Autonomous In-hand Object Modeling from a Mobile Manipulator.

Philipp Feigl[1], Jean-Baptiste Weibel[1] and Markus Vincze[1]

***Abstract*— Robots require knowledge of objects to manipulate and operate them in their environment. However, such object models are not always readily available and must first be created. Service robots are well-equipped to perform this autonomously, thanks to their set of sensors and arm. Once grasped, the object of interest can be captured under many angles and separated from the background, and the relative transformation between views can be measured through proprioceptive sensors. As no object knowledge is available, the approach needs to rely on knowledge of the robot's own manipulator, and the environment stability during the manipulation.**

This work focuses on investigating different methods for segmenting objects moved by a mobile manipulator from captured RGB-D images, using knowledge of the arm and of the scene's background. These segmented views are used to reconstruct the object, based on the arm forward kinematics and Iterative Closest Point (ICP) alignment of a 3D hand model with the scene. We examine the segmentation on different objects, and demonstrate that the proposed method provides accurate results even for transparent objects.

I. Introduction

Robots are expected to perform more and more complex tasks in the coming years to be able to assist humans in dangerous, repetitive or simply boring tasks. With the population ageing in most developed countries, service robots in particular have an essential role to play in helping adults remain independent for longer. Such robots need to develop an understanding of their environment and adapt to an ever-changing set of objects to guarantee safe interaction. Manipulation of objects, for example, needs knowledge about the object shape [9] to decide how to grasp it for a specific purpose.

While humans naturally can perceive objects and estimate their sizes and shapes very quickly, robots, however, require models of objects they encounter to adapt. Manual model creation is time-consuming and generally requires expensive sensors. On the other hand, robots are equipped with suitable sensors and manipulators to create such models autonomously. Object modeling involves the collection of images with known camera poses, and an accurate segmentation of the object in view from the background and robot manipulator, without knowledge of the object in hand [14].

In this paper, we propose to take advantage of the robot depth camera and manipulator to autonomously collect views with and without the object (background view) and combine them using the arm's forward kinematics to obtain relative transformation between views and combine them into a 3D model. First, RGB and depth image differencing is applied between object views and the background view to separate the object and the arm from the background. Then, the knowledge of the arm shape and forward kinematics are used to further segment the arm from the object, and obtain the relative transformation between the end-effector of the robot and the camera. This is illustrated in the Figure 1. We show that our approach for segmentation is competitive, even for transparent objects and can reconstruct 3D models.

After introducing the relevant state-of-the-art approaches in Section II, this paper presents a pipeline to reconstruct 3D models of objects from the robot's in-hand manipulator in section III. Finally, in Section IV, we show the results of the segmentation of the robot's arm and object, as well as the accuracy of the 3D models obtained.

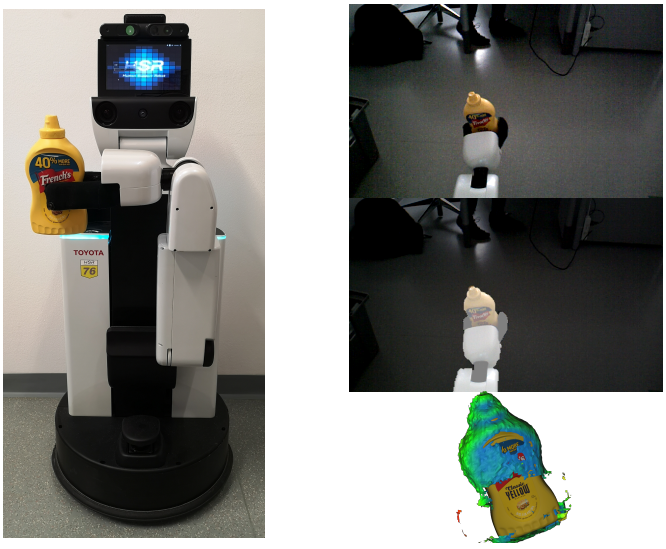

Fig. 1. The Toyota Human Support Robot is used to autonomously collect views with known camera-gripper transformation. These images can be combined, after segmentation, to create a 3D model.

II. Related Work

We present the state-of-the-art methods relevant to this work. In a first time, we focus on methods used to obtain a segmentation of the object from the background. In a second time, we present methods designed to reconstruct objects from multiple observations.

A. Segmentation

No information about the object to be modeled is available, as the purpose of the task is to model an object and therefore obtain that knowledge. Relevant object pixels must be separated from the background. Segmentation is a long-standing

[1]Automation and Control Institute, TU Wien, Vienna, Austria
philipp.feigl@tuwien.ac.at
{weibel,vincze}@acin.tuwien.ac.at

problem, when the background is static. Two categories exist, as the background is either known beforehand or is learned in the process. These classical methods are discussed in depth in [2]. More modern approaches have been developed since, with [4] training a convolutional neural network (CNN) taking as input two frames, in between which the robotic manipulator moved. The network then learns to distinguish between manipulator, background and object based on this movement. This method is powerful but its performance degrades when used with a different manipulator. It is also possible to use modern deep transformer networks and rely on more general semantic understanding of the scene to separate background and foreground as has been studied in [1].

Unknown object segmentation has also been studied based on depth information, in the years since off-the-shelf depth sensors became generally available. In [16], the authors first obtain supervoxels which are then combined based on a convexity criterion. CNN have also been applied to this task with great success [18], refining the depth-based segmentation prediction using color information.

Lastly, methods have been developed to estimate more precisely the manipulator state, enabling its segmentation from the object grasped. Good depth measurements are available for commercial manipulators as they use non-transparent non-reflective surfaces. [17] takes advantage of these depth measurements to refine the pose of every joint using iterative closest point (ICP) [3] for joints equipped with a sensor, and a particle swarm optimization (PSO) for the articulated joints without measurements (usually the fingertips of an end-effector). Depth is also used in [5], which directly classifies depth points as one of the arm joints or background.

B. Object reconstruction

3D models are obtained by combining a set of views with known camera pose. Truncated signed distance function [8] are commonly used for this purpose as they account for the noise present in depth sensors and pose estimation by smoothing out the final surface based on the amount of noise.

The reconstruction of dense surfaces in real-time by means of camera tracking is discussed in [13], using an inexpensive depth camera (KinectFusion). In [6] a signed distance function is used to minimize errors on a depth image to estimate the camera pose.

Camera information can be complemented by the manipulator forward kinematics to create object models. This is done in [12], where the arm state is tracked by a kalman filter based on the forward kinematics and the measurments obtained with an ICP-based registration while the object is manipulated by a robot hand. Because the whole surface of the object is not visible due to the fingers, multiple manipulation sequences are necessary to model the complete object. [11] learns to separate static and dynamic parts of a scene when visiting it on a regular basis to extract multiple partial views of objects and cluster them to obtain complete objects. [14] also demonstrated that it is possible to segment and reconstruct objects using a truncated signed distance function and ICP algorithm in combination with a hand tracker instead of forward kinematics, when objects are manipulated by a human hand.

III. Autonomous in-hand object modeling

The robot used for the in-hand reconstruction is the Toyota Human Support Robot (HSR) [19], [20]. The HSR is designed to support and interact with people around the house. Therefore, it is perfectly suitable for the targeted task and is equipped with a head-mounted RGB-D camera as well as an arm. The RGB-D camera is a Xtion PRO LIVE that measure depth using structured light sensing techniques. Such an approach provides accurate depth but cannot measure depth if the distance between the object and the camera is too small, or if the object is transparent.

We present a pipeline for autonomous in-hand object modeling that is suitable for this platform. Given an object is present in the robot end-effector, we record the background B (with the arm out of view) and then a sequence S of RGB-D images of the object with different arm poses P. The relative transformation between the camera and the gripper is obtained using the forward kinematics of the arm. In this section, we first detail how we segment the object and the arm from the background, then how we refine our estimate of the arm pose as well as the end-effector fingertips to better segment it out from the original recorded sequence S, and finally how we use those segmented views to model the object. An overview of the in-hand modeling pipeline is shown in Figure 2.

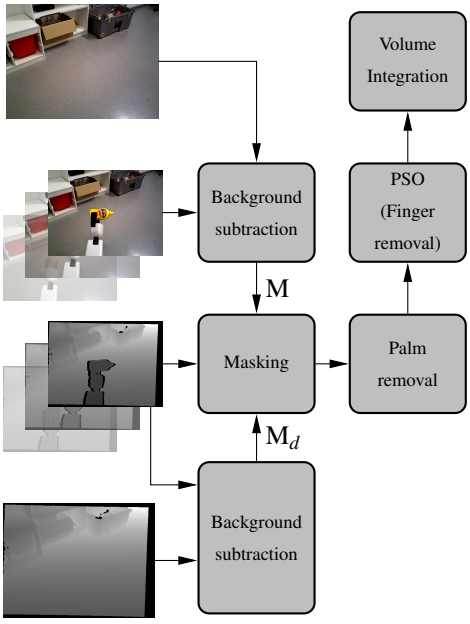

Fig. 2. Overview of the object in-hand modelling.

A. Background removal

In this part, we describe how we propose to segment the arm and object from the background in an image $I_k \in S$, a process known as Background Subtraction (BS). As we have collected a view of the scene with the arm out of view, that is a background image B, all points that are not part of the arm or the object can be expected to have a small distance between I_k and B. Any distance function $d(I_k, B)$ can be used, we apply the euclidean distance pixel-wise in our experiments:

$$d_2 = \sqrt{\sum_{i=0}^{n} \left(I_k^{c_i} - B^{c_i}\right)^2} \tag{1}$$

with n as the number of channels and c_i as the current channel. This process can be applied to the three color channels (R,G,B) or the single channel depth image (where there are valid depth measurements).

A mask can be obtained from that distance image by selecting a threshold t for that distance image, such that pixels with a distance $d_2 > t$ belong to the object or the robot's arm, and pixels with $d_2 \leq t$ belong to the background, that is:

$$M_k^m = \begin{cases} 1\,, & d_2 > t \\ 0\,, & \text{otherwise} \end{cases} \tag{2}$$

with m the modality used (depth d or color c).

To make the selection of that threshold easier, and knowing that there will always be enough difference between B and I_k, we first normalize the difference images and set a threshold to 0.1, that is 10% of the maximum distance value.

Color differencing tend to be quite noisy, and depth differencing suffers from the sensor limitations. In particular, some objects do not have any valid depth measurements, and depth values are missing around object edges due to shadowing effects between the projector and infrared camera of the depth sensor. To overcome those limitations, we also propose to average both differencing images before creating the mask.

This improves the performance but still provides poor estimates for transparent objects as they do not provide any depth measurements. However, this behavior can be taken advantage of. Indeed, if points without depth measurements are added to the depth differencing image M_k^d, we can obtain an under-segmented view of the object (with some background pixels still part of the mask). On the other hand, the mask obtained from combining the differencing images across modalities provides an undersegmentation. To accentuate this effect, we raise the threshold to 0.15 and perform a morphological erosion. With both of these images, the process of recovering the exact object boundary based on the RGB information is called alpha matting, and we propose to use the Grabcut algorithm [15] for this purpose.

For all differencing approaches, dynamic elements of the background are removed simply by picking the biggest connect component in the mask image. Example results are shown in Figure 3.

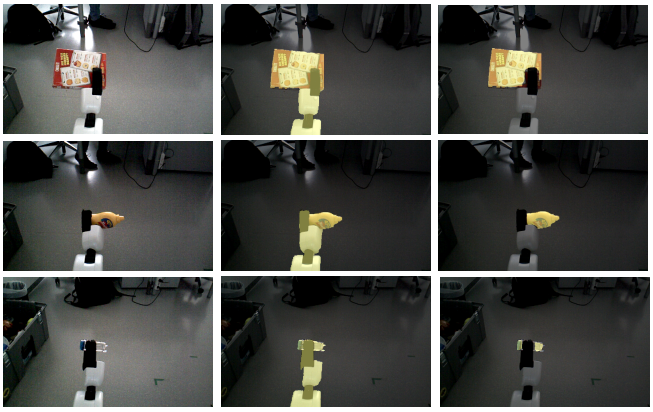

Fig. 3. Left column: RGB images of the cracker box, the mustard bottle and the transparent canister. Middle column: Highlighted are the points left after the background removal. Right column: Highlighted are the points left after the arm and fingers removal.

B. Palm removal

The approximate position of the palm coordinate system is known by the system and can be queried using the tf component of the Robot Operating System (ROS) as it is updated dynamically based on joint encoders measurements as well as arm calibration. This information can be used to initialize the pose of the 3D model of the hand in the camera frame. Noise in the sensor measuring each joint angle and arm calibration errors prevent a perfect alignment. To address this, a refinement of the palm's position is performed through an Iterative Closest Point (ICP) procedure between the hand 3D model and the scene point cloud. Points that are close to the robot's palm are considered to be part of the palm and therefore removed from the point cloud. The transform of the 3D hand model between the initial pose and the ICP-aligned one can also be used in combination with the kinematic-based one to obtain refined poses for each view.

Since the fingers of the HSR consist of two links each and the corresponding joints do not provide position information, it is not possible to adjust the fingers of the 3D model accordingly to remove the fingers with the approach described. Therefore, at this stage, the resulting point cloud still contains the forearm, the fingers and the object.

C. Finger position estimation and object segmentation

Using particle swarm optimisation (PSO), multiple finger positions of the 3D model are estimated and the overlap between the model and the scene is optimized. Points that are close to the estimated finger positions are considered as part of the fingers and removed from the point cloud, such that only the forearm and the object are left. Fingers can however be hidden in the scene or recorded from problematic perspectives. The performance here also suffers from the depth sensors limitations.

The clustering algorithm DBScan [10] further separates the remaining point cloud into several clusters, keeping only those near the fingertip position as part of the object, as can be seen in the last column of Figure 3. Using the pinhole

model and the camera intrinsics, a 2D projection is created, which is needed for the subsequent object reconstruction.

D. Object reconstruction

The above steps are performed for each image I_k of the sequence S to obtain n 2D images of the object. This provides us with mask of the object of interest in every recorded image. The reconstruction is however only possible for objects where depth information is available.

Since the information of the relative position of the hand to the camera is known according to the camera pose tracking, every view will be transformed to the initial palm pose and combined into a single model. Original poses as well as ICP-refined poses can be used for this step. The reconstruction itself is performed using TSDF volume integration.

IV. Experiments and Results

First, the experimental set-up is explained, and the chosen parameters are presented. The HSR platform described in the previous section was used to acquire data sequences for 4 different objects consisting of approximately 13 to 15 images each, under 2 different lighting conditions. The objects chosen are the crackerbox, the mustard bottle and the potted meat can from the YCB dataset [7], as high-quality models are available, as well as one transparent canister.

The initial hand pose was chosen to be the same for each object dataset for better comparability. For the TSDF Volume Integration the parameters `sdf_voxel_length` and `sdf_trunc` were set to 0.003 and 0.01, respectively. For the 3D hand model (consisting of the individual parts wrist, palm, 2 fingers, and 2 tips) 2000 points per part are used.

A. Segmentation Results

For the background segmentation different methods were tested on the captured data. The methods are presented in III and are color differencing, depth differencing, the combination of the two and the grabcut-based method. Using manually created masks of the objects and the arm, the quality of the segmentation can be determined using the precision and recall metrics, which are abbreviated here for simplicity as pr and re. They are given by

$$pr = \frac{tp}{tp+fp}, \qquad re = \frac{tp}{tp+fn} \tag{3}$$

, with tp and fp as the number of true and false positives, as well as fn as the number of false negatives.

For each image of an object sequence precision and recall were calculated to study different cases. Table I lists the precision and recall values averaged over all (arm- and object-) segmentations for each evaluated method. Grabcut, followed by color and depth differencing, are the best performing methods, while color differencing on its own has the lowest values, especially for artificial lighting, as summarized in Table IV, and noisy images.

Method	$\overline{pr}$ in %	$\overline{re}$ in %
Color differencing	81.7	66.4
Depth differencing	97.4	88.3
Color and Depth differencing	97.5	95.0
Grabcut	**98.8**	**95.3**

TABLE I

Precision and Recall of background subtraction methods averaged over all objects

The performance of the object-only segmentation (including the arm removal step) is presented in II when using the Grabcut method for background removal. Due to their simple geometry, the best results can be achieved for the cracker box and the potted meat can. The mustard bottle has a comparatively complicated geometry and reflections of the ceiling lamp reduce the precision for transparent objects.

Object	$\overline{pr}$ in %	$\overline{re}$ in %
crackerbox	92.4	90.4
mustard bottle	84.8	82.3
potted meat can	92.2	85.7
canister (transparent)	79.9	45.6

TABLE II

Precision and Recall of object separated from arm and hand for best performing background subtraction method

As can be seen in Table III, the segmentation of the arm-object combination is an easier task than that of the separation of the transparent object and arm. For color differencing it is difficult to distinguish between background and canister, just as depth differencing has problems due to insufficient depth data. Considering both modalities, on the one hand, or using grabcut, on the other hand, gives good precision. Color and depth differencing however, misses most of the transparent object points as shown by the low recall.

	$\overline{pr}$ in %		$\overline{re}$ in %	
Method	A-O	O-O	A-O	O-O
Color diff.	99.3	6.9	51.4	3.4
Depth diff.	97.1	35.3	84.6	15.2
Color and Depth diff.	97.3	62.5	93.4	23.6
Grabcut	**99.4**	**79.9**	**93.7**	**46.1**

TABLE III

Precision and Recall of background subtraction methods for transparent objects (arm-object and object-only)

In Table IV, we can see that depth-based methods are more robust to light changes. Their performance are however still degrading slightly, essentially due to reflections on shiny arm parts that prevents the depth sensor to measure anything under the light of the additional lamp.

Ceiling lamp	$\overline{pr}$ in % off	on	$\overline{re}$ in % off	on
Color diff.	99.1	64.3	66.2	66.6
Depth diff.	97.5	97.3	88.4	88.2
Color and Depth diff.	97.7	97.3	**95.9**	94.2
Grabcut	**99.5**	**98.1**	**95.9**	**94.7**

TABLE IV

PRECISION AND RECALL OF BACKGROUND SUBTRACTION METHODS DEPENDING ON THE LIGHT CONDITIONS

B. Reconstruction Results

The accuracy of the model can only be evaluated for three models (crackerbox, mustard bottle and potted meat can) as the RGB-D camera cannot measure depth for transparent objects. Distance of our models compared to high-quality models is illustrated in Figure 4.

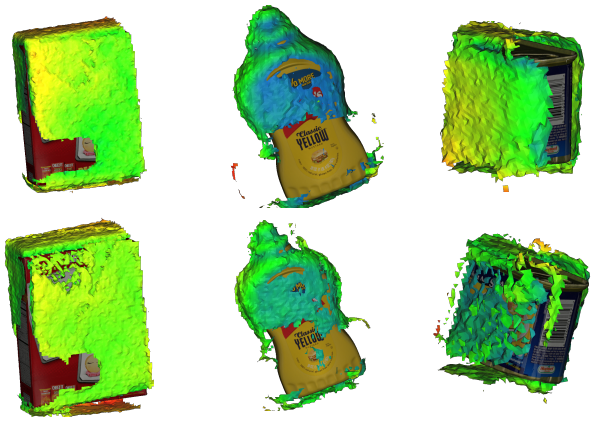

Fig. 4. Distance between our models and high-quality models. Top row: the poses are obtained with forward kinematics. Bottom row: the poses are obtained with ICP refinement

The models present errors centered close to 1 cm, creating fairly accurate 3D models that are however larger than the real objects. While the refinement step using ICP is useful for arm removal, the models obtained by adding that transformation to the forward kinematics are worse. Detailed histograms of the distance between our models and high-quality models are shown in 5. Our assumption is that the ICP refinement does provide better poses when it converges, but diverges for a few views in every object. This leads to a worse overall result, indicating that an ICP convergence criterion should be investigated to improve the results. This intuition is supported by the more accurate tip of the mustard bottle in 4, and the fact that every object presents a stronger peak around 0 in the histogram (but still higher errors overall).

V. CONCLUSION

We presented a pipeline capable of extracting and reconstructing images obtained automatically using a mobile robot manipulator. The method presented is capable of segmenting out the foreground from the background even for transparent

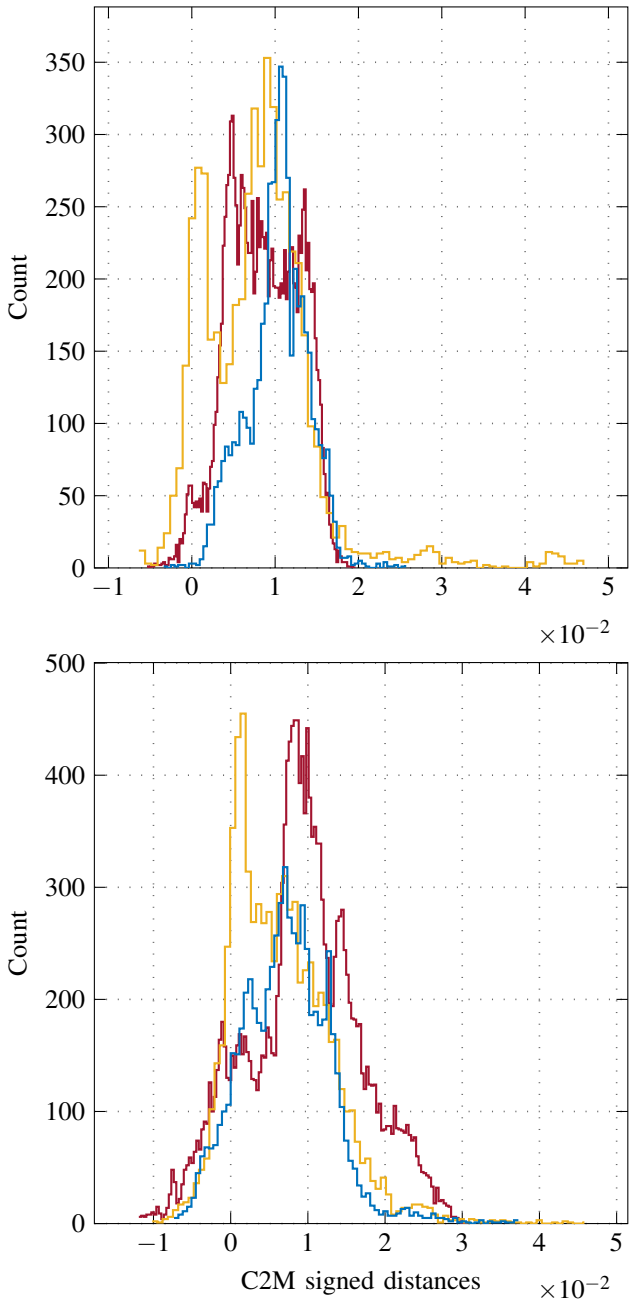

Fig. 5. Histogams of the C2M signed distances (in meters) between the modeled objects and the reference objects. The corresponding histograms, without (top) and with (bottom) refinement, are given for the cracker box in red, the mustard bottle in yellow and the potted meat can in blue.

objects with very high precision, and high recall, and further separates the object from the arm. This segmentation is shown to be applicable for 3D models reconstruction with reasonable accuracy.

We intend to further improves this work by optimizing the relative camera-hand pose estimation through the use of inverse rendering methods, enabling a purely RGB-driven estimation that can account for kinematics constraints. Furthermore, object model completeness can be improved through manipulation of the object and in particular by placing the object and grasping it from another side.

ACKNOWLEDGMENT

The research leading to these results has received funding from EC Horizon 2020 for Research and Innovation under grant agreement No. 101017089 TraceBot.

REFERENCES

[1] S. Amir, Y. Gandelsman, S. Bagon, and T. Dekel, "Deep ViT Features as Dense Visual Descriptors," *ECCVW What is Motion For?*, 2022.

[2] Y. Benezeth, P. Jodoin, B. Emile, H. Laurent, and C. Rosenberger, "Review and Evaluation of Commonly-Implemented Background Subtraction Algorithms," in *2008 19th International Conference on Pattern Recognition*, 2008, pp. 1–4.

[3] P. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, 1992.

[4] W. Boerdijk, M. Sundermeyer, M. Durner, and R. Triebel, "What's This? - Learning to Segment Unknown Objects from Manipulation Sequences," in *ICRA*, 2021.

[5] J. Bohg, J. Romero, A. Herzog, and S. Schaal, "Robot Arm Pose Estimation through Pixel-Wise Part Classification," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 3143–3150.

[6] E. Bylow, J. Sturm, C. Kerl, F. Kahl, and D. Cremers, "Real-Time Camera Tracking and 3D Reconstruction Using Signed Distance Functions," in *Proceedings of Robotics: Science and Systems*, Berlin, Germany, June 2013.

[7] B. Calli, A. Walsman, A. Singh, S. Srinivasa, P. Abbeel, and A. M. Dollar, "Benchmarking in Manipulation Research: Using the Yale-CMU-Berkeley Object and Model Set," *IEEE Robotics Automation Magazine*, vol. 22, no. 3, pp. 36–52, 2015.

[8] B. Curless and M. Levoy, "A Volumetric Method for Building Complex Models from Range Images," in *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '96. New York, USA: Association for Computing Machinery, 1996, p. 303–312. [Online]. Available: https://doi.org/10.1145/237170.237269

[9] B. Drost, M. Ulrich, N. Navab, and S. Ilic, "Model Globally, Match Locally: Efficient and Robust 3D Object Recognition," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 998–1005, 07 2010.

[10] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Knowledge Discovery and Data Mining (KDD)*, vol. 96, no. 34, 1996, pp. 226–231.

[11] T. Fäulhammer, R. Ambruş, C. Burbridge, M. Zillich, J. Folkesson, N. Hawes, P. Jensfelt, and M. Vincze, "Autonomous Learning of Object Models on a Mobile Robot," *IEEE Robotics and Automation Letters*, vol. 2, no. 1, pp. 26–33, 2017.

[12] M. Krainin, P. Henry, X. Ren, and D. Fox, "Manipulator and Object Tracking for In-Hand 3D Object Modeling," *The International Journal of Robotics Research*, vol. 30, no. 11, pp. 1311–1327, 2011. [Online]. Available: https://doi.org/10.1177/0278364911403178

[13] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "KinectFusion: Real-time dense surface mapping and tracking," in *2011 10th IEEE International Symposium on Mixed and Augmented Reality*, 2011, pp. 127–136.

[14] T. Patten, K. Park, M. Leitner, K. Wolfram, and M. Vincze, "Object Learning for 6D Pose Estimation and Grasping from RGB-D Videos of In-hand Manipulation," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 4831–4838.

[15] C. Rother, V. Kolmogorov, and A. Blake, "GrabCut - Interactive Foreground Extraction using Iterated Graph Cuts," *ACM Transactions on Graphics (TOG)*, vol. 23, no. 3, pp. 309–314, 2004.

[16] S. C. Stein, M. Schoeler, J. Papon, and F. Wörgötter, "Object Partitioning Using Local Convexity," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 304–311.

[17] B. Wen, C. Mitash, S. Soorian, A. Kimmel, A. Sintov, and K. E. Bekris, "Robust, Occlusion-aware Pose Estimation for Objects Grasped by Adaptive Hands," *International Conference on Robotics and Automation (ICRA) 2020*, 2020.

[18] C. Xie, Y. Xiang, A. Mousavian, and D. Fox, "Unseen Object Instance Segmentation for Robotic Environments," *IEEE Transactions on Robotics*, vol. 37, no. 5, pp. 1343–1359, 2021.

[19] T. Yamamoto, K.Terada, A. Ochiai, F. Saito, Y. Asahara, and K. Murase, "Development of Human Support Robot as the research platform of a domestic mobile manipulator," *ROBOMECH Journal*, vol. 6, no. 4, pp. 1–15, 2019.

[20] T. Yamamoto, K. Terada, A. Ochiai, F. Saito, Y. Asahara, and K. Murase, "Development of the Research Platform of a Domestic Mobile Manipulator Utilized for International Competition and Field Test," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2018, pp. 7675–7682.