Robust high precision 6D pose determination in complex environments for robotic manipulation

Thilo Grundmann, Robert Eidenberger, Martin Schneider, Michael Fiegert and Georg v. Wichert

Abstract— For many robotic applications including mobile manipulation robust object classification and 6D object pose determination are of substantial importance. This paper presents an object recognition methodology which is capable of complex multi-object scenes. It handles partial occlusions and deals with large sets of different and alike objects.

The object recognition process uses local interest points from the SIFT algorithm as features for object classification. From stereo images spatial information is gained and 6d poses are calculated. All reference data is extracted in an off-line model generation process from large training data sets of a total of 100 different household items. In the recognition phase these objects are robustly identified in sensor measurements.

The proposed work is integrated into an autonomous service robot. In various experiments the recognition quality is evaluated and the position accuracy is determined by comparison to ground truth data.

I. INTRODUCTION

Object recognition comprises the tasks of object class identification and pose determination. Although this plays a major role in several scientific domains, many current approaches to object recognition are limited in their applicability due to inaccuracies in the pose determination, the number of detectable objects or the complexity of scenes. Object occlusions and the appearance of similar objects in the same image often constitute problems to robust detection. In this work we present a detection methodology which allows accurate 6d object pose determination and object class detection. It uses Lowe's SIFT algorithm [1] for determining local, scale-invariant features in images. The application of the SIFT algorithm on stereo images from a camera pair enables the calculation of precise object poses. This appearance- and model-based approach consists of two separate stages, *Model generation* and *Object recognition and pose determination*. Model generation is an off-line process, where the object database is established by determining essential information from training data. We consider a set of 100 household items of different or alike appearance. The challenges lie in the reasonable acquisition and the efficient processing and storing of large data sets.

Object recognition and pose determination aims on satisfactory object classification results, low misclassification rates and fast processing. The precise detection of the object pose allows the accurate positional representation of objects in a common reference frame which is important to many tasks. It is important to notice, that for collision free

Fig. 1. A service robot commonly operates in a complex environment.

operation in realistic household scenarios, low precision of the pose of surrounding objects can jeopardize manipulation attempts due to reduced clearance. For this reason we rely on stereo vision.

The proposed method is embedded in a mobile service robotics framework. Figure 1 shows the prototype. The robust and accurate object recognition system is basis to continuing works such as object manipulation [2] [3], perception planning [4] and physical object dependency analysis [5], which point out its importance.

Section II outlines current state of the art approaches to model-based object recognition and model generation.

In Section III the data acquisition procedure and the methodology for model generation is described. Section IV depicts the principles for object class identification and 6D pose determination.

This paper closes with experiments in Section V which demonstrate the proposed theoretical concepts on real data.

II. RELATED WORK

The problem of visual object pose estimation has been studied in the context of different fields of application, robotics or augmented reality to name two.

All approaches can be characterized by a number of parameters like the number of dimensions of the measured pose (mostly one, three or six), the restrictions on the objects

to be localized (e.g. in shape, color or texture), the ability to estimate the pose of multiple instances of the same class in one image and the capacity for various different classes in the model database of the proposed system. Also the proposed sensor system (mono/stereo camera, resolution), process runtime and the achieved precision in the pose are characteristic.

One of the earlier papers on object pose estimation [6] dealt already with the high number of one hundred different classes. Nevertheless the measured pose in this case is only one-dimensional, and the scene is assumed to have a black background and no occlusion or objects of unknown classes in it.

Later on [7] and [8] extended the estimated pose to 3DOF assuming the objects to be located upright on a table, using one of five objects in their experiments.

Others [9] restricted their objects to forms that can be modeled by wire frames and achieved good results in full 6d pose estimation examining two different objects. The quality of the pose is indirectly evaluated by successful grasping.

True 6d pose measurement using local SIFT features was demonstrated by [1], describing a method that is able to localize flat objects within a range of 20 degrees, demonstrated on two scenes, consisting of three different objects each, without evaluation of the pose accuracy.

Azad et. al. presented a stereo camera based method [10] for full 6 DOF pose retrieval of textured object using classic SIFT interest points. The method requires the objects to posses flat surfaces for the stereo recognition, and the accuracy of the pose estimation is not measured directly.

Recently Collet et al. presented a mono camera object localization system [11], based on SIFT features using ransac and mean shift clustering. They also describe an almost fully automatic process for the model generation and give some experiments with four object classes where measured poses are evaluated against ground truth. The error is described by histograms over the translational and rotational error.

Another model generation system is described in [12] which is able to build a model online from a video stream. The model is also built fully automatically, additionally the user is assisted in the image collection by guiding the direction to which the object should be moved and by the visual feedback of the emerging model.

A comprehensive taxonomic and quantitative comparison of multi-view stereo model building methods for dense models can be found in [13], sparse models are out of scope for this paper though.

Our approach is comparable to the method of [11], using a similar 3d model which is not restricted in the shape based on SIFT features. Through the use of stereo vision and thus conceptual different methods for the pose determination a higher accuracy in pose measurement is achieved.

Note, that an exact direct comparison is difficult without perfect re-implementation, so we generated translational and rotational histograms similar to the ones in [11] which demonstrate a better performance of our approach. When it comes to comparison, the main drawback to the research done on six dimensional object localization is the absence of benchmark datasets.

To overcome this the KIT published[1] all image data for the 100 object classes which we used to build our models. In combination with our test scenes including full camera calibration, ground truth data and results this forms a benchmark dataset for 6d textured object localization using cameras.

III. MODEL GENERATION

Model generation aims on the acquisition of training data and its processing to generate object class models. It is essential to filter significant data and efficiently store it to enable reasonable processing times.

A stereo camera system and a laser scanner (The KIT object modeling center IOMOS [14]) are used to acquire 360 stereo images for each object and the $3d$ surface. Rotating the object on the turntable and moving the sensors results in sensor data at $360°$ latitudinal and $90°$ longitudinal angles in $10°$ steps.

The build process starts with computing the SIFT interest points (IPs) for each image and calculates 3d points by triangulation of IPs in each stereo image. Then matches over all images are determined and equivalence classes from these matching IPs found. At last each equivalence class is represented by one descriptor and one 3d location. All these equivalence class representatives together build up the model.

Now, each step of the build process is described in detail.

1) SIFT interest points calculation
   The base for the process are the SIFT interest points for each image. Each SIFT interest point $s^i = (u, v, s, o, d_l)$ consists of the 2d location (u, v) in the image, its scale s and its orientation o. d_l denotes the l-dimensional descriptor vector. For each training image the set of interest points in the left camera image S_l with $S_l = \{s^1, ... s^n\}$ is determined. The interest point set S_r for the right camera image is acquired respectively.

2) Triangulation
   3d points are computed for corresponding IP's in the left and right images. Corresponding IP's are determined by using the epipolar constraint and SIFT descriptor matching. For every IP in a left image V_l we compute the epipolar line L_e in the right image V_r and determine the subset $S_r^e \subset \{s^i \in S_r | dist(s^i, L_e) < \varepsilon_E\}$ with ε_E as the maximum epipolar distance. Then, SIFT descriptor matching is performed. It is important to do the epipolar examination before the SIFT matching step. This way the set of possible matches is constrained to a region in the image. IPs with similar descriptors from other parts are not considered and cannot distort the result. For each matched IP pair (s_l^i, s_r^j) the corresponding 3d location and orientation are computed using triangulation [15] and transformed

into the objects coordinate system to get the spatial feature representation:

$$s_\# = (x, y, z, x', y', z', s, o, d_l). \qquad (1)$$

The first three elements x,y and z denote the translational coordinates, x',y' and z' represent the direction from where the interest point is visible. Scale s, orientation o and descriptor d_l are equal to the parameters of the 2d interest point.

3) Equivalence Relation

The next step aims at partitioning this set of 3d points into subsets originating from the same physical point. This equivalence relation is seeded from IPs with fitting appearance and location and is completed by the transitive closure.

First candidates are found by appearance. The enormous amount of data with on average 850 3d locations for each training view, 280000 for each object, can be handled efficiently by means of a kd-tree. Using Euclidean distance in the SIFT descriptor space the nearest n_N- neighbors in the kd-tree are searched for each $s_\#^i$.

n_N depends on the sampling rate of the training data as SIFT descriptors are only invariant within a limited angular range and each face of the object is only seen from a certain range of camera positions. It was set to 150 in our case.

Then candidates are checked for their spacial fit by projecting them to the view of their matching partner both ways and calculating the Euclidean distance in the image. If this distance, which is the expected reprojection error, exceeds a threshold (5 pixel in our system) the candidate is rejected. In rare cases it can happen that two IP's from the same view are in one equivalence class. These IP's are removed.

A standard connected component algorithm is used to compute the transitive closure.

4) Subdivision and Representatives

We now seek a simple representation for each equivalence class above a minimal size v_{min}. Very small classes (eg. v_{min}=4) are discarded to supress IPs of low value for the recognition process and noise.

When evaluating classes one finds the locations clustering well, but descriptors to spread considerably. This is not surprising since we do see most points from a wide angle range were the SIFT descriptor cannot be assumed to be invariant. Instead of more complex density models we favor a very simple representative, which is a simple mean for location and normalized mean for descriptor. To make this simple model suitable we sacrifice the simple relation, where one class corresponds to one physical point, and split classes with k-means until they can be represented as spheres. This considerably simplifies and speeds up the recognition process.

The full model for one object needs about 5% of storage of the initial SIFT features. This enables fast recognition since
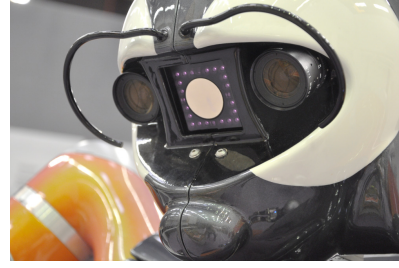

Fig. 2. Sensor head: Stereo cameras and 3d-TOF camera mounted on a pan/tilt unit

big databases can be held in RAM completely.

It had been considered to use sparse bundle adjustment for refining the model data, the camera positions and calibration. But since the provided input data had been compiled by a rather solid contraption where camera positions and calibration are accurately known this turned out not to be necessary. Also, some of the household items in the database suffer from large untextured areas that lead to views without sufficient numbers of interest points. Under such circumstances a pure sparse bundle adjustment without any initial camera pose information is expected to fail due to lack of sufficient data.

IV. OBJECT RECOGNITION AND POSE DETERMINATION

Many applications require a high pose determination accuracy. To account for this we use a stereo camera based approach, as preceding experiments using a mono camera and the posit algorithm [16] lead to unsatisfying results. Comparable results were found in [17] which showed that stereo approaches outperform mono camera approaches by the factor of ≈ 2. Our stereo setup on the robot (Fig.2)
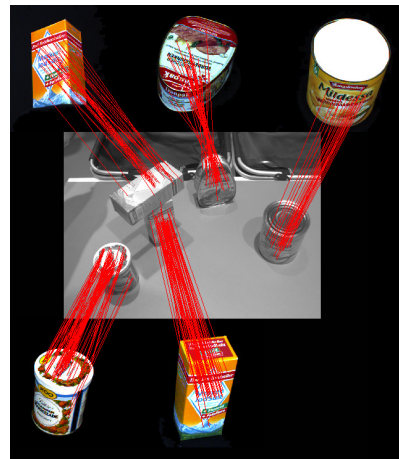

Fig. 3. Recognition principle: finding matches between interest points $s_\# \in M$ from given models and interest points detected in one image $s \in S_l$ (step 2)

consists of two Avt Pike F-145C firewire cameras with a resolution of 1388 x 1038 pixels each, equipped with 8.5mm objectives and mounted with a disparity of $0.12m$.

Precise intrinsic and stereo calibration of the cameras is essential to our algorithms so they were carried out with the Camera Calibration Toolbox for MATLAB, using about 60 stereo image pairs of a custom made highly planar checkers calibration pattern.

The recognition and localization process consists of the following steps:

1) Calculate SIFT interest points

   For each of the stereo images, V_l and V_r a corresponding set of interest points is calculated $S_l = \{s^1, ..., s^n\}$ and $S_r = \{s^1, ..., s^m\}$

2) Find correspondence to object models (Fig.3)

   For all elements $s^i \in S_l$ try to find up to p_{mm} multiple matches $c^k = \{i, j\}$ with a 3d feature from the model database $s_\#^j \in M$. The criterion for a match is that the Euclidean distance in descriptor space is below an absolute threshold $p_{tm} = 0.3$. The multiple match is needed, because with an increasing number of object models in the database, the uniqueness of a lot of features is lost. This happens in a non ignorable way for household items that often follow a corporate design and share large areas of texture.

   To speed up the search for matches, the descriptors from the database M are structured in a kd-tree using the ANN library. To increase the performance, the nearest neighbor search is approximated. The quality of the approximation can be parameterized, we used the value $p_{app} = 5$.

3) Construct stereo interest points

   For all matchings $c^k = \{i, j\}$, we try to find multi stereo matches on the right interest point set S_r. The epipolar constraint is used in the same manner as described in Section III.2, but after the epipolar spatial restriction, a relative multi match is used. This has to be done to account for the classic situation where multiple instances of the same object class are placed side by side on a board, leading to multiple similar features on the epipolar line L_e. After this procedure we obtain a set of l 3d SIFT points $S_\# = \left\{s_\#^1, ..., s_\#^l\right\}$ by triangulation. Each of these 3d SIFT points belongs to an object class, indicated by its corresponding database feature $s_\#^j$. In a system with t known classes this gives a partition of $S_\# = \left\{S_\#^1, ..., S_\#^t\right\}$, whereas some of these class depending partitions might be empty. From here on, all calculation is done separately for each object class.

4) Cluster within class t

   To account for scenes with large numbers of identical objects and to deal with the high number of erroneous 3d SIFT features, we construct initial pose estimates $P = \{p^1, ..., p^x\}$ from $S_\#^t$. This is done by choosing randomly non collinear triplets of 3d interest points from $S_\#^t$ and check whether their mutual Euclidean distances match those in the model database.

   Within the 6d space of the initial pose estimates P, qt-clustering [18] is performed to find consistent 6d pose estimates. The clusters consist of 6d poses p^x which all correspond to a triplet of 3d interest point $S_\#^t$. That way each cluster describes a set of 3d interest points $S_\#^c \subset S_\#^t$.

5) Pose determination

   All 3d interest points from each cluster $S_\#^c$ are used to determine the transformation of the database 3d IPs into the measured cloud by a least squares pose fit (Fig. 3). In the resulting list of object classes and poses a similarity search is performed to eliminate duplicate poses, which emerge from imperfect clustering.

One important advantage of this 3d SIFT stereo approach is, that in contrast to other approaches it is able to handle objects of any shape, as long as the object fulfills the requirements on its texture that are inherent to the SIFT algorithm. It is not generally bound to the usage of SIFT features, so in the future improved interest point methods could replace the SIFT interest points. Note, that the parameters in the algorithm strongly influence the performance and thus have to be optimized to fit an application scenario.

V. EXPERIMENTS

The experiments we conducted should evaluate the performance of our system in terms of detection rate and pose accuracy. Since it is very difficult to acquire pose ground truth data in complex scenarios we split the evaluation into two parts. In the first part, we set up scenes where the ground truth pose can be determined with a calibration pattern, leading to single object scenes of low complexity. The detection rate in these simple settings was 100%.

In order to evaluate our method in more realistic scenarios, we set up a second test, based on complex scenes consisting of up to 30 objects placed in arbitrary positions in a real environment supplemented with unknown objects. This way the method's robustness against occlusion, object ambiguities and environmental influences can be evaluated. In such scenarios, ground truth poses are not available, but it is possible to evaluate the correctness of the detection by visual inspection of the bounding volumes that are projected into the images.

The time required for a one shot recognition of a complete scene can be divided up into two parts: The SIFT calculation takes about 0.6 seconds per image on a 2GHz intel multicore and the processing of the poses takes 0.3 seconds without using the multi cores. Note that there are some parameters like the kd-tree approximation quality which strongly influence the runtime of the processing part.

A. Evaluation of pose accuracy

To evaluate the system's accuracy in the pose estimation, we compared the recognition results of 260 scenes with ground truth. To acquire 6d ground truth data we applied paper sheets with the projected corresponding model onto the bottom of the objects (Fig.4). Since the origin of the objects is located in the bottom plane, this method enables us to place the object onto a calibration pattern with 3DOF and with sub-millimeter accuracy.
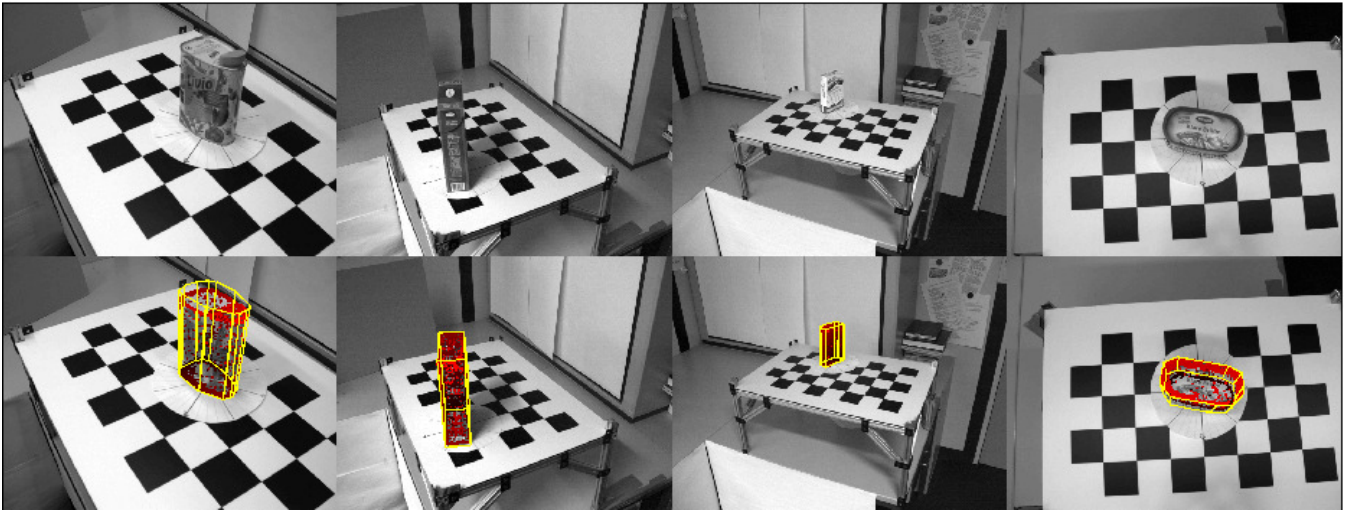
Fig. 4. 1st row: objects, placed precisely onto the calibration pattern. 2nd row: Ground truth values projected into image (dot cloud(red), bounding volume(yellow)).
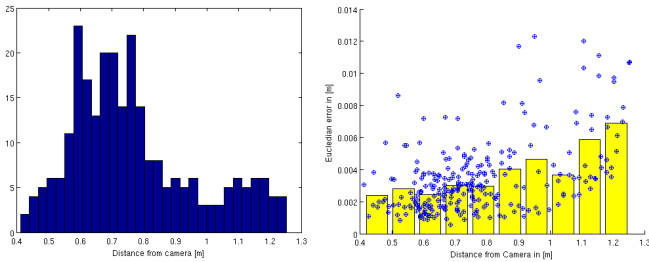

Fig. 5. 1st: Histogram of object's distance to camera over the test set. 2nd: Translational error over distance (dots) and the mean translational error over distance (bars).
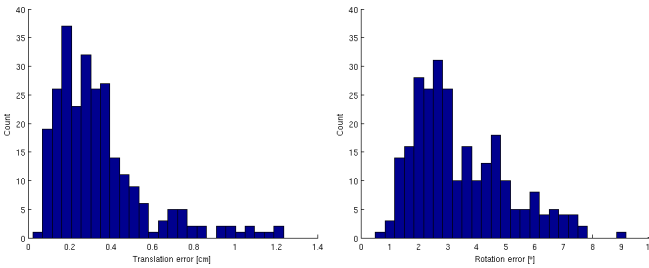

Fig. 6. Histograms of translational(in cm) and rotational error(in ∘).

Fig. 7. The xyz-deviations of all test scenes, depicted in the camera frame, with its covariance ellipsoid (95% quantile).

In the test we placed one object per scene onto a calibration pattern (Fig.4) and measured its pose. Using the camera's pose with reference to the calibration pattern which is determined using the matlab calibration toolbox, we transformed the ground truth pose into the camera coordinate frame where we compared it against the result of the recognition algorithm. We selected seven different objects with varying shapes for this experiment.

The distance of the objects origin and the camera ranged from 0.42m to 1.25m (Fig. 5/1). This area was chosen from our experiences with the working area of the mobile service robot.

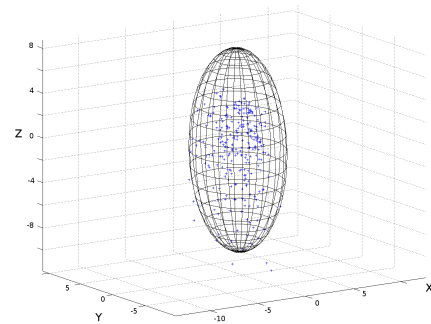Due to the relatively simple nature of the scenes all objects were detected correctly, so we collected 260 pose measurement error vectors. As expected from the measurement principle, the results show increasing translational errors with growing distance (Fig.5/2). For the evaluation of the rotational errors, we calculated the minimal rotation angle from the resulting 6d pose to the ground truth 6d pose. The overall distribution of translational and rotational errors are shown in Fig. 6, the corresponding standard deviation of the translational error is shown in Table I.

To get a more expressive model of this measurement process, we transformed the error into the camera frame, and calculated the covariance for the translational components of the measurement errors (Fig.7). As expected from the measurement principle, the standard deviation of the z_c-component in the camera frame is considerably higher as the standard deviations in the other directions x_c and y_c (Tab.I). Using this model it is possible to model detection inaccuracies fairly precisely. Note, that the inaccuracies also contain potential errors in the 3d models as well as the bias from placing the object imprecisely onto calibration pattern.

(a) Labeled office scene   (b) Detection results   (c) Legend (near objects)

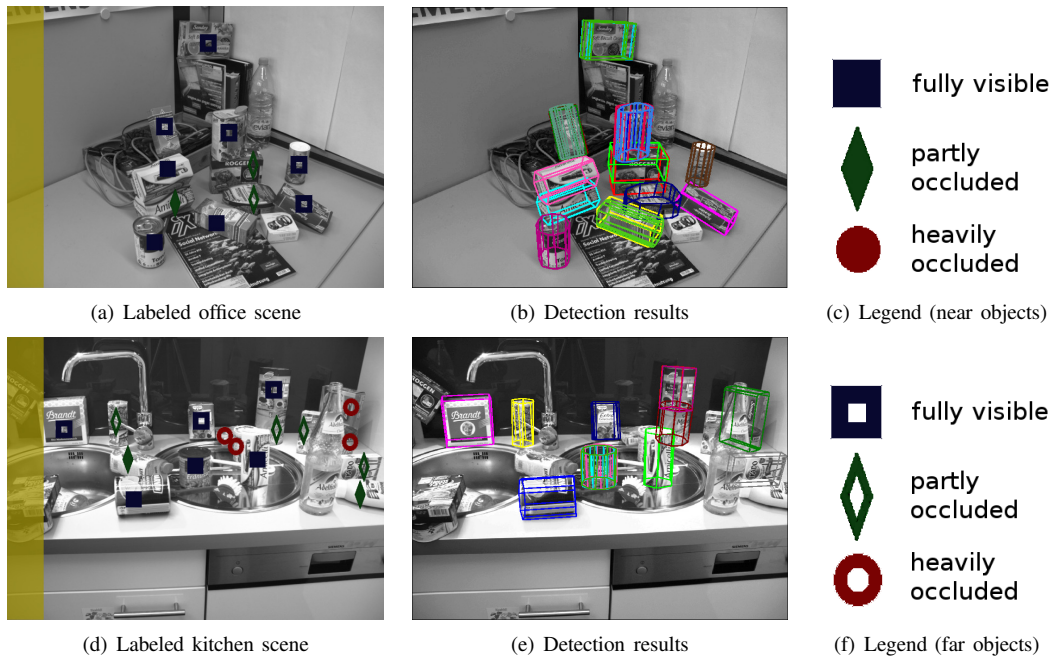(d) Labeled kitchen scene   (e) Detection results   (f) Legend (far objects)

Fig. 8. One shot detection results for two complex scenes in a kitchen and an office environment. The results are evaluated with respect to the a priori categorized objects.

TABLE I

TRANSLATIONAL ERROR IN THE CAMERA FRAME.

stddev	[mm]
$\|(x_c y_c z_c)\|_2$	2.2436
x_c	1.3763
y_c	1.3617
z_c	3.4235

B. Object recognition in complex environments

In this experiment complex and cluttered scenes are considered. A series of 60 images of different scenarios at different locations are investigated. All contain chaotic object arrangements including trained and unknown, random objects. For the evaluation, all objects are categorized according to their distance from the camera and to their occlusions. Thus, we differentiate between close and far items and cluster the objects into fully visible, partly occluded and heavily occluded ones.

Figures 8(a) and 8(d) show two scenes with the initially labeled objects. Squares mark fully visible objects, diamonds indicate partly occluded items and circles denote heavily occluded ones. Close objects are shown solid, far away objects as halves. The yellow area on the left of these images is not considered for the stereo matching, so objects in this part of the image are ignored for the recognition. The respective detection and classification results for the two scenes are illustrated in Fig.s 8(b) and 8(e). The bounding volumes with respect to the 6d poses of the recognized objects are projected into the image. In the first sample, all objects are detected. However, a small number of false positives can be recognized for the salt boxes, the rye bread and the oil can. False positives are objects that are detected although they

are not physically there. They result from ambiguous objects with similar textures on several sides or similar objects in the database, such as the three almost identical salts. In Fig. 8(e) not all objects are recognized because of heavier occlusions. Though, fully visible ones are clearly identified. The successful detection also depends on the lighting as the trained images were acquired under one specific lighting condition. Variations influence the recognition process.

By visual inspection the detection results are compared to the pre-selection. The true positive and false positive classifications rates are manually determined by checking the class and pose recognition results. Figure 9 shows the detection results of each of the 60 scenarios. The total number of objects in the scenes ranges from 7 to 30. The number of recognized objects and the false positives for each scene are also plotted. While the number of false positives is generally low, a few scenes with very dense object arrangements showed more than 10 false object hypotheses.

In Fig. 10 the detection rates are plotted. The overall rates show the number of detected objects over the total objects in the scene. The other curves depict the rate of detecting fully visible objects and partly occluded ones. The total recognition results over all scenes are listed in Table II. Rates for each category a separately and jointly determined. As expected, closer objects perform better in the recognition. In matters of occlusions, a decay can be recognized from fully visible to heavily occluded items. The overall detection rate of 72 per cent is lower than the peak detection rate of 86 per cent for close and fully visible objects. The random object alignments with unfavorable object poses, lighting influences and object occlusions are reasons for recognition failures. However, considering the large database and the complexity
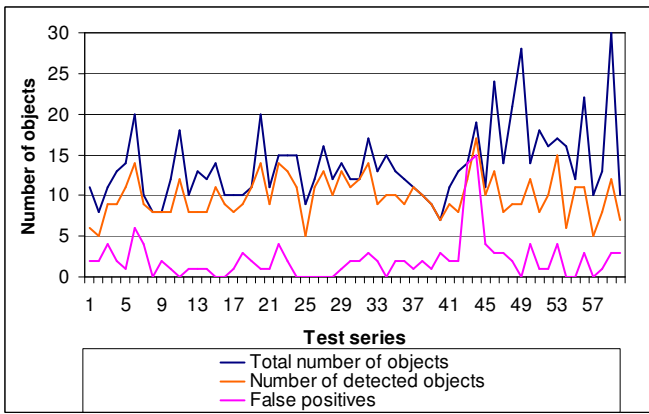
Fig. 9. Object detection results and false positives in comparison to the total number of objects in scene for 60 different scenarios with a total of 826 objects.
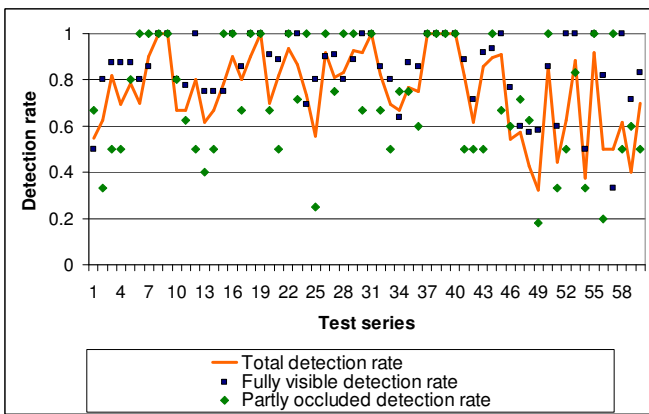

Fig. 10. Overall detection rate and detection rates of fully visible and partially occluded objects.

of the scenes the one shot recognition results are promising.

VI. CONCLUSIONS AND FUTURE WORKS

A. Conclusions

We presented a system that is able to detect and localize objects from up to 100 different classes. The $6d$ detection accuracy of the object pose and the detection rate are evaluated in extensive experiments, which demonstrated a true positive detection rate of 72% in highly complex cluttered multi object scenes with partly occlusions. The resulting pose errors had a standard deviation of $3.4mm$ in the direction of the camera (z_c) and $1.4mm$ in x_c and y_c. This result forms

TABLE II

OBJECT CLASS DETECTION RESULTS BROKEN DOWN TO THE DETECTION CATEGORIES.

| distance | detection rates | | | |
	fully visible	partly occluded	heavily occluded	total
near	0.863	0.763	0.179	0.796
far	0.813	0.593	0.167	0.617
total	0.846	0.684	0.170	0.722

a more new and finer probabilistic model for the recognition process.

A satisfactory trade-off is found between fast processing and good recognition rates and detection errors and failure recognitions. The system is suitable to applications in cluttered environments with random object alignments and unknown objects.

B. Future Works

We plan to include sparse bundle adjustment into the model generation process to increase the precision in the $3d$ models which is expected to increase the pose precision on the one hand, but also to loosen the precision requirements on the camera pose. Nevertheless we assume, that modeling approaches without any camera pose informations will fail on objects with certain texture characteristics such as highly symmetrical patterns or large untextured areas.

REFERENCES

[1] D. G. Lowe, "Object recognition from local scale-invariant features," in *International Conference on Computer Vision*, Corfu, Greece, September 1999, pp. 1150–1157.

[2] J. Kuehnle, A. Verl, Z. Xue, S. Ruehl, J. Zoellner, R. Dillmann, T. Grundmann, R. Eidenberger, and R. Zoellner, "6d object localization and obstacle detection for collision-free manipulation with a mobile service robot," in *Proc. International Conference on Advanced Robotics ICAR 2009*, 2009.

[3] Z. Xue, J. Marius Zoellner, and R. Dillmann, "Grasp planning: Find the contact points," in *IEEE Robio 2007*, 2007.

[4] R. Eidenberger, T. Grundmann, and R. Zoellner, "Probabilistic action planning for active scene modeling in continuous high-dimensional domains," *IEEE ICRA 2009*, 2009.

[5] T. Grundmann, R. Eidenberger, and R. Zoellner, "Local dependency analysis in probabilistic scene estimation," in *ISMA 2008. 5th International Symposium on Mechatronics and Its Applications*, 2008.

[6] S. Nayar, S. Nene, and H. Murase, "Real-time 100 object recognition system," in *IEEE ICRA 1996*, 1996.

[7] J. Zhang, R. Schmidt, and A. Knoll, "Appearance-based visual learning in a neuro-fuzzy model for fine-positioning of manipulators," in *IEEE ICRA 1999*, 1999.

[8] J. A. Walter and B. Arnrich, "Gabor filters for object localization and robot grasping," in *ICPR 2000*, 2000.

[9] D. Kragic, A. T. Miller, and P. K. Allen, "Real-time tracking meets online grasp planning," in *IEEE ICRA 2001*, 2001.

[10] P. Azad, T. Asfour, and R. Dillmann, "Stereo-based 6d object localization for grasping with humanoid robot systems," in *IEEE IROS 2007*, 2007.

[11] A. Collet, D. Berenson, S. Srinivasa, and D. Ferguson, "Object recognition and full pose registration from a single image for robotic manipulation," in *IEEE ICRA 09*, 2009.

[12] Q. Pan, G. Reitmayr, and T. Drummond, "ProFORMA: Probabilistic Feature-based On-line Rapid Model Acquisition," in *Proc. 20th British Machine Vision Conference (BMVC)*, London, September 2009.

[13] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski, "A comparison and evaluation of multi-view stereo reconstruction algorithms," in *IEEE CVPR 2006*, 2006.

[14] Z. Xue, A. Kasper, J. Zoellner, and R. Dillmann, "An automatic grasp planning system for service robots," in *14th International Conference on Advanced Robotics (ICAR)*, June 22nd - 26th, 2009.

[15] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004.

[16] D. F. Dementhon and L. S. Davis, "Model-based object pose in 25 lines of code," *International Journal of Computer Vision, Springer Netherlands*, vol. Volume 15, 1995.

[17] P. Azad, T. Asfour, and R. Dillmann, "Stereo-based vs. monocular 6-dof pose estimation using point features: A quantitative comparison," in *Autonome Mobile Systeme 2009*, ser. Informatik aktuell, 2009.

[18] L. J. Heyer, S. Kruglyak, and S. Yooseph, "Exploring expression data: Identification and analysis of coexpressed genes," *Genome Res.*, vol. 9, November 1999.