An Integration Concept for Vision-Based Object Handling: Shape-Capture, Detection and Tracking*

Matthias J. Schlemmer, Georg Biegelbauer, and Markus Vincze

Automation and Control Institute,
Vienna University of Technology,
1040 Vienna, Austria
{ms,gb,vm}@acin.tuwien.ac.at
http://www.acin.tuwien.ac.at

Abstract. Combining visual shape-capturing and vision-based object manipulation without intermediate manual interaction steps is important for autonomic robotic systems. In this work we introduce the concept of such a vision system closing the chain of shape-capturing, detecting and tracking. Therefore, we combine a laser range sensor for the first two steps and a monocular camera for the tracking step. Convex shaped objects in everyday cluttered and occluded scenes can automatically be re-detected and tracked, which is suitable for automated visual servoing or robotic grasping tasks. The separation of shape and appearance information allows different environmental and illumination conditions for shape-capturing and tracking. The paper describes the framework and its components of visual shape-capturing, fast 3D object detection and robust tracking. Experiments show the feasibility of the concept.

1 Introduction

A lot of detection and tracking methods have been introduced to computer vision, visual servoing gets more and more important in robotic applications and some approaches for visual learning techniques have been presented. However, these techniques are usually dissociated from each other and the connections between them are manually at best.

In this work we present a concept of a vision system that guides the manipulation of convex shaped objects. Robotic applications such as visual servoing or grasping tasks are the goal. Our main contribution is the closing of the gap between shape-capturing, detecting and tracking the object, integrating the individual vision steps in a fully automatic way. The approach is to show the object once to the robot vision system. It is scanned by a laser range sensor that derives a volumetric object description for further detection and tracking. Performing the detection in a totally different environment (e.g. in a home environment on

* This work is supported by the European project MOVEMENT (IST-2003-511670) and by the Austrian Science Foundation grants S9101-N04 and S9103-N04.

potential object places) is possible and results in the object pose, which is the starting pose for the subsequent tracker. This monocular tracker uses the 3D-pose as well as the 3D-object model delivered during the shape-capturing step for continuously updating the pose of the object. Appearance information for the tracker (cues in any form, i.e., interest points in the system proposed) is derived not until now, i.e., from the actual scene – discoupling the illumination and environmental conditions of the shape-capturing and the manipulation steps.

The paper is structured as follows: After an overview of related approaches in the next section, the main concept, the reasons for using Superquadrics and the discrete vision steps are described in detail in Section 2. First experiments are given in Section 3 and further work is outlined in Section 4.

1.1 State of the Art

Kragic and Christensen [7] clearly outline the desire for a fusion of shape and appearance information in robotic servoing and grasping. They emphasize the lack of robustness of model based techniques when trying to track line features of highly textured objects. Their solution is the usage of training images and their projection into the eigenspace. In contrast to this, we are integrating two different sensors, namely a laser scanner for providing the object model (= shape-capturing step) as well as the starting pose of the object in the scene (= detection step) and a CCD-camera (= tracking step). In contrast to the former (shape), the latter exploits appearance information. The problem of line features lies in our understanding not only in textured objects but also in situations where occlusions occur and especially when handling non-rectangular objects. We aim to solve both with our framework.

Currently information for the different tasks is often provided manually. In [5], [6], model databases are required containing local information about the model. Our contribution is a framework that allows model data, initial pose information as well as interest points for the tracking part to be automatically provided by the sensors.

Moreover our framework operates an automatic vision system including the object capturing process for size and shape parameters without any user interaction. Pioneer work in learning for 3D object recognition was done by Mukherjee et al. [14] and an approach for vision-based active learning for robot grasping tasks was introduced by Salganicoff [16]. Our learning – we call it shape-capturing – differs from the latter contributions in that way that we understand the learning process as a coded object description temporarily stored for further processing rather than classifying objects to similar groups by comparing them in a database.

A recent work by Taylor et al. [18] uses a similar full system assembly as we do. They, too, combine a laser scanner with vision but stereo instead of monocular. Their approach is finding geometrically primitive objects (bowls, cylinders, boxes) in a scene without previous learning. To achieve this, a scene segmentation is performed using surface curvature. The main difference to our work is that we divide this step into two parts: shape-capturing (see Sec. 2.1)

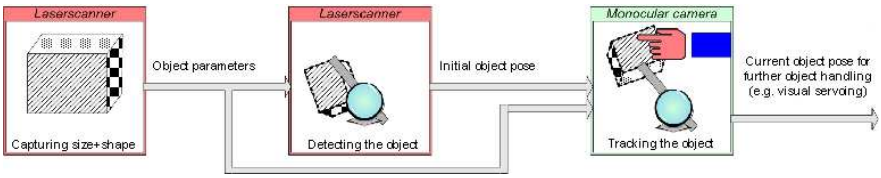


Fig. 1: Concept of our perceptual system: The fully automatic sequence starts with the object capturing where the size and shape parameters are gained that are used for subsequent object detection and tracking in an occluded and cluttered scene.

and detection of the learned object in the scene (Sec. 2.2). That way, we avoid the computational expensive segmentation and enable the handling of convex objects with less geometric constraints than [18].

Concerning the tracking part, we like to pick a recent paper by Yoon et al. [19] who presented a combination of a laser scanner and a camera for a tracking task. The selection of line features for the tracking is done manually from the range image – allowing to track complex objects (such as a toy lorry).

2 Concept

Fig. 1 shows the overall concept. First, the target object is shown unoccluded to the laser scanner which automatically derives shape and size parameters storing them in terms of a Superquadric model description (see Sec. 2.1). Detection (Sec. 2.2) is performed in the real-world scene without further user interaction as the parameters are already known from shape-capturing. The detection leads to the pose (position and orientation) of the object, starting the tracking part (Sec. 2.3) that additionally uses the object dimension acquired in the capturing step. The output, i.e., the updated pose, can be used for any further robotic task: grasping, visual servoing and so on. Fig. 2 shows the experimental assembly of the system with the vision equipment and the linear axis.

Stereo vision is usually prone to show weak accuracy and problems arise when dealing with not or weakly textured objects. Our approach aims at delivering the pose of an object with respect to the robot arm in order to be able to perform visual servoing, grasping tasks etc., which all needs high accuracy. The combination of a laser scanner and a colour camera requires a single parametric description of the object to be handled that can be passed along the different steps. Multiple parametric models have been introduced for 3D object recovery. Superquadrics are perhaps the most popular because of several reasons. The compact shape can be described with a small set of parameters ending up in a large variety of different basic shapes. The recovery of Superquadrics has been well investigated and even global deformations can be easily adopted [17]. Additionally, they can be used as volumetric part-based models desirable for robotic manipulations. These advantages cannot be found in other geometric entities, which predestine the Superquadric model for our application. For further information regarding Superquadrics, please refer to [1].

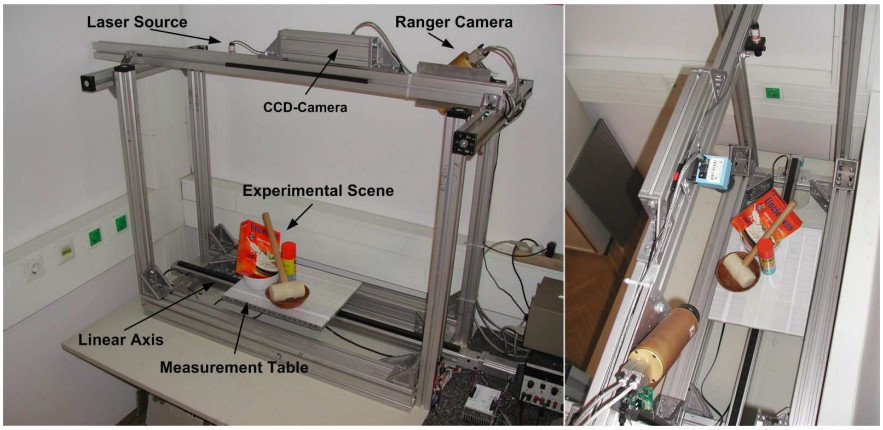


Fig. 2: Experimental sensor assembly with laser source and the ranger camera, the CCD camera for the tracking and a measurement table where the scene for the experiment in Fig. 4 is arranged.

The usage of Superquadrics for a system such as ours has several advantages. First of all, Superquadrics are purely shape-based, which frees us from using approximately the same illumination conditions when acquiring the shape, detecting and tracking the object. Second, it enables the possibility to describe a large variety of different objects especially with the extension of global deformations. Most everyday objects such as commodity boxes, cups or tin cans can be described or well approximated. Third, Superquadrics use only a small set of parameters therefore providing a very compact description of the object's surface. This implies a fourth advantage: The computation of the 3D-model coordinates that is necessary for the tracking part, can numerically be solved in a straight-forward manner.

2.1 Capturing the Shape of the Object

Before a robot can handle a convex shaped object, the vision system needs information about it. We propose a shape-capturing step by showing the object to the system and extracting its geometric properties. We use a laser range finder to acquire a range image in which the 3D shape of the object has to be directly recovered. As many sides of the object as possible (i.e., no degenerate view) and no other objects should be visible to the laser scanner. Due to the symmetry of most every-day objects one view is sufficient.

2.2 Detecting the Object

The task of this module is to scan the scene of interest to obtain a single-view range image and detect the object in process real time. The method needed for this purpose must robustly handle object occlusions in a cluttered scene. In order

to achieve fast detection results a probabilistic approach is used to verify pose hypotheses of the learned model. For keeping the computational effort low, the search process is structured in a two-level hierarchy.

First the low-level search (probabilistic pose estimation) is RANSAC-based [3] with samples on sub-scaled raw data to speed up the Superquadric recovery using the Levenberg-Marquardt [13] minimization. The best fit of the low level search is again refined finding the optimal pose which is saved.

Second the high level selection (pose verification) is necessary due to faulty detections in the low level search results. To resolve these ambiguities a ranked voting [15] of the pose hypotheses is applied considering three constraints: the quality of fit, the number of points on the Superquadric surface and the number of the Superquadric's interior points.

The hierarchical two-level search achieves a fast and robust detection result especially in cluttered scenes. Because of fitting the learned object model to local surface patches and verify them globally within the refinement step, disconnected surface patches can be associated to one entire part. This enables a robust detection of partly occluded objects. For more details on this algorithm please refer to [2]. The detected pose of the object is the initialization for the subsequent tracking with a monocular camera and the recovered shape and size parameters from the shape-capturing process provide the required model to the tracker.

2.3 Tracking the Object

Provided with starting pose information from the detection step, our tracker projects the Superquadric, acquired during the shape-capturing step, into the current camera image. The usage of Superquadrics involves the possibility of a fast computation of the convex hull which provides the boundaries of the projected object, within which interest points are now searched using any detector, e.g. hessian-laplace or harris-affine (a very good comparison can be found in [11]). For each detected point, a descriptor [12] is saved that contains its properties. Here again, any descriptor may be used, e.g. SIFT [9]. The main focus lies on good repeatability as the majority of detected points should be found again in the next frame with a very similar descriptor. However, timing behaviour is of course also a very important issue. Note that the appearance information of the object for the tracking is obtained directly from the actual scene situation, enabling the handling of different illumination and occlusion conditions of the model acquisition and the tracking step. The interest points are finally reprojected into the image for computation of the object coordinates. Here another strength of the used model stands out: A Superquadric describes the closed surface of an object, hence, the computation of the intersection point of the ray of sight through the interest point and the model immediately delivers the 3D-model coordinates of this point. This enables the association of every detected interest point in 2D with its 3D-coordinates on the object.

The tracking loop works as follows: Interest points are searched in the 2D-neighborhood of the points found in the previous image. Correspondences be-

tween the points in the two frames are established via comparing their signatures. Finally, the pose is determined using the algorithm by Lu et al. [10]. The image points from the current image are taken as observed 2D-points and the corresponding points from the previous tracking step provide the 3D object coordinate information. For handling wrong matches, the RANSAC [3] method is applied using the number of point votes and selecting the best result respectively the mean of the largest pose cluster in case of multiple equal votes. The interest points of the current frame are again projected onto the object model and the interest point positions are stored in object coordinates for the next tracking step. All interest points are used for the matching with the next frame, independently of whether they have been used when matching with the previous image or not. Thus, the overall number of points for the tracking may vary and newly appearing points are seamlessly integrated into the tracking process. This way, appearing sides of the object that were occluded before are available for supporting the tracking process. In this way problems with rotational motions are reduced.

2.4 Calibrating the system

First, the sensors have to be calibrated individually for the sake of accuracy. Second, the coordinate systems of the scanner and the camera must be registered onto each other for executing an automatic sequence of the different steps.

The calibration of the laser scanner is done using the geometrical approach. With a 3D calibration object with markers on at least two different planes, the pose of the laser plane and the extrinsic parameters of the camera can be calculated as described in [4].

The tracking camera is calibrated with the calibration tool *Camcalb*, introduced in [20]. This tool provides the intrinsic camera parameters in order to undistort the camera images for enhancement of tracking robustness and additionally gives the extrinsic parameters (position and orientation of the calibration plate) for fulfilling the last calibration step:

Laser coordinate system and camera coordinate system are finally registered via transformation between the respective world coordinate systems. This leads to the possibility of transforming the target object's position and orientation obtained by the laser scanner during the detection step into the coordinate system of the tracking camera.

3 Experimental Results

Fig. 3 shows an uncluttered scene for tracking a cylinder (one of the basic Superquadric shapes). The object (3a) is scanned (3b) and a Superquadric is fitted (3c). Scanning the scene (3d) leads to the location of the learned Superquadric (3e). This provides the starting pose (3f) for the tracker (3g–3i).

Table 1 shows the parameters of the Superquadric – both ground-truth and the captured values.

(a) Object of interest (b) Range image of object (c) Fitted Superquadric

(d) Range image of scene (e) Detection of the object (f) Tracker starting frame

(g) Tracking frame #7 (h) Tracking frame #11 (i) Tracking frame #16

Fig. 3: Experiment 1: Handling of a cylinder. Capturing model (first row), Detecting (second row) and Tracking (last row). The reprojected pose is depicted as mesh-grid.

Fig. 4 shows another whole vision sequence as presented in this paper for a more complex example. Again, we chose an every-day commodity item as object to be retrieved and tracked, this time a rectangular rice box. Table 2 sums up the parameters retrieved by the shape-capturing step. Although the accuracy of the shape-capturing is deficient on the shortest side of the object, tracking is not affected. This leads back to the derivation of the tracking cues, i.e. the interest points, from the actual scene whereas an edge-detector would be misdirected.

Note that during detection (second row of Fig. 4), the rice box now lies in an arbitrary position and is partially occluded by the white bowl, the tin can as well as the mallet shaft. The reprojected white lines in the last two rows refer to the pose of the tracker. The white points are the locations of the interest points. The matching example on the right of the third row is a zoomed clip

parameter	*size [mm]*			*shape*			
	a_1	a_2	a_3	ϵ_1	ϵ_2	k_x	k_y
model	24.5	24.5	85.2	0.1	1.0	0.0	0.0
true object	26.5	26.5	86.5	0.0	1.0	0.0	0.0

Table 1: Summarized learned Superquadric size and shape parameters of the tin can.

parameter	*size [mm]*			*shape*			
	a_1	a_2	a_3	ϵ_1	ϵ_2	k_x	k_y
model	96.8	74.9	27.1	0.2	0.1	0.0	0.0
true object	95.0	75.0	22.5	0.0	0.0	0.0	0.0

Table 2: Summarized learned Superquadric size and shape parameters of the rice box.

of frame #18. The black dots indicate the positions where interest points have been found in the previous step, the white dots the locations of the points in the current frame. Note that there are some white points that have no match with black ones (no white chain). Nevertheless, these points are stored for the next iteration as they may possibly be matched with points of frame #19.

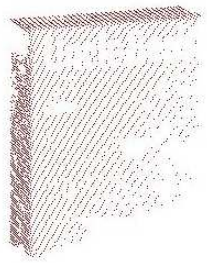
Furthermore the occlusion caused by the mallet shaft is dynamic during tracking due to the motion of the rice box. Additionally, the hand coming from the left also occludes a part of the box. Finally, even the number of visible faces of the box changes. Nevertheless, the pose is recovered with sufficient accuracy.

4 Conclusion and further work

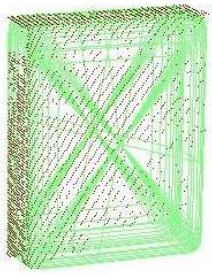
With this work we presented a vision concept that closes the gap between capturing the shape of a convex object and handling it in a cluttered and occluded scene – in an automatic way. The fusion of shape and appearance proved to be well suited for this purpose. A laser range scanner for retrieving object parameters as well as for detecting the object in the scene, is combined with a monocular CCD camera that is liable for the tracking part. This concept has been shown to provide a stable solution for shape-capturing, detecting and tracking different Superquadric shapes as cylinders and boxes.

As further work, tests – including timing analysis and quantitative evaluation – of the system will be done on our existing pan-tilt laser range sensor. Accessibility and grasping analysis will follow as soon as we mount the unit on a mobile platform. As extension for this concept, the shape-capturing and detection of more complex objects will be tackled. These objects may be expressed by a composition of several Superquadrics. This requires a learning process that parses subparts of an object automatically [8]. The current bottleneck of the tracker as far as timing is concerned is the 3D pose estimation. To achieve camera frame rate, code optimization has to be done and matching robustness must be increased in order to reduce the number of required RANSAC-iterations. Furthermore, the additional usage of cues as for example edges may also support the robustness of the monocular pose estimation.

(a) Object of interest

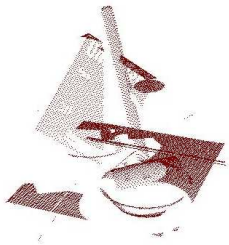


(b) Object range image

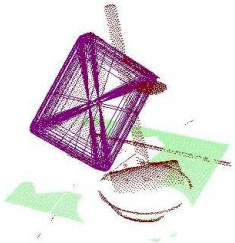


(c) Fitted Superquadric

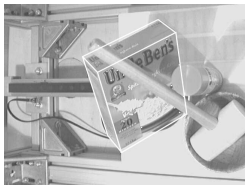
(d) Occluded object



(e) Scene range image

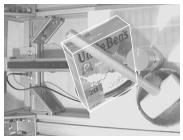


(f) Detected object



(g) Starting pose

(h) Matching example



(i) Frame #1

(j) Frame #18

(k) Frame #23

(l) Frame #28

Fig. 4: Experiment 2: Fig. (a) to (c): Capturing the object parameters; Fig. (d) to (f): Detection of the object in the scene; Fig. (g): Starting pose of the object; Here, the pose is depicted as white lines; Fig. (h): Matching example: Black dots from frame #17 are matched with white dots from frame #18; Fig. (i) to (l): Some tracking frames.

References

1. A. H. Barr, *Superquadrics and Angle Preserving Transformations*, IEEE Computer Graphics and Applications, 1981, Vol. 1(1), pp. 11-23
2. G. Biegelbauer and M. Vincze, *Fast and Robust 3D Object Detetction Using a Simplified Superquadric Model Description*, Proceedings of the 7^{th} Conference on Optical 3-D Measurement Techniques, 2005, Vol. 2, pp. 220-230

3. M.A. Fischler and R.C. Bolles, *Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography*, Communications of the ACM, 1981, Vol. 24, pp. 381-395
4. J. Haverinen and J. Röning, *A 3-D Scanner Capturing Range and Color for the Robotics Applications*, 24th Workshop of the Austrian Association of Pattern Recognition GM/AAPR, 2000, pp. 41-48
5. H.-Y. Jang et al., *A Visibility-Based Accessibility Analysis of the Grasp Points for Real-Time Manipulation*, IEEE/RSJ International Conference on Intelligent Robots and Systems, 2005, pp. 3111-3116
6. S. Kim et al., *Robust model-based 3D object recognition by combining feature matching with tracking*, Proceedings of the IEEE International Conference on Robotics and Automation, 2003, Vol.2, pp. 2123-2128
7. D. Kragic and H.I. Christensen, *Model based techniques for robotic servoing and grasping*, Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and System, 2002, Vol.1, pp. 299-304
8. A. Leonardis and A. Jaklic, *Superquadrics for segmenting and modeling range data*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997, Vol. 19(11), pp. 1289-1295
9. D. Lowe, *Distinctive image features from scale-invariant keypoints*, International Journal of Computer Vision, 2004, Vol. 60(2), pp. 91-110
10. C.P. Lu et al., *Fast and Globally Convergent Pose Estimation from Video Images*, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (June 2000), No. 6, pp. 610-622
11. K. Mikolajczyk et al., *A Comparison of Affine Region Detectors*, International Journal of Computer Vision, 2005, Vol. 65(1/2), pp. 43-72
12. K. Mikolajczyk and Cordelia Schmid, *A performance evaluation of local descriptors*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005, Vol. 27(10), pp. 1615-1630
13. J.J. Moré, *The Levenberg-Marquardt Algorithm: Implementation and Theory*, Numerical Analysis - Lecture Notes in Mathematics, Springer Verlag, 1977, Vol. 630, pp. 105-116
14. S. Mukherjee and S.K. Nayar, *Automatic generation of GRBF networks for visual learning*, Proceedings of the IEEE International Conference on Computer Vision, 1995, pp. 794-800
15. B. Parhami, *Voting Algorithms*, Machine Learning (IEEE Transactions on Reliability), 1994, Vol. 43(4)pp. 617-629
16. M. Salganicoff, *Active Learning for Vision-Based Robot Grasping*, Machine Learning (Kluwer), 1996, Vol. 23(2)pp. 251-278
17. F. Solina and R. Bajcsy, *Recovery of Parametric Models from Range Images: The Case for Superquadrics with Global Deformations*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1990, Vol. 12(2), pp. 131-147
18. G. Taylor and L. Kleeman, *Integration of robust visual perception and control for a domestic humanoid robot*, Proceedings IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2004, Vol.1, pp. 1010-1015
19. Y. Yoon et al., *A New Approach to the Use of Edge Extremities for Model-based Object Tracking*, Proceedings of the IEEE International Conference on Robotics and Automation, 2005, pp. 1883-1889
20. M. Zillich and E. Al-Ani, *Camcalb: A user friendly camera calibration software*, Workshop of the Austrian Association of Pattern Recognition GM/AAPR, 2004, pp.111-116