Robotic Manipulation Based on 3D Vision: A Survey

Huahua Lin†
College of Information Engineering
China Jiliang University
Hangzhou, China
islinhuahua@163.com

ABSTRACT

Grasping has long been studied in the field of robotics. In this paper, we divide the process of robotic grasp into sensing and control. In terms of sensing, 2D vision based sensing relies on accurate feature matching and object surface texture features, resulting in poor performance in the complex environment with occlusion. By contrast, some sensors based on 3D vision are more robust to noise. Processing point clouds in a deep learning method can achieve high accuracy as well as reducing the computation time compared with those using cost volume regularization. For the control part, the traditional trajectory motion methods are limited to generalization and grasping with high degrees of freedom. On the contrary, the methods of reinforcement learning can improve the grasping strategy in the continuous interaction with the environment. We propose some commonly used benchmarks and simulation platforms for simulation experiment using reinforcement learning.

CCS CONCEPTS

•Computing methodologies~Artificial intelligence~Computer vision~Computer vision tasks~Vision for robotics

KEYWORDS

robotic manipulation, deep learning, reinforcement learning

1 Introduction

One of the classic and fundamental problems in robotic research is grasping. From a human standpoint, this is almost a thoughtless process to pick up an object, but for robotic grippers, the process from the sensing of the object to the grasp control of the trajectory planning is normally detailed and complicated.

The role of the vision system in grasping is to identify and locate the target and provide the information of the pose of the target object. Among them, the accuracy of pose estimation is related to the success rate and accuracy of grasping, which is a very important parameter. Due to the rapid development of 3D sensors and the strong demand rapidly derived from many application scenarios, 3D vision is being widely studied and developed in robotics on the basis of 2D vision recently. Although analyses based on 2D images [1, 2] have been relatively mature in planar features of objects, but they have limitations in capturing the depth of the target and its surrounding environment, which cannot satisfy the grasp in the space. The occurrence of 3D vision can make up for these imperfections. It can give the depth information [3, 4] of the target object or the point cloud information [5, 6] on the object surface. Therefore, the shift from 2D to 3D, which is a natural thing to do, allows the robotic grippers to grasp objects in more complex environments with greater precision. Now, for better robotic manipulation, compared with using dense depth information alone, a trained 3D CNN which uses additional sparse tactile information [7] is demonstrated the better quality in predicting and filling in the occlusion of an object.

In the process of computing the grasp plan to perform the physical grasp, the primary task is to define a grasp success criterion to judge whether a stable grasping is formed or not. Force closure is considered to be the simplest and basic grasping stability analysis [8] and a binary analysis of grasp robustness is provided by it. In order to improve the quality of the grasp, some researches have considered the position of the gripper contact point. [9] has increased the point of contact to an acceptable set from which any pair of touch points selected can be successfully used in grasp. [10, 11, 12] applied this method to complex 2D and simple 3D models, while [13] further extended it to complex 3D models. However, these methods do not take into account the effects of calibration errors and the uncertainty of the target pose, and that will lead to the contact sets actually be very different throughout the grasp process, rather than falling near the planned contact points on the target as they assume. Caging [14] is a geometric constraint method to replace the force constraint above. The main idea is that an object can be successful grasped as long as it is within the restriction of a gripper. Although this approach does not require force control, a reliable cage requires a large

PRIS 2020, July 30-August 2, 2020, Athens, Greece

ACM ISBN 978-1-4503-8769-9/20/07...$15.00
https://doi.org/10.1145/3415048.3416116

number of complex structures and is difficult to grasp deformable objects.

Recent motion planning methods are normally data-driven. They use learning-based approaches [15] which include learning from demonstration (LFD) and reinforcement learning (RL)-based techniques [16], however, in this paper, we mainly focus on the RL-based techniques. Reinforcement learning has been widely used in the optimization of grasp strategies [17, 18, 19] compared with prior learning-based methods which cannot reason about the sequence of grasping [20]. However, they mainly focus on the position control, meaning that to reach a certain position regardless of how much force will be applied to the interaction with the environment, which is often dangerous in case of position error. Force control provides an alternative, but in order to make the manipulator more free and safer, it is best to combine two of them [21]. Generalization is often used to test these learning-based grasping methods. To avoid the extreme approach of training millions of data to achieve generalization of unseen objects [22], off-policy reinforcement learning methods often do a better job than on-policy. The benchmark tasks of various Q-function estimation methods are evaluated in [23]. The evaluation results show that among those model-free and off-policy RL methods, the generalization performance of DQL is best in the environment with less data, while the Monte Carlo and modified Monte Carlo methods are best in the environment with the most data.

To our best knowledge, no one has done a relatively comprehensive analysis of grasping. The main contributions of our paper consist of two aspects: surveying the current sensing techniques based on 3D vision, surveying and comparing two kinds of grasping control algorithms. The structure of the rest of this paper is as follows: In Section 2, we explain the various aspects required to complete a grasping task and how they fit together from a comprehensive perspective. We then discuss the algorithm involved from two parts in Section 3: The first part is focused on using 3D vision to perceive information that is useful before grasping an object, and then the grasp control methods, including both conventional and reinforcement learning methods. In Section 4, we show the comparison of different methods and give a conclusion of them.

2 Problem Statement

The grasping tasks can be implemented via two stages: sensing and control. As is shown in Figure 1, Robotic arms such as KUKA LBR iiwa with a paralleled or unparalleled two-finger gripper are able to perform grasping tasks through these two stages. In terms of sensing techniques, there are 2D cameras to perform RGB images, 3D sensors such as Kinect to perform point data and force sensors be utilized to adjust the position uncertainty perceived in 3D. Control methods contain learning-based control such as reinforcement learning based methods compared with model-based traditional control methods. It doesn't matter if the grasp strategy is just one sensing and then perform direct control towards the object, or continuous sensing after each small step of control. The grasping tasks can be divided in two parts:

1. Regular grasping. This grasping task requires to achieve generalization. In the workspace, 800 randomly generated objects with different kinds of shapes are used in the training, and 100 objects that have never been seen before are tested. The gripper is expected to successful grasp 5 non-blocking object during each episode.
2. Grasp a target in clutter. This grasping task requires to grasp a specified object in a cluttered scene with some occlusion. In this process, the manipulator might accidentally touch other objects which is very challenging for grasping strategy. Only if the target is grasped successfully, the robot will be given reward.
3. Algorithm

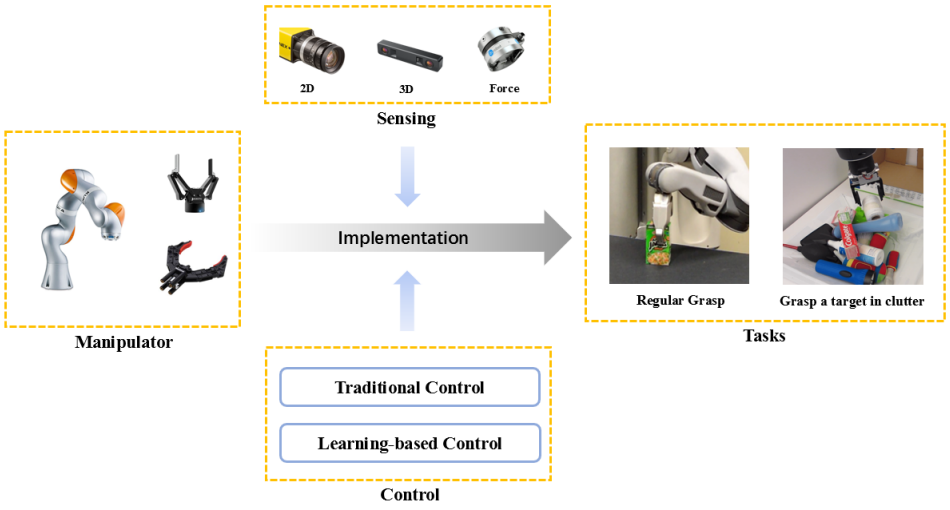

Figure 1: The macroscopic process of completing grasping tasks. A robot arm with a compatible two-finger gripper implements different kinds of grasping tasks by the assistance of sensing and control.

3.1 Sensing based on 3D vision

Estimating the poses of objects is necessary since completing grasp tasks is relatively complex for robotic arms. The first task is for different shapes of objects, it may be an irregular polyhedron or a ball, then the way of grasping will be different. The second task is to grasp the target object in a cluttered scene, which usually results in the inevitable occlusion between the objects. All these complex internal or external factors explain the indispensability of accurate object 6D pose estimation. Therefore, in an ideal case, the estimated results should be robust to the noise of the sensor, the inevitable occlusion in complex environments, and other influence factors so as to ensure a certain grasp speed and success rate.

Previously, researchers tried to use single 2D image to estimate the 6D pose of objects. Methods that appear first focus on finding sufficient correspondence between 3D model features and 2D image features [24, 25], and then use the Perspective-n-Point (PnP) solver to estimate the pose that most conforms to these correspondence relations. In order to eliminate the influence of noise as much as possible, it is common practice to utilize non-linear least-squares minimization techniques [26] to adjust the estimated value. However, there is still an inaccurate feature matching which leads to the failure of the final grasp, and the occlusion problem between objects has not been solved. Although the accuracy of feature matching is improved to some extent for multiple 2D images taken from different angles, these methods are limited to good performance in feature matching only when the object has a rich texture. For the simplest example of an industrial application, it is difficult to precise estimate a black surface object's pose from a black conveyor belt.

Fortunately, the advent of depth cameras greatly solves the problem of poor feature matching performance when the texture of the object is not rich [27, 28]. Three-dimensional sensor data is usually in the form of point clouds, they can be obtained from the depth information captured by the RGB-D sensor through coordinate conversion, or from LIDAR device, which provide great convenience and high accuracy in estimating the object's 6D pose, even under poor lighting conditions. The initial method of pose estimation based on point cloud is to extend the search for 2D to 3D correspondence to the search for 3D to 3D correspondence. The principle is similar to the 2D image-based approach: Features are extracted from the model and object and then matched to estimate the object's 6D pose. It is worth noting that the point cloud registration is a process from coarse to fine. Widely used algorithms like Iterative Closest Point (ICP) [29] and its variants are normally employed to minimize the difference between two clouds of points to acquire fine registration. In addition, by combining these approaches with the robust estimation techniques such as RANSAC [30], a strong robustness against noise can be achieved. Nevertheless, these complex optimization steps are often time consuming thus cannot be applied in real-time interactions with the environment, and cannot be refined together with the final estimate.

In recent years, due to the rapid development of deep learning, a number of pose estimation algorithms based on CNN architectures have emerged. In order to better utilize the global semantic information of the scene, multi-scale 3D CNNs used for cost volume regularization are applied to predict the depth map. However, as the cost volume resolution increases, memory requirements increase exponentially. Chen et al. [31] directly represent the target scene as point clouds, which can save unnecessary computation and gradually approach the exact position. In its course depth estimation network, the memory usage is only 1/20 of MVSNet. In addition, to refine the depth map, they proposed PointFlow module: Generate a series of hypothetical points for the unprojected point, construct a directed graph from these points, and further extract neighborhood features by performing edge convolution. Then the offset position of the unprojected point is determined by MLP, and the offset vector is obtained by averaging the weight of each assumption point. Experiments show that this network structure based on point cloud can achieve higher accuracy, more computational efficiency and more flexibility compared with previous counterparts.

Finding a suitable grasping position for a 3D object is also difficult because the number of possible grasping points is infinite. This position is expected to have stability, task compatibility and adaptability targeted to novel objects and complex scenarios. The quality of grasping can be measured by the position of the contact points previously mentioned. Sahbani et al. [32] have divided approaches into two parts: empirical (data-driven) and analytic. For the former, they typically require a large collection of manually labelled data to train the network, so they only work well on existing data sets. What's worse, it's not realistic to spend a lot of time and effort on manual labelling in industrial applications. In contrast, analytical methods provide many advanced models for measuring grasp quality and constructing accurate object models. Both GPD [33] and PointNetGPD [34] construct a Darboux frame and search its 6D neighborhood in order to sample the suitable grasp regions. However, it is difficult to calculate the normals of thin objects under the interference of noise. Compared with Darboux frame based methods, Qin et al. [35] proposed a new gripper contact model based on force closure analysis to find viable grasps, by detecting all feasible contact pairs for every object and then remove unfeasible one in the cluttered scene. This novel model achieves better performance especially in generating a pair of suitable contact points for the thin wall.

3.2 Grasping control

Motion planning is to find a path from the manipulator to the grasp point position of the object. How to achieve the path of fixed grasping pose is an important research content. Therefore, we group the grasping control methods into two parts below, from Dynamic Movement Primitives methods to reinforcement learning methods.

3.2.1 Conventional controller. The traditional motion planning methods are based on open-loop learning, that is, the collection of data based on static data and labels and the optimization of model

are two independent processes. One of the trajectory planning methods is based on Dynamic Movement Primitives (DMPs) [36]. However, these methods rely heavily on the accuracy of the pose estimation result, which results in less robust. In addition, they do not have a strong generalization ability to grasp diverse objects. In order to obtain the optimal motion path, the trajectory planning method is usually combined with the optimization method. However, the process of utilizing kinematics principle to design the optimal path is relatively complicated, especially for grasping with high degree of freedom. For achieving high-dimensional manipulator, such as humanoid grasp, reinforcement learning based approaches achieve state-of-the-art performance.

3.2.2 Reinforcement learning based controller. Reinforcement learning is a typical representative of the closed-loop learning paradigm of artificial intelligence compared with those of open-loop. It focuses on the interaction with the environment to obtain feedback signals that reflect the achievement of real goals. In the application of robot grasping, reinforcement learning will emphasize the dynamic and long-term effects of trial-and-error learning and sequential decision-making behavior, so that the grasping accuracy for different objects is progressively higher while human effort is greatly reduced. Kalashnikov et al. [37] proposed a method called QT-Opt, a scalable off-policy reinforcement learning framework based on a continuous generalization of Q-learning, which can refine the grasping action and dynamic respond to disturbance by self-supervised feedback. In addition to realizing the generalization ability of unseen objects with high success rate, its closed-loop dynamic strategy also enhances the ability of long-term learning and reasoning. Specifically, on the one hand, for objects with strange stereo shape or smooth appearance, the robot will re-analyze the angle and positioning of grasping, and then firmly grasp; On the other hand, in a cluttered scene, if the target object's suitable position for grasping is blocked by other objects, the robot will take the initiative to separate it and then perform the grasp.

Table 1: Benchmark and simulation platform for manipulation tasks.

Name	Description	Reference
Mujoco	Benchmark for reinforcement learning on manipulation	[23][22]
Pybullet	Vision-based manipulation tasks	[38][39]

4 Comparison and discussion

In this paper, we summarize and compare the sensing methods based on 3D vision, including point cloud based and deep learning-based methods to estimate the pose of objects, as well as the grasp point estimation based on data-driven and analytic methods. The point cloud is more accurate than the image to estimate the pose of the objects, which solves the problem of insufficient object texture. We believe that using 3D visions to sense the objects would be the trend. Furthermore, the deep learning-based method to optimize the point cloud can greatly improve the computing efficiency. For the control part, we compare the trajectory motion method based on DMPs with that based on reinforcement learning. After our research and comparison, reinforcement learning methods hold the promising to learn the general manipulation skills among a wide range of manipulation tasks. However, currently RL algorithms require a large amount of interactions with environment, which is not allowed in real-world tasks. Therefore, the process of learning in simulation at first and then transferred to real-world tasks is necessary. We surveyed several commonly used benchmarks and simulation platforms for manipulation tasks, as shown in Table 1.

REFERENCES

[1] Ashutosh Saxena, Justin Driemeyer, and Andrew Y Ng. Robotic grasping of novel objects using vision. *The International Journal of Robotics Research*, 27(2):157–173, 2008.

[2] Gary M Bone and Yonghui Du. Multi-metric comparison of optimal 2d grasp planning algorithms. In *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164)*, volume 3, pages 3061–3066. IEEE, 2001.

[3] Saurabh Gupta, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Aligning 3d models to rgb-d images of cluttered scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4731–4740, 2015.

[4] Andy Zeng, Kuan-Ting Yu, Shuran Song, Daniel Suo, Ed Walker, Alberto Rodriguez, and Jianxiong Xiao. Multi-view self-supervised deep learning for 6d pose estimation in the amazon picking challenge. In *2017 IEEE international conference on robotics and automation (ICRA)*, pages 1386–1383. IEEE, 2017.

[5] Matei Ciocarlie, Kaijen Hsiao, Edward Gil Jones, Sachin Chitta, Radu Bogdan Rusu, and Ioan A S¸ucan. Towards reliable grasping and manipulation in household environments. In *Experimental Robotics*, pages 241–252. Springer, 2014.

[6] Stefan Hinterstoisser, Stefan Holzer, Cedric Cagniart, Slobodan Ilic, Kurt Konolige, Nassir Navab, and Vincent Lepetit. Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes. In *2011 international conference on computer vision*, pages 858–865. IEEE, 2011.

[7] David Watkins-Valls, Jacob Varley, and Peter Allen. Multi-modal geometric learning for grasping and manipulation. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 7339–7345. IEEE, 2019.

[8] J Kenneth Salisbury and B Roth. Kinematic and force analysis of articulated mechanical hands. 1983.

[9] Van-Duc Nguyen. Constructing force-closure grasps. *The International Journal of Robotics Research*, 7(3):3–16, 1988.

[10] Jean Ponce, Darrell Stam, and Bernard Faverjon. On computing two-finger force-closure grasps of curved 2d objects. *The International Journal of Robotics Research*, 12(3):263–273, 1993.

[11] Jean Ponce and Bernard Faverjon. On computing three-finger force-closure grasps of polygonal objects. *IEEE Transactions on robotics and automation*, 11(6):868–881, 1995.

[12] Jean Ponce, Steve Sullivan, Attawith Sudsang, Jean-Daniel Boissonnat, and Jean-Pierre Merlet. On computing four-finger equilibrium and force-closure grasps of polyhedral objects. *The International Journal of Robotics Research*, 16(1):11–35, 1997.

[13] Máximo A Roa and Raúl Suárez. Computation of independent contact regions for grasping 3-d objects. *IEEE Transactions on Robotics*, 25(4):839–850, 2009.

[14] Alberto Rodriguez, Matthew T Mason, and Steve Ferry. From caging to grasping. *The International Journal of Robotics Research*, 31(7):886–900, 2012.

[15] Jing Xu, Zhimin Hou, Zhi Liu, and Hong Qiao. Compare contact model-based control and contact model-free learning: A survey of robotic peg-in-hole assembly strategies. *arXiv preprint arXiv:1904.05240*, 2019.

[16] Jing Xu, Zhimin Hou, Wei Wang, Bohao Xu, Kuangen Zhang, and Ken Chen. Feedback deep deterministic policy gradient with fuzzy reward for robotic multiple peg-in-hole assembly tasks. *IEEE Transactions on Industrial Informatics*, 15(3):1658–1667, 2018.

[17] Jens Kober and Jan R Peters. Policy search for motor primitives in robotics. In *Advances in neural information processing systems*, pages 849–856, 2009.
[18] Petar Kormushev, Sylvain Calinon, and Darwin G Caldwell. Robot motor skill coordination with em-based reinforcement learning. In *2010 IEEE/RSJ international conference on intelligent robots and systems*, pages 3232–3237. IEEE, 2010.
[19] Freek Stulp, Evangelos Theodorou, Jonas Buchli, and Stefan Schaal. Learning to grasp under uncertainty. In *2011 IEEE International Conference on Robotics and Automation*, pages 5703–5708. IEEE, 2011.
[20] Ian Lenz, Honglak Lee, and Ashutosh Saxena. Deep learning for detecting robotic grasps. *The International Journal of Robotics Research*, 34(4-5):705–724, 2015.
[21] Mrinal Kalakrishnan, Ludovic Righetti, Peter Pastor, and Stefan Schaal. Learning force control policies for compliant manipulation. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4639–4644. IEEE, 2011.
[22] Sergey Levine, Peter Pastor, Alex Krizhevsky, and Deirdre Quillen. Learning hand-eye coordination for robotic grasping with large-scale data collection. In *International symposium on experimental robotics*, pages 173–184. Springer, 2016.
[23] Deirdre Quillen, Eric Jang, Ofir Nachum, Chelsea Finn, Julian Ibarz, and Sergey Levine. Deep reinforcement learning for vision-based robotic grasping: A simulated comparative evaluation of off-policy methods. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6284–6291. IEEE, 2018.
[24] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Epnp: An accurate o (n) solution to the pnp problem. *International journal of computer vision*, 81(2):155, 2009.
[25] Daniel F Dementhon and Larry S Davis. Model-based object pose in 25 lines of code. *International journal of computer vision*, 15(1-2):123–141, 1995.
[26] Donald W Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the society for Industrial and Applied Mathematics*, 11(2):431–441, 1963.
[27] Eric Brachmann, Alexander Krull, Frank Michel, Stefan Gumhold, Jamie Shotton, and Carsten Rother. Learning 6d object pose estimation using 3d object coordinates. In *European conference on computer vision*, pages 536–551. Springer, 2014.
[28] Radu Bogdan Rusu, Gary Bradski, Romain Thibaux, and John Hsu. Fast 3d recognition and pose using the viewpoint feature histogram. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2155–2162. IEEE, 2010.
[29] Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In *Sensor fusion IV: control paradigms and data structures*, volume 1611, pages 586–606. International Society for Optics and Photonics, 1992.
[30] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.
[31] Rui Chen, Songfang Han, Jing Xu, and Hao Su. Point-based multi-view stereo network. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1538–1547, 2019.
[32] Anis Sahbani, Sahar El-Khoury, and Philippe Bidaud. An overview of 3d object grasp synthesis algorithms. *Robotics and Autonomous Systems*, 60(3):326–336, 2012.
[33] Andreas ten Pas, Marcus Gualtieri, Kate Saenko, and Robert Platt. Grasp pose detection in point clouds. *The International Journal of Robotics Research*, 36(13-14):1455–1473, 2017.
[34] Hongzhuo Liang, Xiaojian Ma, Shuang Li, Michael Görner, Song Tang, Bin Fang, Fuchun Sun, and Jianwei Zhang. Pointnetgpd: Detecting grasp configurations from point sets. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 3629–3635. IEEE, 2019.
[35] Yuzhe Qin, Rui Chen, Hao Zhu, Meng Song, Jing Xu, and Hao Su. S4g: Amodal single-view single-shot se (3) grasp detection in cluttered scenes. *arXiv preprint arXiv:1910.14218*, 2019.
[36] Stefan Schaal. Dynamic movement primitives-a framework for motor control in humans and humanoid robotics. In *Adaptive motion of animals and machines*, pages 261–280. Springer, 2006.
[37] Dmitry Kalashnikov, Alex Irpan, Peter Pastor, Julian Ibarz, Alexander Herzog, Eric Jang, Deirdre Quillen, Ethan Holly, Mrinal Kalakrishnan, Vincent Vanhoucke, et al. Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation. *arXiv preprint arXiv:1806.10293*, 2018.
[38] Andy Zeng, Shuran Song, Johnny Lee, Alberto Rodriguez, and Thomas Funkhouser. Tossingbot: Learning to throw arbitrary objects with residual physics. *arXiv preprint arXiv:1903.11239*, 2019.
[39] Jan Matas, Stephen James, and Andrew J Davison. Sim-to-real reinforcement learning for deformable object manipulation. *arXiv preprint arXiv:1806.07851*, 2018.