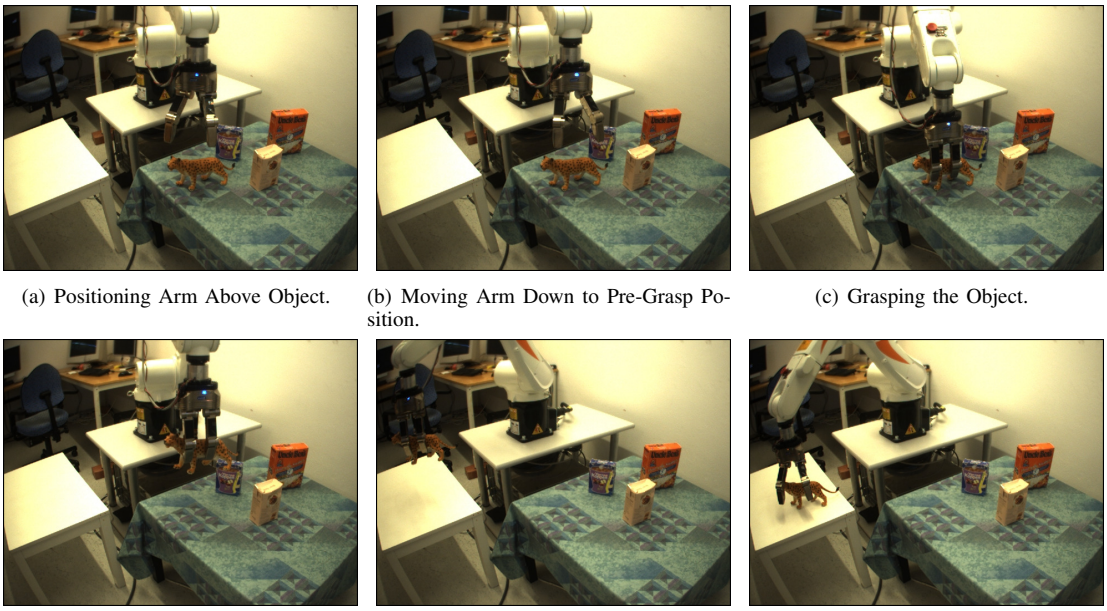
V. Conclusion and Future Work

In this paper, we presented an active vision system capable of forming a 3D scene representation and picking up known objects.

Several assumptions have been made, e.g., that the objects belong to a set of known objects or that they are standing on a flat surface. The modular design of the system allows for the replacement of individual parts with functionally equivalent modules that can remove some of these assumptions.

The pose estimation described in Section III-C.2 is simple and effective in our scenario, but introduces several assumptions. Objects are for example approximated with either boxes or cylinders in which cylinders are assumed to be always standing upright. Since we are able to densely reconstruct point clouds, approaches for pose estimation like [31] are feasible. This would offer more flexibility in terms of object pose and more robustness against object occlusion. However, exact models of the known objects are needed.

Grasping of unknown objects is a much more challenging task compared to grasping known objects. Several research groups have considered this problem and usually take the 3D



(a) Positioning Arm Above Object. (b) Moving Arm Down to Pre-Grasp Position. (c) Grasping the Object.

(d) Lifting Object. (e) Moving Object to Target Position. (f) Release Object at Target Position.

Fig. 6. Example Grasp for the recognized Tiger. The whole table being emptied is shown in [30].

structure of an object hypothesis into account for inferring a grasp, [10], [11], [13]. In our vision system, none of the modules that are responsible for building up the scene representation are dependent on any prior knowledge on objects (see Figure 2). Therefore, this representation is general enough to serve as an input to methods like [10], [11] that rely on segmented point clouds.

As we explained in Section III-D.3, using a simple marker such as the LED for visual control provides good results for this system. However, it is a simplification in which the LED is not allowed to be occluded or out of view. Furthermore, it does not scale well to situations where a more precise control over the grasp process is needed. A significant improvement would be to estimate the position and orientation of the robotic hand. Together with the known hand geometry, this would allow us to know the position, in image space, of the relevant parts of the hand, and use a pure visual servoing approach to bring these parts to the desired points.

The accuracy of the offline calibration process makes it possible to integrate the point clouds obtained from different saccades into a large point cloud that represents the whole scene (see Figure 7 for an example). This point cloud can be fed into a simulator, which would then be used to perform grasp, path and trajectory planning.

ACKNOWLEDGEMENTS

This work was supported by the EU through the project PACO-PLUS, IST-FP6-IP-027657, and GRASP, IST-FP7-IP-215821 and the Swedish Foundation for Strategic Research.

REFERENCES

[1] J. Gibson, "The Theory of Affordances," in *Perceiving, Acting, and Knowing: Toward an Ecological Psychology*, R. Shaw and J. Bransford, Eds. Erlbaum, NJ, 1977, pp. 67–82.

[2] J. Glover, D. Rus, and N. Roy, "Probabilistic Models of Object Geometry for Grasp Planning," in *IEEE International Conference on Robotics and Automation*, Pasadena, CA, USA, May 2008.

[3] K. Huebner, K. Welke, M. Przybylski, N. Vahrenkamp, T. Asfour, D. Kragic, and R. Dillmann, "Grasping Known Objects with Humanoid Robots: A Box-based Approach," in *International Conference on Advanced Robotics*, 2009, pp. 1–6.

[4] B. Rasolzadeh, M. Björkman, K. Huebner, and D. Kragic, "An Active Vision System for Detecting, Fixating and Manipulating Objects in Real World," *Int. J. of Robotics Research*, 2009, to appear.

[5] P. Azad, T. Asfour, and R. Dillmann, "Stereo-based 6D Object Localization for Grasping with Humanoid Robot Systems," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2007, pp. 919–924.

[6] J. Bohg and D. Kragic, "Learning Grasping Points with Shape Context," *Robotics and Autonomous Systems*, 2009, in Press.

[7] A. Saxena, J. Driemeyer, J. Kearns, and A. Y. Ng, "Robotic Grasping of Novel Objects," *Neural Information Processing Systems*, vol. 19, pp. 1209–1216, 2007.

[8] J. Speth, A. Morales, and P. J. Sanz, "Vision-Based Grasp Planning of 3D Objects by Extending 2D Contour Based Algorithms," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008, pp. 2240–2245.

[9] M. Stark, P. Lies, M. Zillich, J. Wyatt, and B. Schiele, "Functional Object Class Detection Based on Learned Affordance Cues," in *6th International Conference on Computer Vision Systems*, ser. LNAI, vol. 5008. Springer-Verlag, 2008, pp. 435–444.

[10] K. Huebner and D. Kragic, "Selection of Robot Pre-Grasps using Box-Based Shape Approximation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems.*, 2008, pp. 1765–1770.

[11] M. Richtsfeld and M. Vincze, "Grasping of Unknown Objects from a Table Top," in *ECCV Workshop on 'Vision in Action: Efficient strategies for cognitive agents in complex environments'*, Marseille, France, September 2008.

[12] D. Aarno, J. Sommerfeld, D. Kragic, N. Pugeault, S. Kalkan, F. Wörgötter, D. Kraft, and N. Krüger, "Early Reactive Grasping with Second Order 3D Feature Relations," in *ICRA Workshop: From Features to Actions*, 2007, pp. 319–325.

[13] N. Bergström, J. Bohg, and D. Kragic, "Integration of Visual Cues for Robotic Grasping," in *Computer Vision Systems*, ser. Lecture Notes in Computer Science, vol. 5815. Springer Berlin / Heidelberg, 2009, pp. 245–254.

[14] T. Asfour, K. Welke, P. Azad, A. Ude, and R. Dillmann, "The Karlsruhe Humanoid Head," in *IEEE/RAS International Conference on Humanoid Robots (Humanoids)*, Daejeon, Korea, December 2008.

Fig. 7. Two examples (one each row) for a point cloud representing a whole scene merged from different view points. Left Column: View from the side. Right Column: View from the top.

[15] KUKA, "KR 5 sixx R850," www.kuka-robotics.com, Last Visited 2009.

[16] SCHUNK, "Sdh," www.schunk.com, last visited 2009.

[17] A. Ude, D. Omrcen, and G. Cheng, "Making Object Learning and Recognition an Active Process," *Int. J. of Humanoid Robotics*, vol. 5, no. 2, pp. 267–286, 2008.

[18] N. Vahrenkamp, S. Wieland, P. Azad, D. Gonzalez, T. Asfour, and R. Dillmann, "Visual Servoing for Humanoid Grasping and Manipulation Tasks," in *IEEE/RAS Int. Conf. on Humanoid Robots (Humanoids)*, 2008, pp. 406–412.

[19] M. Björkman and J.-O. Eklundh, "Vision in the real world: Finding, attending and recognizing objects," *International Journal of Imaging Systems and Technology*, vol. 16, no. 5, pp. 189–209, 2006.

[20] M. Björkman and D. Kragic, "Active 3D Segmentation through Fixation of Previously Unseen Objects," in *British Machine Vision Conference*, September 2010, to appear.

[21] A. Ude and E. Oztop, "Active 3-d vision on a humanoid head," in *Int. Conf.on Advanced Robotics (ICAR)*, Munich, Germany, 2009.

[22] K. Welke, M. Przybylski, T. Asfour, and R. Dillmann, "Kinematic calibration for saccadic eye movements," Institute for Anthropomatics, Universität Karlsruhe, Tech. Rep., 2008.

[23] M. Björkman and D. Kragic, "Active 3D Scene Segmentation and Detection of Unknown Objects," in *IEEE International Conference on Robotics and Automation*, 2010.

[24] D. Lowe, "Object recognition from local scale-invariant features," in *IEEE Int. Conf. on Computer Vision*, September 1999, pp. 1150–1157.

[25] T. Gevers and A. Smeulders, "Colour based object recognition," *Pattern Recognition*, vol. 32, pp. 453–464, March 1999.

[26] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 1330–1334, 2000.

[27] L. Itti and C. Koch, "Computational modeling of visual attention," *Nature Reviews Neuroscience*, no. 2, pp. 194–203, 2001.

[28] G. Bradski and A. Kaehler, *Learning OpenCV*. O'Reilly Media Inc., 2008. [Online]. Available: http://oreilly.com/catalog/9780596516130

[29] M. Fischler and R. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Comm. of the ACM*, vol. 24, pp. 381–395, June 1981.

[30] X. Gratal, J. Bohg, M. Björkman, and D. Kragic, "Scene representation and object grasping using active vision," Movie, http://www.csc.kth.se/\~{}bohg/2010_IROS_WS.mpg.

[31] C. Papazov and D. Burschka, "Stochastic Optimization for Rigid Point Set Registration," in *5th International Symposium on Visual Computing (ISVC)*, Las Vegas, Nevada, USA, December 2009, to appear.