Informed Active Perception with an Eye-in-hand Camera for Multi Modal Object Recognition

Dennis Stampfer, Matthias Lutz and Christian Schlegel

***Abstract*— For service robots that work in an everyday environment, reliable perception of the environment, its objects and their properties is a mandatory prerequisite.**

However, this is still a challenge. Commonly used approaches for object recognition are passive and will never be able to definitely identify objects in all cases. We propose an approach of active perception for object recognition by systematic acquisition of new and previously hidden information. This is an efficient and reliable way to improve perception abilities and overcome weaknesses of passive approaches.

We make use of cues of an initial recognition in the full scene and then systematically move a camera mounted on a manipulator around an object to acquire new information. We describe the overall recognition system in a real scenario using contextual knowledge for the purpose of object recognition with multiple algorithms and views.

The approach is demonstrated in real-world experiments with a service robot acting as butler. It uses active perception to distinguish similar objects and to get additional object properties such as their filling level or current mode of devices (e.g. coffee machine).

I. INTRODUCTION

Typical tasks of autonomous mobile service robots include mobile manipulation in environments made for humans, not for robots. Therefore it is a mandatory prerequisite to perceive the environment and especially objects and their properties the robot has to work with. This research topic has been recognized and is being addressed in several competitions (e.g. [1]). Although a huge variety of methods for object recognition is available, it is still an open challenge. The environment is unstructured, complex and regardless of the robots abilities and goals. The robot therefore has to deal with the challenges that the real world defines. In such environments, methods for everyday use have to be robust, efficient and aware of (limited) resources.

Perception capabilities are limited by the sensors reach. Objects may be visible only partially due to occlusion, may be too far away or the important information to recognize an object is printed on its back and thus hidden.

Many approaches rely on a single method or algorithm for object recognition, but it requires the skillful combination and integration of a variety of available approaches. This requires new methods to interpret the output of the existing approaches in order to combine them.

The authors are with the University of Applied Sciences Ulm, Department of Computer Science, Prittwitzstr. 10, 89075 Ulm, Germany. {stampfer,lutz,schlegel}@hs-ulm.de. This work has been conducted within the *ZAFH Servicerobotik* (http://www.zafh-servicerobotik.de/). The authors gratefully acknowledge the research grants of the state of Baden-Württemberg and the European Union.

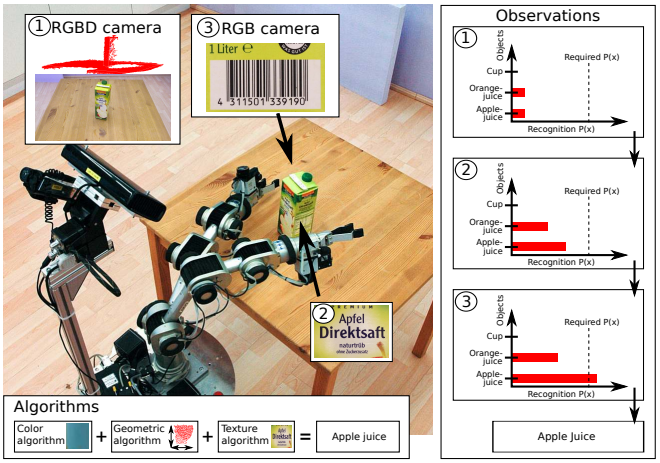

Fig. 1. A service robot in an everyday environment. The initial object recognition on the full scene (1) brings cues about objects in the scene. The cues are then exploited to actively find new information (2), (3) about the objects using a camera mounted on a manipulator and multiple object recognition algorithms. This stepwise refines the recognition probability.

As service robots for mobile manipulation are equipped with manipulators, we propose an approach that combines action and perception to make the otherwise passive process of object recognition an active one (fig. 1).

An active recognition process particularly has to be resource efficient. Mobile manipulation requires a lot of resources due to its complexity. The robot therefore has to carefully evaluate the effort and reward of actions to be taken by weighting them. In this process, the robot systematically obtains more information about the environment to enhance its world knowledge. This is done by taking into account the context of the current task and current knowledge as well as prior knowledge about the environment and objects. Throughout the whole process, we use different sensor data, algorithms and views in combination with the help of probabilistic methods for robust object recognition.

At a glance, our approach runs object recognition on a scene image to get a first cue of objects in the scene. Using prior knowledge, these cues are then exploited to generate an efficient and informed behavior in which an eye-in-hand camera mounted on a manipulator systematically inspects objects to extract further cues (systematic object inspection). These semantic cues found during exploration are fused probabilistically to a final recognition result.

This paper combines the approaches of [2], [3] and integrates them into one scenario. We take advantage of active perception to get properties such as filling level or

current mode of a device and describe how the overall object recognition system works, putting focus on the active perception part. We address the topic of object recognition using multiple algorithms, selection of viewpoints for acquisition of new information based on an expected utility, collision free placement of an eye-in-hand camera and the integration into the overall system architecture.

The experiments show a butler robot in a home environment to prepare coffee and deliver similar appearing objects.

II. RELATED WORK

In active perception for object recognition, the next best viewpoint has to be determined. The authors in [4], [5] find viewpoints by considering the geometry of objects and select their views on the basis of visible surfaces. Our work considers features as basis for the determination of the next viewpoints as they are an information about what is on a surface. This enables us to evaluate the benefit of viewpoints with respect to the expected classification quality.

In contrast to our work, viewpoint planning is often used for model learning [6]. We know the object model and find the next best view for recognition by taking the object properties and environment constraints (e.g. occlusions) into account. Learned models and views, however, can be used in our approach as input for viewpoint selection.

In [7], a robot recognizes objects placed on a table by driving around it on a circular path. Object hypotheses from different viewpoints are integrated. They do not consider different kinds of features or algorithms for viewpoint selection. Observing the scene on a circular path is still limited compared to an eye-in-hand camera.

The use of foveal cameras [8] brings data of higher quality (close-up, high-res image) but no new additional data since the perspective stays the same.

In [9], the authors modify the environment for the purpose of perception. They segment cluttered scenes by pushing objects and observing the generated motion. In [10], objects are grasped and rotated to find a barcode without any knowledge about where it may be. However, modifying the environment for object recognition is not always necessary or even possible. We propose our approach as an option to consider before such complex actions are taken.

The problem of object search [11] differs from our problem such that objects have to be found rather than recognized. The shared problem between object search and this paper is where to look next. The search space in object search is the complete environment (room, building). In our approach we only focus on the area surrounding a single object where we need to find features for recognition. Our work can be applied after object search for reliably identification.

For object recognition itself, impressive approaches like MOPED [12] or algorithms in the Point Cloud Library [13] exist. It is our goal to combine algorithms to use their strengths. The authors in [14] combine the output of algorithms for color, depth and texture features for object recognition based on histograms. The common output of algorithms is a distance measure of histogram comparisons.

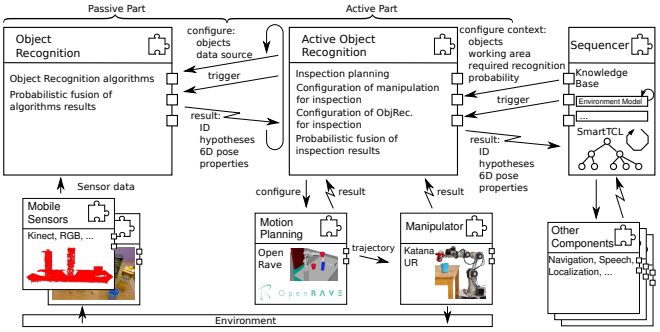

Fig. 2. The system view of the approach showing relevant components for object recognition and their interactions. It can be grouped in passive and active part of object recognition and system coordination.

The object identification is done using k-Nearest Neighbor on the feature level but not by a higher level probabilistic fusion of algorithm results.

III. SYSTEM OVERVIEW

From a system point of view, the individual parts are separated in components. Object recognition can be grouped in a *passive part* and *active part* (fig. 2). The current implementation uses SmartSoft [15], a component based approach for robotics software.

The *passive part* runs object recognition on the input of different sensors such as a RGBD camera or an eye-in-hand RGB camera. Multiple algorithms classify objects and estimate their pose. The results are fused probabilistically.

The *active part* manages the acquisition of new information of objects. It decides from where to look at objects, positions the sensors, triggers and configures the object recognition (*passive part*) to process the data from new perspectives and finally fuses the results of individual inspections as reported.

A knowledge base keeps a model of the environment which is updated based on the results of the recognition process. Regarding object recognition, it holds object names, instance identifiers, location and additional properties. The sequencer [16] uses the knowledge base and is responsible for the overall control and execution of the robots current task. It configures the object recognition and all other components in the system at runtime depending on the current context of the task.

For example, if we know to stand in front of a table, the *working area* to search for objects can be limited to the known height of the table. When in front of the dining table in the living room, no coffee machine is expected. When the task is to fetch a cup from the kitchen counter, no coffee machine must be recognized. We make use of this contextual knowledge and configure the object recognition to only recognize *objects* that are expected or relevant (encoded depending on task and location in the knowledge base).

The *required recognition probability* is a threshold until which objects are further inspected. This value could derived from a safety module, e.g. high probability for medicine.

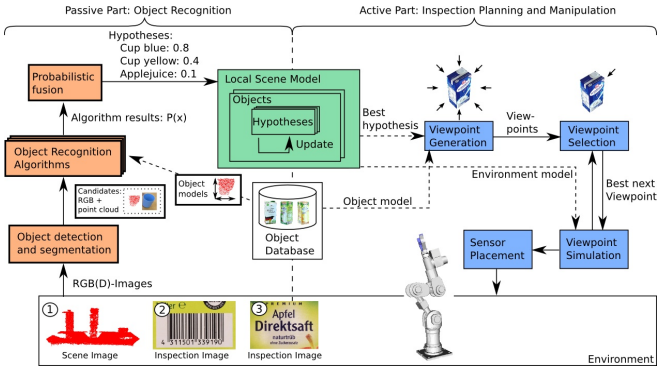

Fig. 3. The methodical view on the approach. An initial scene image (1) is segmented to get object candidates. These are processed using multiple algorithms and the results fused probabilistically. Based on these initial cues, viewpoints are generated, the best selected and the camera positioned. Images acquired this way (2), (3) are used as next input for object recognition.

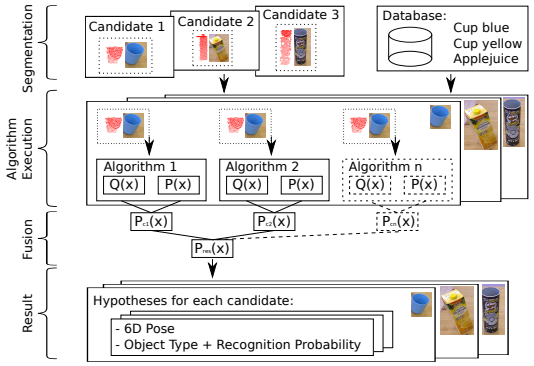

Fig. 4. Algorithm execution processes object candidates and matches them against the database. They output a recognition probability and pose which are fused to a final belief.

All these parameters depend on the current context (task, location, etc.). This is currently prior knowledge in the knowledge base (cf. [16]) and may also be determined at runtime. The configuration also enables the reduction of resources, since fewer algorithms need to be run and fewer inspections are necessary.

Besides classification of objects, the sequencer may require more information on the objects properties to finish the current task. It may therefore trigger the *active part* to inspect object properties. A property in the context of object inspection is a descriptive element of an object that can change during its existence and is thus not used for classification. This is for example the fill level (cups might be full or empty) or mode of a device (a coffee machine may be in standby, making coffee or needing refill). Therefore, the color of an object is not a property for property inspection but could be a feature used for classification that can be inspected. We realize this by also considering the properties during object training.

A methodical view on the object recognition process is given in fig. 3 and will be addressed in the following sections.

IV. PASSIVE PART: OBJECT RECOGNITION

Object recognition can take both RGBD images for initial recognition and RGB images for inspection as input (fig. 3, left). A first detection step segments the objects from the scene into object candidates, processes them in several algorithms, and fuses them to a final result. It is saved in the scene model which is shared with the active part.

A. Multiple algorithms for recognition

Multiple algorithms are run on each candidate (3D point cloud and cropped RGB image) for recognition and pose estimation. The algorithms use different features, e.g. color, texture or geometrical properties and match them to references from the database (fig. 4).

A simple custom feature based 3D model matcher is used to recognize objects based on their shape, a color histogram algorithm (standard OpenCV implementation) to consider the objects color and MOPED [12] for textured objects. Since household goods usually have text or barcode labels, two OCR and a barcode algorithm are used for inspection.

The set of algorithms to execute is determined at runtime depending on the recognition context (objects to recognize and input source). For example, OCR is only run on high-res eye-in-hand camera images for objects with text labels.

B. Probabilistic output and fusion

The algorithms output a probability $P(x)$ [2] that indicates the quality of recognition (fig. 4), i.e. the belief that the candidate is of type x. The recognition probability is a uniform and semantic interface which allows the combination of algorithms at the level of results. In contrast to combining results at the feature level, it improves the integration of existing algorithms since they can be extended to return a recognition probability.

In a last step, the results are probabilistically fused to a final hypothesis. Fusion considers the recognition probability $P(x)$ and combines it with the probabilistic algorithm quality $Q(x)$ [2]. $Q(x)$ states how well an algorithm is able to identify an object (e.g. color can not distinguish shapes).

V. ACTIVE PART: ACQUISITION OF NEW INFORMATION

Cues about object types from the local scene model are used to generate and select the next best views (fig. 3, right). The manipulator movement to place the camera is calculated, simulated and then executed. The object recognition is triggered again (fig. 2 and 3) with the eye-in-hand camera image as input. The result is a new hypothesis from an independent observation. It is probabilistically fused with existing ones and updates the scene model. This brings a stepwise enhancement of the recognition probability in each inspection step (fig. 1). The inspection of one object continues until the recognition probability reaches the previously configured threshold (*required recognition probability*, fig. 2) or until no further unvisited viewpoints exist.

A. Inspection planning

To generate viewpoints, the half sphere around the normal vectors of the object features (e.g. text or barcode labels) are

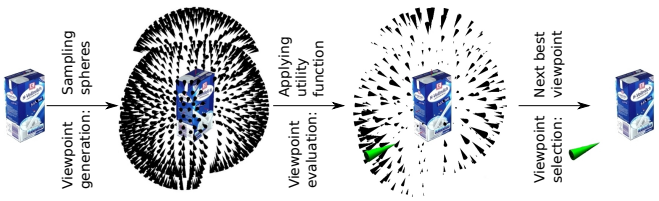

Fig. 5. Based on an initial cue of the object type, viewpoints are generated and a utility function is applied for evaluation. The next best viewpoint (green) in this example is directed at the reliable barcode.

Fig. 6. The eye-in-hand camera mounted on the manipulator (left) and a typical image captured by this camera during object inspection (right).

sampled based on the object type of the best hypothesis (fig. 5) [3]. A viewpoint is the 6D pose at which the camera is placed to look at the object.

A single viewpoint is not sufficient because it might not be possible to position the camera at that viewpoint due obstacles, kinematic constraints or no path to reach the position. They are regenerated as soon as the object type of the best hypothesis changes. This is necessary since viewpoints were generated based on the features of the previous hypothesis.

Viewpoints are chosen with the strategy to confirm the current hypothesis. The viewpoint promising the best recognition result considering the effort is the best one. For this purpose, viewpoints are evaluated by weighting cost and benefit for recognition which results in an utility value.

For the utility, we reuse the quality $Q(x)$ as benefit and use angular deviation to the surface normal and euclidean distance from current camera position to the viewpoint as costs. This can be extended to use manipulation time or real travel distance/time of the trajectory. The utility is formulated as the sum of these values (equally weighted, costs negative) and updated for each object inspection [3]. The viewpoint with the highest utility is the "best viewpoint" (fig. 5).

This method of inspection planning is also used for positioning the camera when properties have to be detected. Special property features are used for the generation of viewpoints.

B. Manipulation

Since the robot is not a passive observer anymore, it must be ensured that there are no collisions with the environment. OpenRave [17] is used for collision free manipulation planning to position the camera (no grasp planner is used). It loads the environment model from the local scene model (fig. 3). If an object is identified sufficiently enough, its exact 3D model from the object database is used for manipulation planning. Otherwise, the bounding box of the objects is

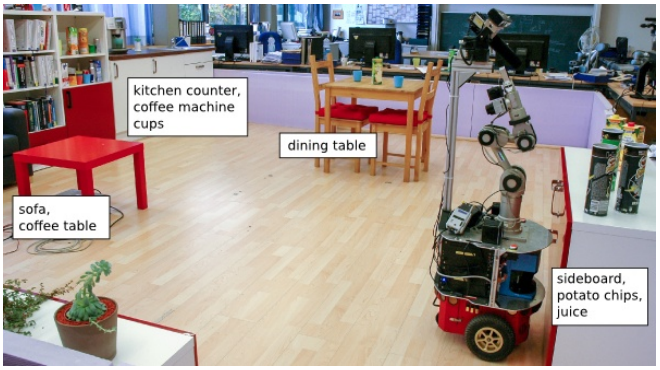

Fig. 7. The robot in a home environment while running Experiment II.

used for manipulation planning. This adds to the robustness required for everyday use, as 3D sensor data seen from one view is actually no full 3D measurement.

If simulating the camera movement fails (e.g. configuration collides with obstacle), the viewpoint is dropped and the next best is tried. Otherwise, the path is planned and the camera is positioned. After successful recognition, other viewpoints pointing at the same feature are discarded assuming that they will not enhance the recognition since the best one was already used.

VI. EXPERIMENTS AND RESULTS

Two experiments are conducted using the service robot "Kate" in a home environment (fig. 7). The first one shows excerpts of a 30 minute "robot butler" scenario where Kate can be called over to take orders from persons through speech. We discuss those parts that use active perception. In the second experiment, we discuss the recognition performance of a pick-up and drop-of task in detail. The experiments focus on recognizing few but similar objects and their properties which are representative for household goods. Videos of all experiments at [18].

A. Experimental Setup

Kate is based on a Pioneer P3DX platform and equipped with a Microsoft Kinect RGBD camera on a pan-tilt-unit. A Neuronics Katana manipulator is used for object manipulation and inspection using a small high-resolution (2560x1920 pixel) RGB iDS imaging uEye camera mounted near the tool center point (fig. 6).

B. Experiment I: The Robot Butler Scenario

In the course of the demonstration, the robot is called to the dinner table (fig. 7) where two persons order coffee with sugar and a pineapple juice. After delivering the orders, it is called to clean up the table.

All orders are different and demonstrate object recognition for mobile manipulation. In cases where a reliable object recognition is required, objects are actively inspected.

During the scenario, the robot fetches objects from different locations (fig. 7). The locations of objects are predefined in the knowledge base, so the robot knows for example that cups can be fetched from the kitchen counter.

Fig. 8. The robot getting the property (mode of coffee machine) by reading the display of the coffee machine (left). The control panel and two messages on the display telling that one coffee is being made and the machine being in standby/ready (right).

Fig. 9. The eye-in-hand camera positioned to capture the text label of a pineapple juice/"Ananas" (right object). It is of similar appearance as grapefruit juice (left object). They differ in their barcode and text labels naming their flavours. Texture is very similar. They are of the same color.

System coordination is done with SmartTCL [16], which handles all variations that come by failures during execution (see video "Robot Kate cleans up the table" at [18]).

1) Operating the Coffee Machine: In order to deliver the coffee with sugar, the robot first approaches the kitchen counter. The robot positions the scene camera to point at the kitchen counter based on the current context (location: kitchen counter). A single run of the object recognition brings enough recognition reliability to identify cups and sugar. After pouring sugar, the robot takes the cup to the coffee machine, recognizes the coffee machine, puts the cup into the machine and presses the button to make coffee.

Working in the real and complex world, mistakes and problems occur. So to be sure that the machine is making coffee, the robot reads its display to know about the current mode of the coffee machine. The eye-in-hand camera is thus positioned to point at the display for inspection of property "current mode" of the coffee machine (fig. 8). The display showing "Ready" tells that the machine is still in standby. "1 Standard Coffee" tells that coffee is being made. The robot then delivers the cup of coffee or pushes the button again.

2) Fetching Juice: When the robot has to fetch juice, it has to decide which of two similar appearing juices is the correct one (fig. 9). From the current context (location = sideboard), the robot knows that juices with different flavours are to be expected. Thus, the object recognition is configured for high probability ($P(x) > 0.65$).

The initial object recognition distinguishes the juices only very little, the overall probability is therefore not high enough. The robot then inspects the two objects. The first object is identified reliably enough by reading its text label. The robot chooses the text label instead of the barcode as the text label can be reached with lower effort. The second object is identified by reading the barcode. The robot then grasps the pineapple ("Ananas") juice and delivers it. A detailed evaluation of a similar experiment can be found in [3].

Fig. 10. Input image for initial recognition (left) of the scene and inspection images during inspection of the property "fill level" (right). Objects from left to right: onion chips, an empty and a full "hot and spicy" chips can.

3) Cleaning up the Table: At the end, the robot is ordered to clean up the table. Its task is to put reusable items into the kitchen sink and other items in the trash bin. The knowledge base contains entries that tell which objects have to be put where. For this task, the objects do not need to be recognized very reliable, that is, the flavour of the juice on the table does not matter. The required recognition probability is thus lowered ($P(x) > 0.3$). The robot does not actively inspect the objects on the table. The robot is thus not sure about the flavour of the juice but knows that it is some kind of juice which has to be thrown into the trash bin. When fetching cups, we are able to look inside if there is coffee left (property: fill level), to not to spill when stacking them.

C. Experiment II: Potato Chips

Three cans of potato chips are standing on a sideboard (fig. 10). One can is of flavour onion and two cans are of flavour "hot and spicy", but only one of them contains potato chips. The two flavours have the same texture and differ only in their color (green/black), barcode and text labels that tell the flavour and ingredients. The robot is ordered to bring "hot and spicy" potato chips and has to be sure that it is filled.

It can be observed that the robot first takes a look at the whole scene (fig. 10, left). It acquires new information by reading all their barcodes. It randomly looks into the first of the two "hot and spicy" cans (fig. 10, right) which turns out to be empty, looks into the other one which turns out to be full and finally grasps and delivers it.

When the initial object recognition is run, the recognition probability is very low for all objects (fig. 11). The hypothesis is correct for the onion flavour but the result also identifies "hot and spicy" as onion. The simple model matcher is not able to distinguish the objects due to their identical shape and therefore gives the same recognition probability for both hypotheses. MOPED is unable to identify the objects because both the bad angle to the camera and almost

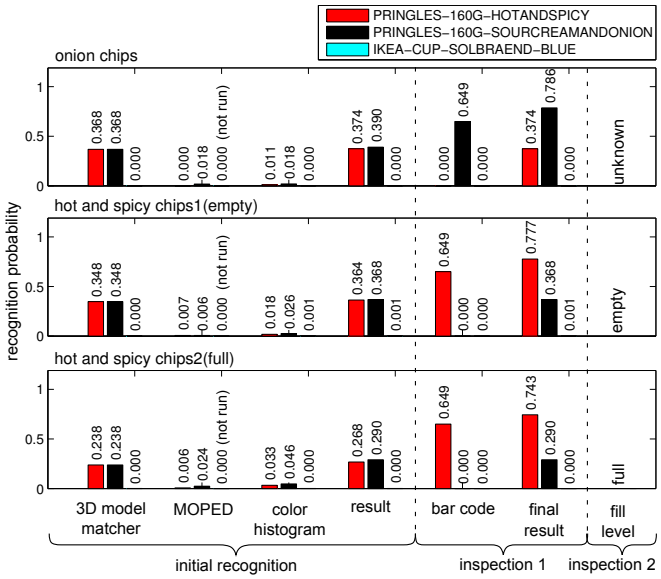

Fig. 11. Rounded recognition probabilities for each of the three candidates. The left part shows the intermediate probabilities and fused result. The right part shows inspection of barcode and property "fill level".

identical appearance (MOPED uses grayscale images for identification). The used simple color histogram algorithm has a poor recognition quality $Q(x) = 0.4$ and its result is therefore also low. The fused recognition probability is still below the threshold ($P(x) > 0.65$). Even though the flavour was not correctly recognized, the initial recognition was able to tell that the objects are potato chips and not cups (fig. 11, result of initial recognition). The robot can therefore use this knowledge to inspect the objects and acquire new information for final identification. The systematic inspection of the objects using the barcode recognition is successful. Since two objects of the same flavour were recognized, the robot looks into the cans and delivers the full one. Full and empty cans are distinguished using color histograms trained for the inside view.

D. Results

The experiments demonstrate that the proposed recognition system works both in identifying similar objects as well as in acquiring properties of objects reliably in an everyday environment. They prove that the exploitation of cues for systematic object inspection to actively acquire new information improves or even realizes object recognition. The overall performance can only be achieved using passive and active perception together. Both make use of another and cannot solve the perception task on their own in every case. Due to the careful evaluation and selection of actions to be taken, the robot was able to minimize its effort. Only thus, the robot was able to recognize objects with one manipulator movement. Without active perception, a complex scenario as in experiment I would not be possible.

VII. CONCLUSIONS AND FURTHER WORK

This paper proposed a method of active perception that combines multiple data sources, multiple views, multiple algorithms and the use of context for the purpose of object recognition. This approach closes the loop between perception, action and knowledge by using cues of an initial recognition of the scene to point an eye-in-hand camera at objects. By acquiring new information from other views on objects, they can be identified more reliable than it would be possible without the ability to change the viewpoint. Active perception is therefore a mandatory skill for service robots that can perform complex tasks in everyday environments. The proposed object recognition system is being used in several real world mobile manipulation scenarios. Videos thereof and of both experiments are available at [18].

In future, we will evaluate the selection of viewpoints and prove the approach on another platform and manipulator with more degrees of freedom.

REFERENCES

[1] "Solutions In Perception Challenge," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, Shanghai, China, 2011.

[2] M. Lutz, D. Stampfer, S. Hochdorfer, and C. Schlegel, "Probabilistic Fusion of Multiple Algorithms for Object Recognition at Information Level," in *IEEE Int. Conf. on Technologies for Practical Robot Applications*, Woburn, MA, USA, 2012.

[3] D. Stampfer, M. Lutz, and C. Schlegel, "Information Driven Sensor Placement for Robust Active Object Recognition based on Multiple Views," in *IEEE Int. Conf. on Technologies for Practical Robot Applications*, Woburn, MA, USA, 2012.

[4] D. Roberts and A. Marshall, "Viewpoint Selection for Complete Surface Coverage of Three Dimensional Objects," in *Proc. of the British Machine Vision Conference*, 1998, pp. 740–750.

[5] P.-P. Vázquez, M. Feixas, M. Sbert, and W. Heidrich, "Viewpoint Selection using Viewpoint Entropy," in *Proc. of the Vision Modeling and Visualization Conference 2001*, 2001, pp. 273–280.

[6] M. Krainin, B. Curless, and D. Fox, "Autonomous generation of complete 3D object models using next best view manipulation planning," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, May 2011, pp. 5031–5037.

[7] R. Eidenberger and J. Scharinger, "Active Perception and Scene Modeling by Planning with Probabilistic 6D Object Poses," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2010, pp. 1036–1043.

[8] K. Welke, T. Asfour, and R. Dillmann, "Active Multi-View Object Search on a Humanoid Head," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, May 2009, pp. 417–423.

[9] K. Hausman, C. Bersch, D. Pangercic, S. Osentoski, Z.-C. Marton, and M. Beetz, "Segmentation of Cluttered Scenes through Interactive Perception," in *ICRA Workshop on Semantic Perception and Mapping for Knowledge-enabled Service Robotics*, St. Paul, MN, USA, 2012.

[10] E. Klingbeil, D. Rao, B. Carpenter, V. Ganapathi, A. Y. Ng, and O. Khatib, "Grasping with Application to an Autonomous Checkout Robot," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, May 2011, pp. 2837–2844.

[11] Y. Ye and J. Tsotsos, "Where to look next in 3D object search," in *Int. Symposium on Computer Vision*, Nov. 1995, pp. 539 –544.

[12] A. Collet, M. Martinez, and S. S. Srinivasa, "The MOPED framework: Object Recognition and Pose Estimation for Manipulation," *The International Journal of Robotics Research*, 2011.

[13] Point Cloud Library, http://pointclouds.org, visited: 08/08/2012.

[14] M. Attamimi, A. Mizutani, T. Nakamura, T. Nagai, K. Funakoshi, and M. Nakano, "Real-Time 3D Visual Sensor for Robust Object Recognition," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Taipei, Taiwan, Oct. 2010, pp. 4560–4565.

[15] SmartSoft Website, http://smart-robotics.sf.net/, visited: 08/08/2012.

[16] A. Steck and C. Schlegel, "Managing execution variants in task coordination by exploiting design-time models at run-time," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2011, pp. 2064 –2069.

[17] R. Diankov and J. Kuffner, "OpenRAVE: A Planning Architecture for Autonomous Robotics," Robotics Institute, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-08-34, July 2008.

[18] YouTube: Robotics@HS-Ulm, http://www.youtube.com/roboticsathsulm.