Learning to Segment Object Affordances on Synthetic Data for Task-oriented Robotic Handovers

Albert Daugbjerg Christensen[1]
albertdaugbjerg@gmail.com
Daniel Lehotsky[1]
lehotsky97@gmail.com
Marius Willemoes Jørgensen[1]
marius139@gmail.com
Dimitrios Chrysostomou[2]
dimi@mp.aau.dk

[1] Department of Electronic Systems
Aalborg University
Fredrik Bajers Vej 7, DK-9220, Aalborg East, Denmark

[2] Robotics & Automation Group,
Department of Materials and Production
Aalborg University
Fibigerstraede 16, DK-9220, Aalborg East, Denmark

Abstract

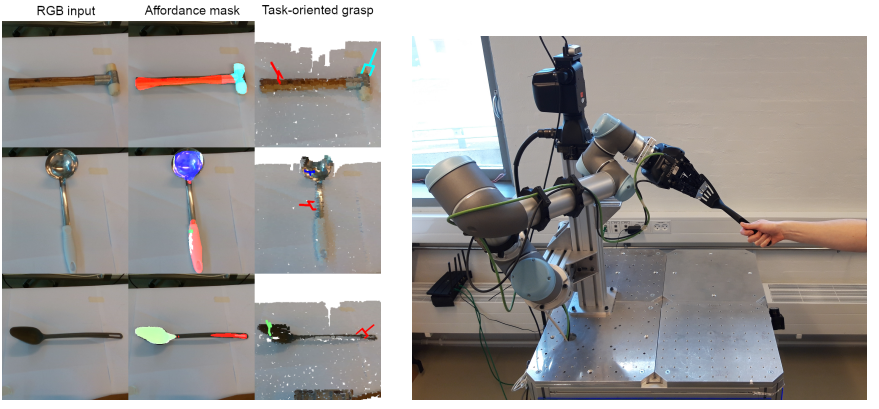
The ability to perform successful robot-to-human handovers has the potential to improve robot capabilities in the circumstances involving symbiotic human-robot collaboration. Recent computer vision research has shown that object affordance segmentation can be trained on large hand-labeled datasets and perform well in task-oriented grasping pipelines. However, producing and training in such datasets can be time-consuming and resource-intensive. In this paper, we eliminate the necessity for training in these datasets by proposing a novel approach in which training occurs on a synthetic dataset that accurately translates to real-world robotic manipulation scenarios. The synthetic training dataset contains 30245 RGB images with ground truth affordance masks and bounding boxes with class labels for each rendered object. The object set used for rendering consists of 19 object classes capturing 7 affordance classes.

We propose a variant of AffordanceNet enhanced with domain randomization on the generated dataset to perform affordance segmentation without the need of fine-tuning on real-world data. Our approach outperforms the state-of-the-art method on synthetic data, by 23%, and achieves performance levels similar to other methods trained on massive, hand-labeled RGB datasets and fine-tuned on real images from the experimental setup. We demonstrate the effectiveness of our approach on a collaborative robot setup with an end-to-end robotic handover pipeline using various objects in real-world scenarios. Code, the synthetic training dataset, and supplementary material are publicly available at: https://bit.ly/AffNet-DR.

1 Introduction

Successful robotic handovers are critical for seamless operation in collaborative tasks where both humans and robots have to handle diverse objects and tools, from disassembly scenarios

© 2022. The copyright of this document resides with its authors.
It may be distributed unchanged freely in print or electronic forms.



(a) (b)

Figure 1: **(a)** Object affordances generated by our proposed framework. **From left to right:** RGB input images, the segmented object affordance masks and the task-oriented grasps generated with our approach. Colors indicate the affordances as red: Grasp, teal: Pound, blue: Contain, lime green: Scoop (best viewed in color). **(b)** Our collaborative robotic setup performs a successful robot-to-human handover based on the proposed work. A full video demonstration is available in our code repository at: `https://bit.ly/AffNet-DR`.

in industrial shopfloors [13] to robotic-assisted surgery [31]. From the human perspective, a handover is a natural, collaborative action between two people where a giver attempts to deliver an object to the receiver for a specific task. Nonetheless, in the robotic world, a handover is an intricate procedure that requires accurate object detection, precise grasping, adequate trajectory planning, and proper anticipation of the human's position.

A handover can be either task-agnostic or task-oriented. In the case of task-agnostic handover, we are primarily interested in the success rate of the action itself. However, in the context of symbiotic human-robot collaboration (HRC), we also need to consider the subsequent task of the user after the handover action completes successfully. Therefore, the tool must be grasped using task-oriented grasping methods and result in a task-oriented handover for an uninterrupted workflow.

Using the affordance theory as first introduced by Gibson [10], several methods for task-oriented grasping have been proposed [1, 17, 37]. Object affordances refer to the functionalities that an object facilitates irrespective of the current state of the object [14]. Naturally, the constituting parts of an object have different affordances, which allow object affordance detection to be treated as a pixel-wise segmentation problem [35]. Therefore, task-oriented grasps can be generated by first segmenting the object affordances from the visual input e.g., 3D CAD models and RGB images, and then computing a grasp associated with the proper affordance [15]. Fig. 1 demonstrates some examples of object affordance segmentation and task-oriented grasps generated from our proposed method.

Recent affordance segmentation methods are based on deep neural networks which are known to require vast amounts of data in order to learn and generalize [15, 22, 23]. On the one hand, annotating datasets with labels for each available pixel of the respective objects is a resource-demanding task that does not scale well for large datasets. On the other hand, datasets containing synthetic data are easier to generate and annotate while they require sig-

nificantly less resources [4]. However, such datasets suffer from the sim2real gap, meaning that frameworks trained on synthetic data might perform poorly on real-world situations [30].

In order to alleviate this problem, domain randomization can be used [28]. By randomizing parameters such as scene lightning, object poses and textures in the simulation environment, the real world could appear as another variation to the model. Work such as [29] indicates that methods using domain randomization can achieve comparable or better results than utilizing real-world data only.

In this paper, we propose a novel framework for robotic handovers where we use synthetic data generated with domain randomization for the training of an AffordanceNet deep neural network variant.We validate the outcome of the training with a real-world dataset and thereafter we prove that it generalizes well by performing successful robotic handovers with our collaborative robot setup. The contributions of our work are:

- A novel method which improves the segmentation of object affordances using a deep neural network, even when it is trained solely with synthetic data.
- A new approach for generating synthetic data for object affordance detection using domain randomization. We show that domain randomization is sufficient for reducing the sim2real gap significantly and enabling real-world robotic handovers.
- An open dataset consisting of synthetic data, all implementation details and ROS packages required for reproducing the robotic handovers shown in this work.

The code used in this paper is made publicly available via our github repository where the synthetic dataset, the deep neural network, and the ROS packages for robot control can be found.

2 Related Work

Detecting affordances based on visual data has been studied and used in many robotic applications [19, 20] with great focus on robotic manipulation [33]. Earlier research proved that it was possible to detect graspable object parts by extracting geometric properties from point clouds [24], geometric features from RGB-D images [21] and semantic scene information [5]. However, such approaches suffered a significant drop in performance when generalization was needed.

Deep learning methods have been proven to outperform traditional affordance segmentation approaches [22] and object-based approaches predicting the position, class, and object affordances simultaneously have become popular in task-specific grasping [15]. Nguyen et al. [23] proposed applying a modified Faster-RCNN network to detect dense feature maps on depth data and post-processed with dense CRFs to improve the performance further.

Recently, a method called AffordanceNet [6] outperformed other methods by proposing a modified Mask-RCNN network to detect object affordances. AffordanceNet performs well in real-world scenarios and often serves as the baseline in robotic manipulation research. The approaches of [7] and [25] experimented with changing the quality of the features extracted by the backbone of AffordanceNet scoring slightly higher when replaced VGG16 with ResNet-based networks.

Attention modules have recently been applied in object affordance detection methods [11, 32, 34, 36]. The methods are not object-based but instead approach the problem as a segmentation problem. Zhao et al. [36], draw inspiration from the potential symbiotic relationship of

affordances and objects and propose an attention and relationship-aware module to improve the segmentation of affordances. Yin et al. [34] applied recent advances in image segmentation methods in their SEANet network and modified it to incorporate a spatial gradient fusion module and a shared gradient attention module. Recently, Gu et al. [11] utilized an encoder-decoder architecture with a DRN network to extract features with attention modules to improve upon the salient details of the affordance map.

Most of the related work is focused on improving the architectures for object affordance detection based on tailored real-world data with limited variation. The UMD dataset [21] contains hand-annotated, pixel-wise affordance labels from RGB-D input of 105 tools captured in calibrated conditions. At the same time, the IIT-AFF [23] dataset was collected to address the lack of generalization and included a subset of images from ImageNet in order to introduce variation and diverse contexts. The necessity for considerable resources to facilitate pixel-wise affordance annotation is a significant barrier in developing such datasets. Weakly supervised approaches have been proposed to overcome this issue, but the challenge persists when the datasets contain a large quantity of data [9, 26, 27].

Therefore, Chu et al. [2] proposed to learn affordances by training on synthetic images with their method, AffNet-DA. They mainly collected RGB-D images captured in a Gazebo simulation and domain adaptation was used to bridge the sim2real gap by training unsupervised on the UMD dataset after training on the synthetic dataset. However, a performance drop of 30% compared to techniques trained only on real-world data was observed.

Inspired from the work in [2], we propose an improvement in the context of affordance segmentation using synthetic data by applying domain randomization. The proposed method is trained using RGB images, and it can localize affordance candidates in multiple objects in the UMD and our dataset. We achieve a rise of 23% in segmentation performance compared to AffNet-DA, while the real-world handover experiments with our collaborative robot setup validate that we bridge the sim2real gap successfully.

3 Methodology

3.1 Generation of the synthetic dataset

A synthetic dataset generator was implemented based on the Unity game engine, which produces synthetic images with corresponding ground truth in the form of pixel-wise affordance masks and bounding boxes with a class label, as illustrated in Fig. 2. The dataset is generated using domain randomization principles, in order to overcome the sim2real gap. The premise of domain randomization is to train the network on a dataset with a great variance, such that the corresponding objects in the real world is seen as just another variation to the network. Similar to the work of [28] and [29], we vary a wide list of parameters, including object textures, object poses and illumination conditions, during the dataset generation. A full list of the parameters can be found in Table 1 and sampled them based on a uniform distribution.

A synthetic version of the UMD dataset was created by annotating 84 different objects covering 19 object classes (Knife, saw, scissors, shears, scoop, spoon, trowel, bowl, cup, ladle, mug, pot, shovel, turner, hammer, mallet, tenderizer, bottle and drill) with affordance labels covering 7 affordance classes (grasp, cut, scoop, contain, pound, support and wrap-grasp) present in the UMD dataset. The objects are imported into the Unity game engine as a set of a mesh and a texture. Each object is annotated with their associated affordances by editing the texture. Geometric primitives with randomized dimensions and textures were

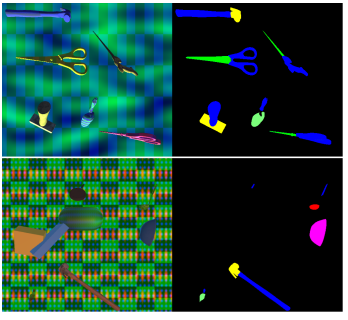


Figure 2: Two scenes from the dataset. **Top row:** A scene without distractors. **Bottom row:** A scene with distractors. Distractors are annotated as background.

Table 1: Parameter values of our synthetic dataset

Randomized paremeter	**Values**
No. objects in scene	3 to 25
No. distractors in scene	5 to 10
Pose of objects and distractors	Random position and random orientation
Scale of objects and distractors	1 to 5 times
Object distortion	Each axis is scaled 0.75 to 1.25 times.
Textures	Sinusoid-, perlin-noise, checkerboard and photo-realistic.
Light sources	1 to 3 with random poses
Light color	0 to 255 in all rgb channels
Light intensity	1 to 4 Unity game engine units
Screen res. width	400 to 600 pixels
Screen res. height	400 to 600 pixels

used as distractors. More complex shapes were occasionally achieved by distractors occluding each other. The purpose of including distractors is to train the network for situations when foreground objects are not all classifiable [28]. Furthermore, occluded or partially visible objects were removed from the dataset. Typically, objects were always spawned within the camera's field of view. However, after the objects' sizes were randomized, the bounding boxes often partially exceeded the camera's field of view. Consequently, such objects were despawned and removed. Similarly, when the initially spawned object was occluded by either a distractor or a newly spawned object, the new object occluding the initial object was removed. Therefore, the generated dataset only contains fully visible items inside the camera's field of view.

3.2 AffordanceNet implementation

Our approach is inspired by AffordanceNet [6] and it can simultaneously predict the position, class and affordances of objects in RGB images. Unlike [7], we do not make use of depth images as simulated depth images vary substantially from real-world depth images. Instead, we rely only on RGB visual input due to the high performance and resolution of available cameras providing us RGB images.

The network is akin to Mask-RCNN. It consists of a CNN backbone in combination with a feature pyramid network for feature extraction. Region proposals are fed into the three task branches. The classification branch outputs C object categories. The classification branch

loss L_{cls} is cross entropy loss calculated on the softmax normalised output as in (1).

$$L_{cls} = -\sum_{c=1}^{C} log \frac{exp(x_{n,c})}{exp(\sum_{i=1}^{C} x_{n,i})} y_{n,c} \tag{1}$$

Where x is the prediction and y is the binary target. Likewise, the affordance mask branch loss L_{aff} also uses cross-entropy loss but in a pixel-wise manner.

The regression layer predicts four bounding box coordinates for each object. The output is $N \times 4$, where N is the amount of predicted objects. The regression loss L_{loc} is smooth L1 loss, and computed as in (2):

$$L_{loc} = \sum_{i \in x,y,w,h} Smooth_{L1}(t_i^u - v_i) \tag{2}$$

Where

$$Smooth_{L1}(x) = \begin{cases} 0.5x^2, & \text{if } |x| < 1. \\ |x - 0,5|, & \text{otherwise.} \end{cases} \tag{3}$$

We train the whole network end-to-end with a multi loss function, with loss functions specific to the task of each branch as in (4).

$$L = L_{loc} + L_{cls} + L_{aff} \tag{4}$$

3.2.1 Key changes from AffordanceNet

There are a couple of differences between our implementation and the original AffordanceNet. To begin with, AffordanceNet's VGG16 backbone has been replaced with a ResNet backbone, since ResNet-based backbones improve the extracted features and in turn provide better affordances [7, 25]. Due to computational resource limits, the mask branch has been reduced compared to AffordanceNet as well. Each upsampling layer contains 128 channels as opposed to 512 in AffordanceNet. The kernel size of the first upsampling layers has also been reduced to 4 from 8 and the stride from 4 to 2.

3.3 Task-oriented grasping pipeline for robotic handover

We utilize a two-stage method for task-oriented grasping by combining the segmented affordance masks and task-agnostic grasps as shown in the pipeline depicted in Fig. 3. Affordance

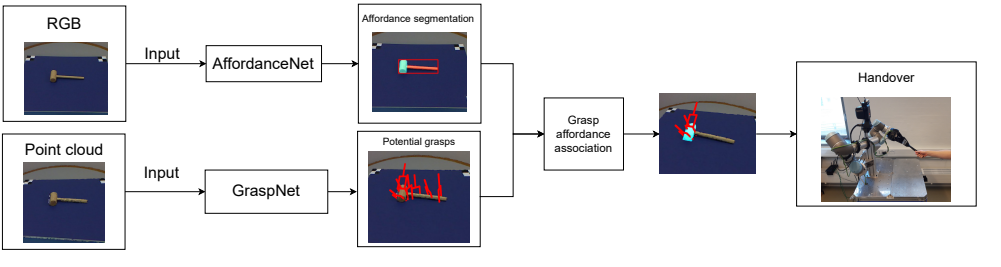


Figure 3: Proposed two-stage method for task-oriented grasping by combining affordance segmentation and task-agnostic grasp sampling.

masks are segmented in an RGB image with our implementation of AffordanceNet while we sample 6-DOF task-agnostic grasps using GraspNet [8]. Afterward, we compute non-axis aligned bounding boxes for the segmented affordances of each object in the point cloud used from GraspNet. Then, potential grasps are associated with an affordance based on the direction vector of the 6-DOF grasp. We extrapolate the direction vector in 3D space to check whether it intersects with any affordance bounding boxes or not. If any of the extrapolated points violate the boundaries of one of the axis-aligned bounding boxes, we associate the grasp with the affordance of the bounding box. Finally, after the grasps are associated with the detected affordances, we use MoveIt [3] to calculate and execute the trajectory so the robot can reach its grasping pose and subsequent handover.

4 Experiments & Results

The proposed system is evaluated with two different tests. The first test measures the performance of the proposed affordance segmentation method on the UMD dataset. The second test measures the handover success rate, when the affordance segmentation method is integrated into a system capable of performing real-world robot-to-human handover experiment.

4.1 Performance on object affordance segmentation

4.1.1 UMD dataset

A baseline network was trained on real-world data to compare our solution against. The UMD dataset [21] consists of 105 object categories captured as 28844 RGB images covering 17 object classes. Each image in the dataset depicts a single item captured on a turn-table. Object affordance ground truth is provided for each for each of the 17 objects classes in the RGB images. Rectangular bounding boxes for training our network were computed as the smallest bounding box fitting all present pixel affordances in an image.

4.1.2 Synthetic dataset

A synthetic dataset was generated to train the implemented AffordanceNet following the outlined methodology. In total, 30245 synthetic images were generated in the Unity game engine with associated ground truth.

4.1.3 Training details

Two versions of our AffordanceNet were trained, a baseline trained on the UMD dataset and one trained on the synthetic dataset. Both networks were trained in the same manner. A ResNet-50 [12] pre-trained on the COCO dataset [16] serves as the network's backbone. All weights of the backbone were frozen during training and were, therefore, not altered to take advantage of the backbone's generalized weights. We trained the UMD baseline for 15 epochs with the category-split dataset. The synthetic variant trained on the synthetic dataset for 6 epochs. We used a learning rate of 0.05 with a stochastic gradient descent optimizer. The learning rate was decreased every third epoch by a factor of 10.

Table 2: Performance on the UMD dataset in terms of average F_β^ω scores

	Real-world data		Synthetic data	
	AffordanceNet [8]	**Baseline**	**AffNet-DA [2]**	**Our method**
Grasp	0.731	0.482	0.473	0.611
Cut	0.762	0.575	0.599	0.604
Scoop	0.793	0.647	0.332	0.639
Contain	0.833	0.859	0.83	0.710
Pound	0.836	0.655	0.224	0.804
Support	0.821	0.519	0.541	0.578
W-grasp	0.814	0.848	0.821	0.785
Average	**0.799**	**0.655**	**0.546**	**0.676**

4.1.4 Evaluation metric and results

For evaluating the affordance maps, we make use of the commonly used F_β^ω score as in (5).

$$F_\beta^\omega = (1+\beta^2)\frac{Precision^\omega \cdot Recall^\omega}{\beta^2 \cdot Precision^\omega + Recall^\omega} \tag{5}$$

Where $Precision^\omega$ and $Recall^\omega$ are weighted as specified in [18]. We set the object threshold at 0.9, and the affordance masks are treated with argmax. The F_β^ω are computed on the UMD category-split evaluation dataset. The average F_β^ω scores are reported in Table 2 while qualitative results obtained on the UMD dataset are depicted in Fig. 4.

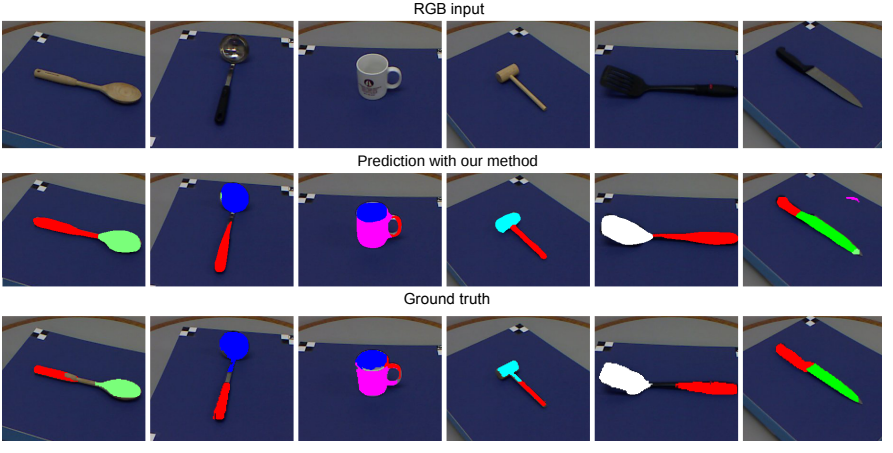


Figure 4: Predictions made on the UMD dataset. The proposed method directly generalizes to real-world data while only training on synthetic data. The colors of the pixels indicate the segmented object affordances. Red: Grasp. Lime-green: Scoop. Blue: Contain. Pink: Wrap-grasp. Teal: Pound. White: Support. Green: Cut (best viewed in color).

4.2 Performance on real-world robotic handovers

Real-world robotic handover experiments were conducted to confirm that the proposed method generalizes successfully to real-world conditions. We used a set of four objects, a hammer,

Table 3: Success rates of real-world robotic handovers

		Handover success rates	
Object	**Affordance**	**Baseline**	**Our method**
Hammer	Grasp	40 %	90 %
	Pound	0 %	90 %
Spoon	Grasp	50 %	90 %
	Scoop	20 %	70 %
Ladle	Grasp	80 %	80 %
	Contain	90 %	80 %
Turner	Grasp	90 %	70 %
	Contain	70 %	90 %
Average success rate		**55** %	**82.5** %

a spoon, a ladle and a turner with the main goal to grasp them based on the segmented affordances. Tests were conducted with our method trained on synthetic data and the baseline trained on the UMD dataset. The experiments were carried out with a collaborative robot platform consisting of a UR5 robot, a Robotiq 3-finger gripper as the end-effector, and a Pan-Tilt unit supporting an Intel RealSense D435 camera, as Fig. 1.b shows. All the trajectories were generated with MoveIt on a workstation running Ubuntu 18.04 and ROS Melodic.

4.2.1 Testing procedure

During the manipulation experiments, the following test procedure was followed for both the synthetic data and real-world data baseline networks:

(1) The test is carried out with one item at a time. The item is placed fully visible in the camera view. (2) The network performs object detection with a confidence threshold of 0.5. If the the item is misclassified the test run is considered failed. If no detection is found, the item is re-positioned up to three times before the test run is considered failed. (3) A grasp targeting a specific affordance is attempted. The binary success metric reports if the correct affordance was used for successful grasping operation.

We performed robotic handovers with each item for twenty times, ten per grasp affordance and ten per functional affordance e.g., support affordance for the spatula. The success rates of handovers achieved with the baseline and our method are reported in Table 3.

5 Discussion

The F_β^ω scores presented in Table 2 show that our domain randomization approach outperforms the current state-of-the-art method AffNet-DA by a significant margin. Unlike the previous method, our method does not need to fine tune on real-world data but directly generalizes to the UMD dataset from training on synthetic data only.

The results also show that we outperform our baseline trained on the UMD dataset and real-world data, by a slight margin. However, neither our method trained solely on a synthetic data, or the baseline performs as well as AffordanceNet [6]. This indicates that the architecture of our network can be significantly improved to achieve higher F_β^ω scores and higher success rates in robotic handover experiments.

Nevertheless, the results presented in Table 3 show that our method trained only on synthetic data, translates well into the real world. Moreover, because our method achieves a 27.5% higher real-world manipulation success rate compared to the baseline, we can there-

fore assume that our method, trained on synthetic data, generalizes better to the real-world data despite the comparable F_β^ω score with the baseline.

6 Conclusion & Future work

We presented a novel approach for real-world robotic handover based on the prediction of object affordances using domain randomization on synthetic data. Qualitative and quantitative experiments showed that our method, does not suffer from sim2real gap while it outperforms the current state-of-the-art synthetic method AffNet-DA by 23% in terms of F_ω^β score on the UMD dataset. High success rate on real-world robotic handover experiments proved that even though the F_β^ω scores of our method trained on synthetic data are comparable to a baseline, trained on the real-world UMD dataset, our method generalizes better to real-world visual input.

Interesting future directions of this research could include experimentation with novel and more efficient neural network architectures in order to achieve better segmentation masks and higher success rate on the robotic handover experiments. Additionally, our synthetic dataset generator could be enhanced with structured domain randomization.

Finally, as the UMD dataset is rather simplistic in terms of complexity. It would be interesting to investigate, if synthetic data can improve object affordance detection in more complex scenes, such as the ones found in the IIT-AFF dataset [23], which has both object occlusions and clutter and more varied environments and viewpoints.

7 Acknowledgements

This research was partly supported by EU's SMART EUREKA programme S0218-chARmER, Innovation Fund Denmark (Grant no. 9118-00001B) and the H2020-WIDESPREAD project no. 857061 "Networking for Research and Development of Human Interactive and Sensitive Robotics Taking Advantage of Additive Manufacturing – R2P2".

References

[1] Fu-Jen Chu, Ruinian Xu, Landan Seguin, and Patricio A. Vela. Toward affordance detection and ranking on novel objects for real-world robotic manipulation. *IEEE Robotics and Automation Letters*, 4(4):4070–4077, 2019. doi: 10.1109/LRA.2019.2930364.

[2] Fu-Jen Chu, Ruinian Xu, and Patricio A. Vela. Learning affordance segmentation for real-world robotic manipulation via synthetic images. *IEEE Robotics and Automation Letters*, 4(2):1140–1147, 2019. doi: 10.1109/LRA.2019.2894439.

[3] David Coleman, Ioan Sucan, Sachin Chitta, and Nikolaus Correll. Reducing the barrier to entry of complex robotic software: a moveit! case study. *Journal of Software Engineering for Robotics*, 5(1):3–16, 2014. doi: 10.6092/JOSER_2014_05_01_p3.

[4] Celso M de Melo, Antonio Torralba, Leonidas Guibas, James DiCarlo, Rama Chellappa, and Jessica Hodgins. Next-generation deep learning based on simulators and synthetic data. *Trends in cognitive sciences*, 2021. doi: 10.1016/j.tics.2021.11.008.

[5] Renaud Detry, Jeremie Papon, and Larry Matthies. Task-oriented grasping with semantic and geometric scene understanding. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3266–3273. IEEE, 2017.

[6] Thanh-Toan Do, Anh Nguyen, and Ian Reid. Affordancenet: An end-to-end deep learning approach for object affordance detection. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5882–5889, 2018. doi: 10.1109/ICRA.2018.8460902.

[7] Chau Nguyen Duc Minh, Syed Zulqarnain Gilani, Syed Mohammed Shamsul Islam, and David Suter. Learning affordance segmentation: An investigative study. In *2020 Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–8, 2020. doi: 10.1109/DICTA51227.2020.9363390.

[8] Hao-Shu Fang, Chenxi Wang, Minghao Gou, and Cewu Lu. Graspnet-1billion: A large-scale benchmark for general object grasping. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11441–11450, 2020. doi: 10.1109/CVPR42600.2020.01146.

[9] Juergen Gall and Johann Sawatzky. Adaptive binarization for weakly supervised affordance segmentation. In *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 1383–1391, 2017. doi: 10.1109/ICCVW.2017.164.

[10] James J Gibson. The theory of affordances. *Hilldale, USA*, 1(2):67–82, 1977.

[11] Qipeng Gu, Jianhua Su, and Lei Yuan. Visual affordance detection using an efficient attention convolutional neural network. *Neurocomputing*, 440:36–44, 2021. doi: 10.1016/j.neucom.2021.01.018.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. doi: 10.1109/CVPR.2016.90.

[13] Sebastian Hjorth and Dimitrios Chrysostomou. Human–robot collaboration in industrial environments: A literature review on non-destructive disassembly. *Robotics and Computer-Integrated Manufacturing*, 73:102208, 2022. doi: 10.1016/j.rcim.2021.102208.

[14] Lorenzo Jamone, Emre Ugur, Angelo Cangelosi, Luciano Fadiga, Alexandre Bernardino, Justus Piater, and José Santos-Victor. Affordances in psychology, neuroscience, and robotics: A survey. *IEEE Transactions on Cognitive and Developmental Systems*, 10(1):4–25, 2018. doi: 10.1109/TCDS.2016.2594134.

[15] Mia Kokic, Johannes A Stork, Joshua A Haustein, and Danica Kragic. Affordance detection for task-specific grasping using deep learning. In *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, pages 91–98. IEEE, 2017. doi: 10.1109/HUMANOIDS.2017.8239542.

[16] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. doi: 10.1007/978-3-319-10602-1_48.

[17] Weiyu Liu, Angel Andres Daruna, and S. Chernova. Cage: Context-aware grasping engine. *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2550–2556, 2020. doi: 10.1109/ICRA40945.2020.9197289.

[18] Ran Margolin, Lihi Zelnik-Manor, and Ayellet Tal. How to evaluate foreground maps. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2014. doi: 10.1109/CVPR.2014.39.

[19] Huaqing Min, Chang'an Yi, Ronghua Luo, Jinhui Zhu, and Sheng Bi. Affordance research in developmental robotics: A survey. *IEEE Transactions on Cognitive and Developmental Systems*, 8(4):237–255, 2016. doi: 10.1109/TCDS.2016.2614992.

[20] Luis Montesano, Manuel Lopes, Alexandre Bernardino, and JosÉ Santos-Victor. Learning object affordances: From sensory–motor coordination to imitation. *IEEE Transactions on Robotics*, 24(1):15–26, 2008. doi: 10.1109/TRO.2007.914848.

[21] Austin Myers, Ching L. Teo, Cornelia Fermüller, and Yiannis Aloimonos. Affordance detection of tool parts from geometric features. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1374–1381, 2015. doi: 10.1109/ICRA.2015.7139369.

[22] Anh Nguyen, Dimitrios Kanoulas, Darwin G. Caldwell, and Nikos G. Tsagarakis. Detecting object affordances with convolutional neural networks. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2765–2770, 2016. doi: 10.1109/IROS.2016.7759429.

[23] Anh Nguyen, Dimitrios Kanoulas, Darwin G. Caldwell, and Nikos G. Tsagarakis. Object-based affordances detection with convolutional neural networks and dense conditional random fields. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5908–5915, 2017. doi: 10.1109/IROS.2017.8206484.

[24] Andreas ten Pas and Robert Platt. Localizing handle-like grasp affordances in 3d point clouds. In *Experimental Robotics*, pages 623–638. Springer, 2016. doi: 10.1007/978-3-319-23778-7_41.

[25] Kun Qian, Xingshuo Jing, Yanhui Duan, Bo Zhou, Fang Fang, Jing Xia, and Xudong Ma. Grasp pose detection with affordance-based task constraint learning in single-view point clouds. *Journal of Intelligent & Robotic Systems*, 100(1):145–163, 2020. doi: 10.1007/s10846-020-01202-3.

[26] Johann Sawatzky, Abhilash Srikantha, and Juergen Gall. Weakly supervised affordance detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5197–5206, 2017. doi: 10.1109/CVPR.2017.552.

[27] Johann Sawatzky, Martin Garbade, and Juergen Gall. Ex paucis plura: learning affordance segmentation from very few examples. In *German Conference on Pattern Recognition*, pages 169–184. Springer, 2018. doi: 10.1007/978-3-030-12939-2_13.

[28] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 23–30, 2017. doi: 10.1109/IROS.2017.8202133.

[29] Jonathan Tremblay, Aayush Prakash, David Acuna, Mark Brophy, Varun Jampani, Cem Anil, Thang To, Eric Cameracci, Shaad Boochoon, and Stan Birchfield. Training deep networks with synthetic data: Bridging the reality gap by domain randomization. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1082–10828, 2018. doi: 10.1109/CVPRW.2018.00143.

[30] Jean-Baptiste Weibel, Timothy Patten, and Markus Vincze. Addressing the sim2real gap in robotic 3-d object classification. *IEEE Robotics and Automation Letters*, 5(2): 407–413, 2020. doi: 10.1109/LRA.2019.2959497.

[31] Albert Wilcox, Justin Kerr, Brijen Thananjeyan, Jeffrey Ichnowski, Minho Hwang, Samuel Paradis, Danyal Fer, and Ken Goldberg. Learning to localize, grasp, and hand over unmodified surgical needles. *arXiv preprint arXiv:2112.04071*, 2021.

[32] Ruinian Xu, Fu-Jen Chu, Chao Tang, Weiyu Liu, and Patricio A. Vela. An affordance keypoint detection network for robot manipulation. *IEEE Robotics and Automation Letters*, 6(2):2870–2877, 2021. doi: 10.1109/LRA.2021.3062560.

[33] Natsuki Yamanobe, Weiwei Wan, Ixchel G Ramirez-Alpizar, Damien Petit, Tokuo Tsuji, Shuichi Akizuki, Manabu Hashimoto, Kazuyuki Nagata, and Kensuke Harada. A brief review of affordance in robotic manipulation research. *Advanced Robotics*, 31 (19-20):1086–1101, 2017. doi: 10.1080/01691864.2017.1394912.

[34] Congcong Yin, Qiuju Zhang, and Wenqiang Ren. A new semantic edge aware network for object affordance detection. *Journal of Intelligent & Robotic Systems*, 104(1):1–16, 2022. doi: 10.1007/s10846-021-01525-9.

[35] Philipp Zech, Simon Haller, Safoura Rezapour Lakani, Barry Ridge, Emre Ugur, and Justus Piater. Computational models of affordance in robotics: a taxonomy and systematic classification. *Adaptive Behavior*, 25(5):235–271, 2017. doi: 10.1177/1059712317726357.

[36] Xue Zhao, Yang Cao, and Yu Kang. Object affordance detection with relationship-aware network. *Neural Computing and Applications*, 32(18):14321–14333, 2020. doi: 10.1007/s00521-019-04336-0.

[37] Yixin Zhu, Yibiao Zhao, and Song-Chun Zhu. Understanding tools: Task-oriented object modeling, learning and recognition. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2855–2864, 2015. doi: 10.1109/CVPR.2015.7298903.