Learning Grasp Evaluation Models Using Synthetic 3D Object-Grasp Representations

Minh Nguyen[1], Paul G. Plöger, Alex Mitrevski[1], and Maximilian Schöbel[1]

Hochschule Bonn-Rhein-Sieg, Grantham-Allee 20, 53757 Sank Augustin, Germany
minh.nguyen@smail.h-brs.de,
{paul.ploeger,aleksandar.mitrevski,maximilian.schoebel}@h-brs.de

Abstract. This project considers the problem of generating data for training grasp evaluation models. Recent advances are reviewed for four main aspects most relevant to labeled grasp data synthesis, namely feature extraction from perceptual data, object-grasp representation, grasp evaluation techniques, and data generation techniques. From this review, one may conclude that while data synthesis for learning a grasp evaluation model is promising, recent approaches are either limited by difficulties in collecting large-scale human grasp experience, or by the shortcomings of using analytical metrics to label generated data. Additionally, a completed object grasping pipeline is integrated, from object detection to grasp pose detection and grasp execution. Two set of experiments are performed on the Toyota Human Support Robot for two pose estimation methods using this grasping pipeline. The pipeline proves reliable and fast enough for performing the experiments, being able to execute 20 grasps per object without interruption. While further extension and optimization are needed, the pipeline enables directly examining and comparing more advanced grasp planning methods in the future.

Keywords: Grasp learning · Data synthesis.

1 Introduction

Robot grasping with multi-fingered robotic hands is a challenging problem, and finding a grasp planning solution which resembles humans' grasps in dexterity and robustness is still an area of active research. This project focuses on grasping tasks relevant to the Robocup@Home competition [1]. These tasks involve typical objects in a domestic environment, some of which can be seen in figure 1. The grasp experiments described in section 4 will be conducted on the Human Support Robot (HSR) from Toyota [2].

Figure 2 outlines the aspects relevant to generating a grasp strategy for an arbitrary object. Grasp synthesis approaches are generally classified as either analytical or empirical [23]. Analytical methods typically consider the following mechanical properties of the contact points between the gripper's fingers and

[1] http://www.robocupathome.org
[2] https://www.toyota-global.com/innovation/partner_robot/robot/

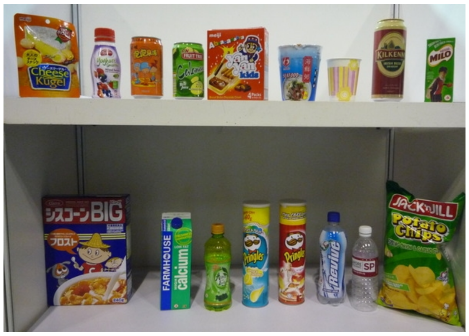


Fig. 1: Typical objects in the Robocup@Home competition [17].

an object [21,23,25]: disturbance resistance, dexterity, equilibrium and stability. Empirical approaches, on another hand, typically rely on some form of grasp experience for learning to evaluate grasp candidates. Bohg et al. [3] further categorize these methods based on how much information is assumed about the object: whether they are known, familiar or completely unknown to the robot.

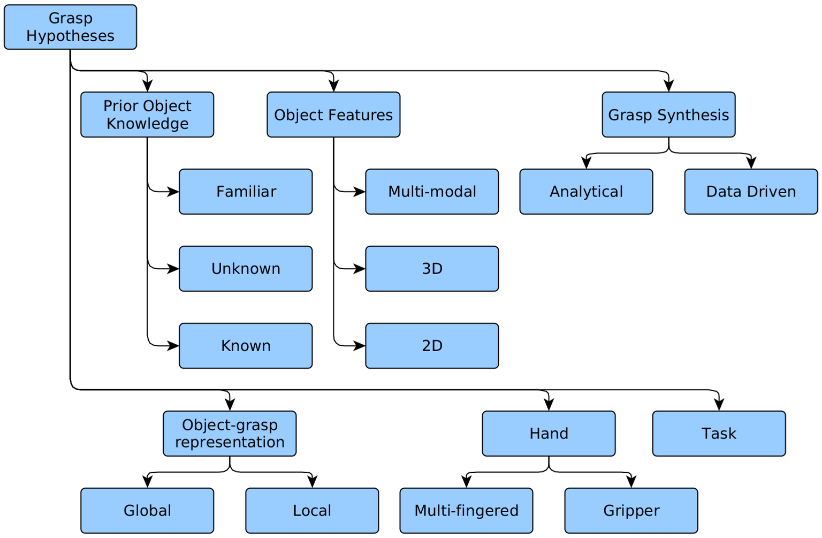


Fig. 2: Aspects which may influence generation of grasp hypotheses [3].

Because of their dependence on knowledge about the object, the end-effector and the environment, which maybe missing or incomplete in real world applications, analytical methods have been shown to be unreliable in synthesizing stable grasps when applied on real robots [13,22,28]. Grasping data, however, are time-consuming and costly to collect. In this context, the next section will discuss recent advances in fields relevant to synthesizing data for training grasp evaluation models, specifically: how to extract features from perceptual data, how to represent object-grasp relations from these features, how to evaluate these representations, and how to synthesize and augment grasp datasets.

2 Advances in aspects relevant to empirical grasp synthesis

2.1 Feature extraction from perceptual data for grasping

RGB-D cameras are becoming the perceptual sensor of choice for robotic systems because of their richer features compared to pure RGB images [14,7,11,12], especially in the context of grasping where 3D information of an object's surface heavily influences how it can be grasped. Extracting features from RGB-D data often requires consideration of their inherent multi-modality. Bo et al. [1] build sparse coding dictionaries for RGB-D data using the K-SVD algorithm and propose to use a hierarchical matching pursuit (HMP) algorithm to compute a feature hierarchy for new RGB-D images. Lenz et al. [14] use deep auto-encoders to build a representation for each feature channel, then introduce "structured regularization" to combine the depth and color representations.

More recent approaches leverage the success of Convolutional Neural Networks (CNN) in image processing for extracting features from RGB-D data. Gupta et al. [11] propose the HHA representation, which encodes the depth each pixel of the depth image with horizontal disparity, height above ground, and the angle between the local surface normal and direction of gravity. Eitel et al. [7] propose to use convert the depth value directly into RGB values using a jet color mapping. Porzi et al. [19] introduce a convolutional block called *DaConv* to learn scale awareness using depth information.

Instead of projecting RGB-D data onto different feature channels, several methods are developed to deal with 3D information directly, via training either a volumetric CNN architecture or multiple CNN models from different perspectives on data generated from object meshes in simulation. Notably, Qi et al. [20] develop two novel volumetric CNN architectures and extend the multi-view CNN technique proposed by Su et al. [26] for object classification.

Techniques are also developed to deal with occluded regions in point clouds. Bohg et al. [2] search for a symmetric plane perpendicular to the workspace surface (i.e. table) and mirror the incomplete point cloud across this plane to fill in the missing information. Varley et al. [27] train a CNN to predict whether the obscured cells in an occupancy grid are occupied.

2.2 Object-grasp representation

Bohg et al. categorizes object-grasp representations into ones using local (i.e. curvature, contact area with the hand) or global features (center of mass, bounding box).

Techniques based on global features. Ciocarlie and Allen [4] introduce Eigengrasps and defined them as principal components of the dataset of human hand configurations. The Eigengrasp planner is proposed where a quality metric are optimized for the Eigengrasp representations of grasp candidates. The planner is also used by Goldfeder and Allen [8] to synthesize a grasp dataset for object models. Mahler et al. [16] represent grasps in 2D images by aligning the image center to the gripper central point and the image's middle row to the grasp axis. The grasp is assumed to have an approach vector perpendicular to the table and hence can be characterized by the gripper center and the angle of the gripper axis with respect to the table.

Techniques based on local features. Several methods represent grasp candidates for bipedal gripper as rectangles positioned in RGB and/or depth images at contact point with the object [12,5,14]. Features extracted from the RGB-D image regions within these rectangles are then used to train a model (e.g. Multilayer Perceptron (MLP) [14] or CNN [12]) to evaluate the respective grasp candidates.

Kappler et al. [13] extract local shape representations (or templates) as projection of the objects' point clouds onto grids with a predefined resolution. These grids are aligned with the plane tangent to the object surface at the intersection point between the grasp approach vector and the object.

Gualtieri et al [10] propose to represent a grasp candidate by the cuboid swept out by a two-fingered gripper as it closes on the object, where the occluded points are sampled. The regions are voxelized into $60 \times 60 \times 60$ grids and projected onto three orthogonal planes to create 15 channels before being used as input for a CNN grasp evaluation model.

2.3 Grasp evaluation

Analytical grasp metrics. Roa and Suárez [21] group analytical grasp quality measures into approaches focusing on contact point position, hand configuration, and ones which combine both metric types. Contact point grasp quality measures focus on the object's properties, friction constraints, and form/force closure conditions, most often via analyzing properties of the grasp matrix G or the grasp wrench space (GWS) $\mathcal{P}$. Methods focusing on hand configuration often extends metrics calculated from G to the hand-object Jacobian H. Analytical metrics can be extended to consider task affordance via limiting the analysis to only movements relevant to performing a specific task.

Most popular among analytical metrics is the ϵ-metric (also known as the ball metric). The metric can be geometrically characterized by the radius of the largest sphere that can be contained in $\mathcal{P}$ and centered at its origin [21]. Weisz and Allen [28] extend the ϵ-metric by introducing noise to object pose in simulations and rating grasp candidates by the probability that they obtain a certain ϵ-metric score for all pose perturbations. Evaluation using this extension shows that grasps with the best ϵ score is much more likely to have low ϵ score when pose variations are introduced. While the proposed approach is more robust to pose uncertainty, calculation of the metric is intractable in real time.

Learning to predict grasp quality. Table 1 summarize the most recent and prominent approaches to training a grasp evaluation model. Jiang et al.'s [12] trains an SVM model to learn the weight matrices which linearly combine pixel-wise values of numerous filters of the original RGB-D data to produce a final grasp quality score. Lenz et al [14] use auto-encoders as feature extractors from RGB-D point clouds and use the encoder weights to initialize the MLP model before training it to predict the probability of success for grasp candidates. More recent approaches [13,10,16] train CNN models directly on 2D projections or filters of their object-grasp representations to produce a grasp quality value.

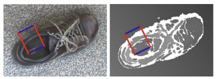
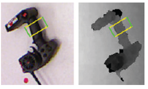
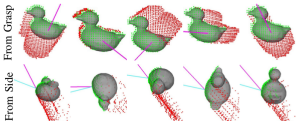
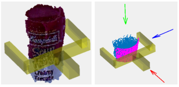
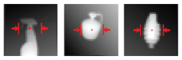
Method	Object-grasp representation	Feature extraction & learning model	Data generation
[12]		Histogram of hand-crafted filters; Model: SVM.	Rectangles manually annotated.
[14]		Auto-encoders to initialize weights, structured regularization to combine depth and RGB data; Model: MLP.	Extension of the dataset from [12] (above).
[13]		RGB rendering of "template grids"; Model: LeNet CNN	Quality of grasps are calculated in simulation for object meshes, verified via crow-sourcing.
[10]		Filters of cuboid regions projected onto 3 orthogonal planes, creating 15 channels; Model: LeNet CNN.	Quality of grasps are calculated for object meshes using force-closure
[16]		Depth images cropped and aligned to gripper; Model: CNN combined with single-layer NN.	Quality of grasps are calculated for object meshes using a variant of ϵ-metric from [28]

Table 1: Five recent empirical approaches to grasp quality prediction

2.4 Generating data for grasp success prediction

Training a grasp evaluation model is heavily dependent on the quality of the grasp dataset. Humans are the most versatile and dexterous manipulators known, but human grasp experience data are often costly and time-consuming to collect. In table 1, the first two approaches use human-labeled data. However, Jiang et al.'s [12] dataset contains only 300 data points for both training and testing, and the extended version used by Lenz et al. [14] also has only 1000 data points.

This motivates developing data synthesis techniques for creating large amount of data automatically and efficiently.

The latter three approaches in table 1 are different attempts at this task. They rely on analytical grasp metrics instead of human grasp experience to label grasp data, however, either calculated directly [13] or in simulation [10,16] for 3D object models. Evaluation models trained on these datasets, therefore, suffer from the same weaknesses of analytical metrics in real environments as discussed in 2.3.

In order to increase robustness against noises in real environments, augmentation techniques are also applied to the original dataset before training. Eitel et al. [7] sample noise patches of fixed size from real RGB-D data, divide them into five groups and randomly combine instances from these groups to create noise patterns for augmenting original training samples. Kappler at al [13] introduces noise to the object poses while sampling for reference grasps before generating each data point. Gupta et al. [11] simply add a low-frequency white noise to the disparity images to augment the depth information. RGB data augmentation techniques include geometric transformations such as mirroring, rotating, shifting [9], or photometric ones such as color scaling, contrasting [6].

3 Grasp planning

In order to perform grasp experiments, a simple pose estimation is implemented, and the Grasp Quality CNN (GQCNN) approach proposed by Mahler et al. [16] is integrated. The Single-Shot Multibox Detector (SSD) [15] architecture is integrated to detect objects in RGB images as rectangle regions, which are then used to crop point clouds and depth images for the pose estimation and GQCNN algorithms.

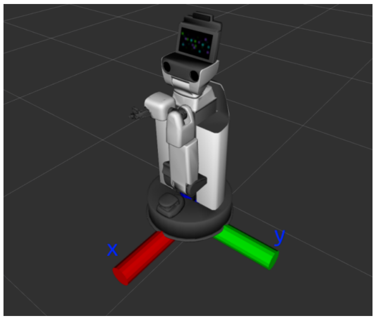


Fig. 3: `base_link` coordinate frame.

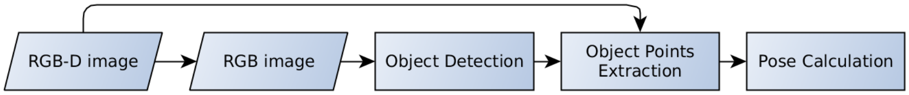


Fig. 4: Flowchart of the baseline method for grasp experiments.

3.1 Baseline pose estimation method

The first method assumes the grasp approach to be along the x-axis of the base_link coordinate frame, illustrated in figure 3. The object points are cropped from the detected rectangular regions as described above, and the grasping position $\mathbf{p} = (x, y, z)$ is calculated from the extracted 3D coordinates. If matrix B of shape $M \times 3$ contains all the 3D coordinates extracted using the detected object's bounding box, we either take the closest point along the x-axis of the base frame:

$$p = \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \min_{i=1}^{M} B_{i,1} \\ \frac{1}{M} \sum_{i=1}^{M} B_{i,2} \\ \frac{1}{M} \sum_{i=1}^{M} B_{i,3} \end{pmatrix} \tag{1}$$

, or the mean along all three axes:

$$p_{j=1,2,3} = \frac{1}{M} \sum_{i=1}^{M} B_{i,j} \tag{2}$$

3.2 GQCNN method

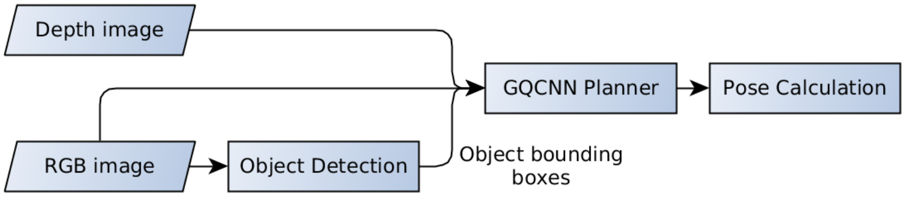


Fig. 5: Flowchart of the integrated GQCNN grasp planner [16].

As mentioned in section 2.3, the integrated GQCNN model [16] is trained on the synthetic Dex-Net 2.0 dataset to predict grasp robustness. The provided implementation assumes the grasp approach vector to align with the camera axis and predict the gripper's orientation, position and success probability using depth and RGB images. Figure 6 visualizes a successfully planned grasp returned from the GQCNN planner. In practice, the planner only detects grasps for one of the objects used for experimentation, taking between 30 seconds and a minute for planning each grasp. Experiments are therefore not performed using the GQCNN planner.

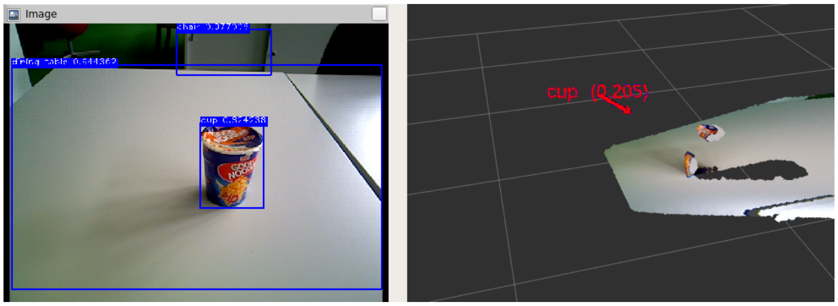
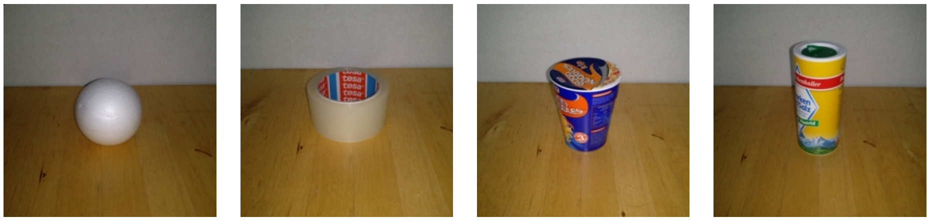


Fig. 6: A successful GQCNN grasp plan. Arrow and number on the right indicate grasp pose and quality returned from the GQCNN planner.

4 Experiments

4.1 Experimental Setup



(a) Styrofoam ball (b) Duct tape (c) Noodles box (d) Salt container

Fig. 7: Objects selected for the experiments.

Two sets of experiments are performed for the variants of the baseline method described in 3.1. Before each grasp, the robot is moved to a marked position facing a grasp surface, (in this case a dining table). In the `base_link` frame, objects further 0.9m in the positive x-axis and lower than 0.75m in the positive z-axis are ignored. We use the SSD model trained on the COCO dataset [3] for detection. Arm collisions and object slipping from the gripper are counted as failures. Only one of the objects in figure 7 is grasped at a time.

[3] http://cocodataset.org/

4.2 Results and discussion

Object	Mean x		Minimum x	
	Success	Failure	Success	Failure
Salt	17	3	16	4
Ball	8	12	15	5
Noodle box	16	4	12	8
Duct tape	7	13	13	7

Table 2: Results of the grasp experiments. The left and right columns contain results from using the mean and minimum x coordinates (`base_link frame`) for estimating the grasp pose respectively.

Table 2 shows the grasp success/failure results of the experiments. During the experiments, many of the failures while grasping the styrofoam ball is caused by the gripper pushing on the ball and making it roll forward (an example is shown in figure 8). For this object, grasping using the minimum coordinate along the x-axis proves to be much more reliable. Low objects like the roll of duct tape give low estimates for the z coordinate, causing many failures via collisions between the arm and the table. This suggests that a different approach vector from above may perform better for such objects. Many of the failures also occur because of slipping while the arm is moving back, which suggests that verification from the force sensors may boost grasp reliability.

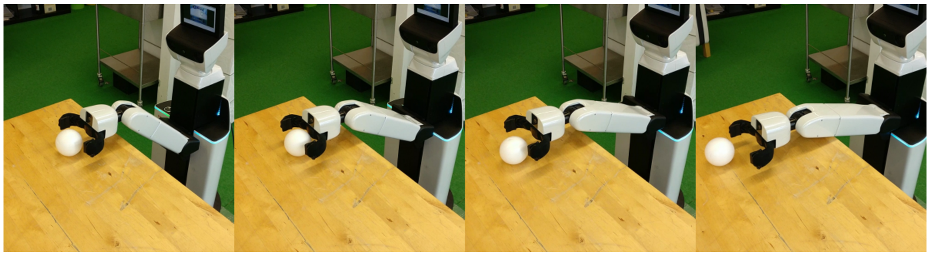


Fig. 8: A failure instance where the gripper pushes on the ball and it rolls forward.

5 Conclusions

5.1 Contributions

The first contribution of this work is a detailed review of recent advances in aspects most relevant to generating data for training a grasp evaluation models, namely feature extraction from perceptual data, object-grasp representation, grasp evaluation metrics, and data generation techniques. Additionally, five recent, prominent approaches to data synthesis for grasp evaluation are examined, and their solutions for each of the four aspects mentioned above are summarized in table 1.

The second contribution of this project is the implementation of a full grasping pipeline, from perceiving objects to grasp execution. Two pose estimation methods are implemented, serving as baselines for experimenting and comparing with more advanced grasp planning techniques.

The review of recent approaches to grasp data synthesis demonstrates their limitations either in dataset size or by using theoretical approaches to generate data labels, suggesting possible extensions and improvements with larger human grasp experience database [24] or more advanced feature extraction methods [27].

5.2 Future work

Several extensions and optimizations for the current implementation of the grasping pipeline are possible. First, the grasp execution implementation can be extended to allow the robot to grasp from different directions. Next, the detection model for the SSD architecture can also be fine-tuned to perform better in detecting RoboCup@Home objects. The grasp pipeline will also be tested on the Care-o-bot 3 [4], the other robot platform available to the project. Nearest neighbor algorithms can also be used to improve object pose estimation from the points extracted from RGB-D clouds. Surface normal calculations can also be implemented to replace the grasp pose calculation provided with the GQCNN software. Specifically, the pixel coordinate of the detected object can be directly transformed to its corresponding 3D coordinate, and the surface normal around this point can be used as the grasp's approach vector. This surface normal can also be used as the approach vector for the implemented pose estimation algorithms.

Furthermore, several of the approaches reviewed in chapter 2 can be integrated. Particularly, the shape completion technique introduced by Varley et al. [27], can be used as a prior for Gualtieri et al. [10] sampling of the occluded points for their 12-channel 2D representation. Training on the new human grasp experience dataset by Saudabayev et al. [24] can also be examined for a direct comparison with training on data synthesized using analytical grasp metrics and simulation. Techniques to introduce task awareness into data generation can also be examined, i.e how the knowledge of the task to be performed with the manipulated object can be encoded into the data synthesis process.

References

1. Bo, L., Ren, X., Fox, D.: Unsupervised Feature Learning for RGB-D Based Object Recognition, pp. 387–402. Springer International Publishing, Heidelberg (2013). https://doi.org/10.1007/978-3-319-00065-7_27

[4] https://www.care-o-bot.de/en/care-o-bot-3.html

2. Bohg, J., Johnson-Roberson, M., Len, B., Felip, J., Gratal, X., Bergstrm, N., Kragic, D., Morales, A.: Mind the gap - robotic grasping under incomplete observation. In: Zheng, Y.F. (ed.) 2011 IEEE International Conference on Robotics and Automation. pp. 686–693 (May 2011). https://doi.org/10.1109/ICRA.2011.5980354
3. Bohg, J., Morales, A., Asfour, T., Kragic, D.: Data-driven grasp synthesis: A survey. IEEE Transactions on Robotics **30**(2), 289–309 (April 2014). https://doi.org/10.1109/TRO.2013.2289018
4. Ciocarlie, M.T., Allen, P.K.: Hand posture subspaces for dexterous robotic grasping. The International Journal of Robotics Research **28**(7), 851–867 (2009). https://doi.org/10.1177/0278364909105606
5. Detry, R., Ek, C.H., Madry, M., Piater, J., Kragic, D.: Generalizing grasps across partly similar objects. In: Bicchi, A. (ed.) 2012 IEEE International Conference on Robotics and Automation. pp. 3791–3797 (May 2012). https://doi.org/10.1109/ICRA.2012.6224992
6. Eigen, D., Fergus, R.: Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In: Ikeuchi, K., Schnrr, C., Sivic, J., Vidal, R. (eds.) 2015 IEEE International Conference on Computer Vision (ICCV). pp. 2650–2658 (Dec 2015). https://doi.org/10.1109/ICCV.2015.304
7. Eitel, A., Springenberg, J.T., Spinello, L., Riedmiller, M., Burgard, W.: Multimodal deep learning for robust rgb-d object recognition. In: Burgard, W. (ed.) 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 681–687 (Sept 2015). https://doi.org/10.1109/IROS.2015.7353446
8. Goldfeder, C., Allen, P.K.: Data-driven grasping. Autonomous Robots **31**(1), 1–20 (Jul 2011). https://doi.org/10.1007/s10514-011-9228-1
9. Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., Chen, T.: Recent advances in convolutional neural networks. Pattern Recognition **77**, 354 – 377 (2018). https://doi.org/10.1016/j.patcog.2017.10.013
10. Gualtieri, M., ten Pas, A., Saenko, K., Platt, R.: High precision grasp pose detection in dense clutter. In: et al., W.B. (ed.) 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 598–605 (Oct 2016). https://doi.org/10.1109/IROS.2016.7759114
11. Gupta, S., Girshick, R., Arbeláez, P., Malik, J.: Learning rich features from rgb-d images for object detection and segmentation. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) Computer Vision – ECCV 2014. pp. 345–360. Springer International Publishing, Cham (2014)
12. Jiang, Y., Moseson, S., Saxena, A.: Efficient grasping from rgbd images: Learning using a new rectangle representation. In: Bicchi, A. (ed.) IEEE International Conference on Robotics and Automation (ICRA). pp. 3304–3311 (May 2011). https://doi.org/10.1109/ICRA.2011.5980145
13. Kappler, D., Bohg, J., Schaal, S.: Leveraging big data for grasp planning. In: Okamura, A. (ed.) IEEE International Conference on Robotics and Automation (ICRA). pp. 4304–4311 (May 2015). https://doi.org/10.1109/ICRA.2015.7139793
14. Lenz, I., Lee, H., Saxena, A.: Deep learning for detecting robotic grasps. The International Journal of Robotics Research **34**(4-5), 705–724 (2015). https://doi.org/10.1177/0278364914549607
15. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) Computer Vision – ECCV 2016. pp. 21–37. Springer International Publishing, Cham (2016)

16. Mahler, J., Liang, J., Niyaz, S., Laskey, M., Doan, R., Liu, X., Ojea, J.A., Goldberg, K.: Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. CoRR **abs/1703.09312** (2017), http://arxiv.org/abs/1703.09312
17. Matamoros, M., Rascon, C., Hart, J., Holz, D., van Beek, L.: Robocup@home 2018: Rules and regulations. http://www.robocupathome.org/rules/2018_rulebook.pdf (2018)
18. Padalkar, A.: Dynamic motion primitives. Tech. rep., Hochschule Bonn-Rhein-Sieg (2018)
19. Porzi, L., Bul, S.R., Penate-Sanchez, A., Ricci, E., Moreno-Noguer, F.: Learning depth-aware deep representations for robotic perception. IEEE Robotics and Automation Letters **2**(2), 468–475 (April 2017). https://doi.org/10.1109/LRA.2016.2637444
20. Qi, C.R., Su, H., Niener, M., Dai, A., Yan, M., Guibas, L.J.: Volumetric and multi-view cnns for object classification on 3d data. In: Agapito, L., Berg, T., Kosecka, J., Zelnik-Manor, L. (eds.) 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5648–5656 (June 2016). https://doi.org/10.1109/CVPR.2016.609
21. Roa, M.A., Suárez, R.: Grasp quality measures: Review and performance. Autonomous Robots **38**(1), 65–88 (Jan 2015). https://doi.org/10.1007/s10514-014-9402-3
22. Rubert, C., Kappler, D., Morales, A., Schaal, S., Bohg, J.: On the relevance of grasp metrics for predicting grasp success. In: Maciejewski, T. (ed.) 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 265–272 (Sept 2017). https://doi.org/10.1109/IROS.2017.8202167
23. Sahbani, A., El-Khoury, S., Bidaud, P.: An overview of 3d object grasp synthesis algorithms. Robotics and Autonomous Systems **60**(3), 326–336 (Mar 2012). https://doi.org/10.1016/j.robot.2011.07.016
24. Saudabayev, A., Rysbek, Z., Khassenova, R., Varol, H.A.: Human grasping database for activities of daily living with depth, color and kinematic data streams. Scientific Data **5**, 180101 (May 2018), data Descriptor
25. Shimoga, K.: Robot grasp synthesis algorithms: A survey. The International Journal of Robotics Research **15**(3), 230–266 (1996). https://doi.org/10.1177/027836499601500302
26. Su, H., Maji, S., Kalogerakis, E., Learned-Miller, E.: Multi-view convolutional neural networks for 3d shape recognition. In: Ikeuchi, K., Schnrr, C., Sivic, J., Vidal, R. (eds.) 2015 IEEE International Conference on Computer Vision (ICCV). pp. 945–953 (Dec 2015). https://doi.org/10.1109/ICCV.2015.114
27. Varley, J., DeChant, C., Richardson, A., Ruales, J., Allen, P.: Shape completion enabled robotic grasping. In: Maciejewski, T. (ed.) 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 2442–2447 (Sept 2017). https://doi.org/10.1109/IROS.2017.8206060
28. Weisz, J., Allen, P.K.: Pose error robust grasping from contact wrench space metrics. In: Bicchi, A. (ed.) 2012 IEEE International Conference on Robotics and Automation. pp. 557–562 (May 2012). https://doi.org/10.1109/ICRA.2012.6224697