Evaluation of ML-Based Grasping Approaches in the Field of Automated Assembly

Oliver Petrovic, Philipp Blanke, Manuel Belke, Eike Wefelberg,
Simon Storms and Christian Brecher

Abstract

Current trends in the manufacturing industry lead to high competitive pressure and requirements regarding process autonomy and flexibility in the production environment. Especially in assembly, automation systems are confronted with a high number of variants. Robot-based processes are a powerful tool for addressing these challenges. For this purpose, robots must be made capable of grasping a variety of diverse components, which are often provided in unknown poses. In addition to existing analytical algorithms, empirical ML-based approaches have been developed, which offer great potentials in increasing flexibility. In this paper, the functionalities and potentials of these approaches will be presented and then compared to the requirements from production processes in order to analyze the status quo of ML-based grasping. Functional gaps are identified that still need to be overcome in order to enable the technology for the use in industrial assembly.

Keywords

Robotics • Grasping • Assembly • Computer vision • Machine learning

O. Petrovic (✉) · P. Blanke · M. Belke · E. Wefelberg · S. Storms · C. Brecher
Laboratory for Machine Tools and Production Engineering (WZL) of RWTH Aachen University,
Steinbachstr. 19, 52074 Aachen, Germany
E-mail: O.Petrovic@wzl.rwth-aachen.de
URL: <http://www.wzl.rwth-aachen.de>

© The Author(s) 2022

T. Schüppstuhl et al. (eds.), *Annals of Scientific Society for Assembly, Handling and Industrial Robotics 2021*,
https://doi.org/10.1007/978-3-030-74032-0_28

1 Introduction and Motivation

In the manufacturing industry, a trend towards robot-based automation has been observed for years. From 2013 to 2018, the number of new robot installations has increased by an average of 19% per year [1]. Reasons for this are the rising quality standards and labor costs which lead to high competitive pressure. In countermove, economic automation is becoming more and more difficult as product life cycles become shorter and batch sizes smaller. This places great demands on the flexibility and autonomy of the technologies to handle this variety [2]. Especially in assembly, the degree of automation is often still very low, as the generation of variants is usually shifted as far back in the value chain as possible to final assembly in order to minimize its impact. Therefore, a key challenge is the flexible interaction of the robot with its environment. It must be able to handle a wide range of components, which are often fed in an unknown position and orientation, and this with an economical level of implementation effort. In recent years, the robotics and computer vision community has contributed a wide range of different approaches to solve the grasping problem. Analytical approaches consider kinematic and dynamic formulations in grasp synthesis [3]. However, these approaches are characterized by high computational complexity and cannot be generalized well to unknown objects, which is why the developed ML-based methods are promising approaches [4].

The aim of this paper is to provide an overview of these current approaches in research and to highlight the remaining challenges for their use in production, especially in assembly: In Sect. 2, state-of-the-art on ML-based grasping approaches in research is given. Section 3 analyzes the production requirements for the application, followed by the presentation of a derived integration approach of grasping into the digital process chain of assembly in Sect. 4. Finally, Sect. 5 identifies the gaps that need to be closed in order to implement an integration to meet the requirements.

2 State of the Art

Sensor-based perception of the environment are fundamental capabilities of a smart robot. In this regard, autonomous or partially autonomous grasping based on vision systems is one of the sub-disciplines of robotics that can contribute greatly towards improving the flexibility of robotic applications. According to Kumra et al. the vision-based grasping process can be seen as a sequence of three sub-steps: Grasp detection, trajectory planning and execution of the grasp [5]. This paper will mainly focus on the first step of this sequence, which again can be divided into three sub-problems shown in Fig. 1.

Fig. 1 Subtask classification of vision-based grasp detection systems [6]

2.1 Object Localization

The object localization task can be further divided into pure localization and localization including the detection of the class of the object. Since the manipulation requires spatial knowledge about the object, only the 3D localization methods will be described. To use these 3D localization methods, a RGB-D camera is used, that provides depth data in addition to the RGB image [6]. The pure localization task can be applied to simple objects like cubes or cylinders but the method can be improved by combining shape primitives with the use of triangular meshes to be able to map various types of objects Rusu et al. [7]. The localization of objects with no restriction regarding its shape is called salient object detection. Many approaches take the pixels of an image as inputs. Instead, salient feature vectors can be used as inputs to a CNN to learn the combination of different salient features for the recognition of salient objects [8]. Outputs of the detection of objects are the 3D bounding box and the class label of the object, which can either be detected sub-sequentially as in ImVoteNet [9] or at once by using a regression method like 3DSSD [10]. To further refine the position of the object, instance segmentation can be used. The starting point is the bounding box of the object, within which the 3D position of the object is detected. OccuSeg uses the occupancy to cluster segments despite partial occlusions [11].

2.2 Object Pose Estimation

The second subproblem is the Object Pose Estimation, where the 6D pose of the localized part must be determined. The degrees of freedom to be determined can be reduced by a predefined part feeding. Du et al. cluster the existing methods into three categories [6]. Firstly, there are the correspondence-based methods, in which corresponding feature points between captured image information and the object to be grasped are searched for. It is possible to utilize deep learning algorithms and to work with 2D RGB images like HybridPose [12] as well as with

3D point clouds like 3DMatch [13]. These methods are suitable if the object has a rich texture and geometric details.

The second group of algorithms are the template-based methods. A multitude of templates are labeled with corresponding 6D poses for the object to be grasped. If 2D images are used as in [14], the 6D problem is reduced to an image retrieval problem, because the image is only compared with a known set of 2D images of the object. If, on the other hand, 3D template-based methods are used, the recorded point cloud is directly compared with the 3D model of the object as in MaskedFusion [15]. In general, template-based methods are suitable especially if the object has few distinctive textures and geometric details.

The third category are the voting-based methods. Here, not the whole image is analyzed at once, but every single 2D pixel respectively every 3D point is considered separately and contributes a vote to the estimation. If the objects to be grasped have a high degree of occlusion, then voting-based methods can be effective. On the one hand, there are indirect voting-based methods in which the image points first contribute a vote for higher-level features from which the 6D pose can be indirectly derived. This is shown in YOLOff Gonzalez et al. [16]. On the other hand, direct voting-methods can be used, where the pixels vote directly for the 6D pose of the object as in DenseFusion [17].

2.3 Grasp Estimation

The goal of Grasp Estimation is to find a robust grasp pose. According to Du et al. the algorithms for Grasp Estimation can be divided into 2D planar grasps and 6D grasps [6]. The 2D planar grasp has two fixed axes of rotation, so that only the height of the plane, the position in the plane and the rotation around the normal vector are determined. The developed algorithms of both categories refer to either analytical or ML-based approaches. Due to the dependence on assumptions to be made (friction, object stiffness, object complexity etc.) analytical approaches in practice do not generalize well over new objects [18].

One of the ML based grasp methods is the project Dex-Net presented by Mahler et al.. The input is a recorded point cloud, which is evaluated by a CNN regarding the grasp quality of all grasping candidates. Zeng et al. performs a pixel-wise evaluation of the grasp affordance for different grasping primitive actions and perform the end-effector position and orientation with the highest affordance. Furthermore, the project Form2Fit [21] not only deals with grasping new objects but also with placing them in the desired position. A trained fully convolutional network (FCN) detects correspondences between the object surface and the shape of the target position.

3 Production Requirements on Vision-Based Grasping

In order to evaluate the industrial applicability of today's algorithms for ML-based grasping, the production requirements for such a system must first be analyzed. In this chapter, these requirements are categorized into six categories (Fig. 2) to identify gaps in the usability and functionality of today's solutions.

First, the required performance of the system is derived directly from quality and productivity requirements, which can be translated into the required precision and speed of the grasp detection. The second category is the robustness of the system against external influences such as poor lighting conditions, humidity and a dynamic image background. Another important factor are the components to be grasped. On the one hand, the components themselves, i.e. their variance, dimensions, shape, transparency and surface, and on the other hand the way they are fed to the process, has to be considered. The feeding can vary in the level of order, the degree of occlusion and hooking as well as the distance between the components. The hardware is required to provide the necessary computing power for the execution of the algorithms in a cost-effective manner in order to enable a profitable operation of the system. The available interfaces of the software as well as the range of compatible hardware like robots, grippers and sensors like cameras have a great influence on the integratability and transferability of the solution. Finally, required data sets and programming efforts should be mentioned, which directly impact the implementation effort and the competence hurdle for the programmer. The number, scope and quality of compatible data sets for training the algorithms, on the other hand, have great influence on the performance of the system. Furthermore, to achieve good industrializability, it must be possible to integrate existing product and process data. Physical component data, functional

Fig. 2 Clustering of production requirements on vision-based grasping

surfaces and the requirements of subsequent process steps are some examples of important parameters when selecting a grasp. In addition, parameters of the equipment, such as force limits and workspaces of robots and grippers, must also be taken into account.

4 Integration of ML-based Grasping in Assembly Processes

To meet these requirements, the three presented steps of a grasping system need to be embedded into a novel end-to-end system and closely linked to the digital process chain and its corresponding product lifecycle. Such a concept is proposed in Fig. 3. The product lifecycle can be sub-divided into engineering, production planning, production, usage and recycling. During the engineering phase, the product is designed and can be disassembled into product specifications, drawings, CAD models of each single part. In the production planning, the data of the engineering phase is used to plan the production and especially the process and assembly sequence. The grasping system is implemented in this phase. In the subsequent production, the actual gripping process is carried out. Throughout the entire life cycle, product and process data must be made available in accessible formats via a

Fig. 3 Integration of ML-based grasping approaches in the digital process chain

central digital process chain which is an important enabler for the seamless integration of engineering data into the robot-based assembly process.

Before the individual components are selected, the overall performance and robustness of the system required to fulfill the task at hand have to be defined. The precision and speed required to assemble the components is determined by the product while the light and background as robustness parameters are given by the environment. This narrows down the suitable algorithms to perform the tasks. Another important factor for selecting the algorithms is defined by the programming requirements. In order to make the system versatile, it should be intuitively operable and have enough autonomy to make the supervision by a human operator redundant.

The object localization task requires RGB and depth images belonging to a CAD model. With the ImVoteNet architecture for example, an object locator is trained on both RGB and depth data to efficiently detect the 3D bounding boxes of the objects as well as the class [9]. As batch sizes of products continue to shrink, multiple object classes are placed at the assembly station at once. The classification is therefore a crucial step during the object localization to be able to choose the right object which must be assembled next. Depending on the algorithm, the hardware is chosen based on the required interfaces, the transferability of the system and economic aspects. This is closely connected to the data input coming from the digital process chain. The latter serves as the connection between the product lifecycle and the grasping process and has to deliver the product data, process data and hardware parameters in the format processable by the algorithms.

Based on this first classification and the calculated bounding box, the pose estimation of the object follows. Most objects which are assembled in the production do not have rich texture which make the correspondence-based methods unsuitable in many cases. For weak texture and geometric detail, the template-based methods perform good, while for occlusion which is common in the production, voting-based methods are a good choice. The DenseFusion algorithm uses both RGB images and depth data for the pose estimation of objects, which are fed into the process via the digital process chain [17]. Before estimating the pose of the object, DenseFusion does an object segmentation on the RGB image to detect the pixels belonging to a specific object. After this step, both the RGB and depth data are fused accurately to predict the 6D pose of the desired object. Each pixel of the RGB image votes for a 6D pose which results in a good estimation even if parts of the object are occluded.

The last step is the selection of grasps based on the object pose. With the ML approach of DexNet, object localization and pose estimation do not have to be done but possible grasps are generated directly based on depth data [19]. The biggest drawback of this approach is the lack of object specific data for grasp generation. In the production the exact grasping location is highly relevant. The functional surfaces, the weight, center of gravity and the position where the object has to be placed in the assembly are known from the engineering phase. These factors are combined in the component requirement consisting of parts and feeding. Using these factors, grasp positions are generated. During the production, after the

6D pose of the object was detected, one grasp is selected. The selection process takes into account the pose of the object, the position of the assembly, the used robotic hardware and the environment to avoid collisions but at the same time optimize the time used to assemble the object. To make use of DexNet's good grasp selection and at the same time use object specific data, we used DexNet's grasp selection as a starting point. With this selection, we estimated the 6D pose of the object with an iterative closest point algorithm. The advantage of this approach is the good quality of the pre-selection by DexNet followed by an exact pose estimation of the object.

To train the ML-algorithms, training datasets generated either synthetically or by physical experiments are necessary. The advantage of synthetical data is the cheap generation and the possibility to include unexpected scenarios, but the different physical conditions and parameters have to be considered nonetheless. This makes the transfer of the algorithms from the simulation to reality a challenging task. The conduction of physical experiments to collect the data is more expensive and time consuming, but the data is closer to reality and can thus lead to more robust solutions [19].

5 Current Challenges

The previous section highlighted examples of how an intelligent combination of information from the product life cycle with existing ML approaches can sustainably improve the robustness and performance of gripping systems and thus find more use in assembly. However, it also becomes clear, that it is difficult to compare the existing algorithms on a common ground. This is partly because they sometimes focus on individual steps or combine several steps, and partly because they are tested with different data sets. This makes it difficult to find the optimal combination for the individual application. In order to make this possible, a test framework is required in which the models or a combination of algorithms can be tested against each other in a defined setting, as shown in Fig. 3. In such a model, the constraints of the environment are set. Since the environmental conditions and specific hardware properties can only be modeled to a limited extent, there must be a defined input stream that, in addition to the input data, also provides reference data for evaluating the result. Based on this, the individual models can then be exchanged or arranged differently until the intended requirements are met. Via defined interfaces, the algorithms can also access information from the product life cycle to improve the overall result.

Beside that, the robustness and safety of such systems must be further improved. While robustness to different lighting conditions can be achieved by training with a heterogeneous data set, the problem of reflective surfaces remains even when using stereo camera systems. Strategies must also be developed to continue operating efficiently in the event of a system failure. The system should be able to overcome such errors by having an alternative solution especially in safety critical processes and to learn from its mistakes for a continuous optimization of the solution.

Moreover, it is important to incorporate significantly more process and product knowledge into the decision-making processes of the algorithms. Therefore, there is a demand for research regarding the incorporation of domain knowledge in the training process, in transfer learning as well as on methods of data augmentation for those data types that are particularly relevant for industrial use. On the one hand, the algorithms have to offer appropriate interfaces and on the other hand, the corresponding data has to be converted into compatible and standardized formats. In general, ML approaches must be considered more in the overall tool and value chain of the process in which they are to be integrated.

Finally, the acceptance and transparency of ML solutions must also be addressed. It is important that ML based systems shift from current black box models into comprehensible systems. Explainable AI is an important keyword here, without which the broad industrial use of the algorithms is difficult to implement.

6 Conclusion and Outlook

In the context of this paper it became obvious that there are still some challenges to be solved in order to enable ML-based gripping for broad industrial use in assembly. It was shown that the requirements from production are very complex and multilayered. In particular, the parameters influence each other very strongly, so that a generalization is only possible to a limited extent or only for individual domains. On the other hand, it became clear that the described approaches offer advantages over classical, analytical approaches. For example, flexibility was derived from the assembly perspective as a central requirement, which can be achieved much more easily through ML.

However, it also became apparent that a chaining of different modules with an underlying end-to-end data process chain is absolutely necessary to achieve the higher-level objectives. For the daily use in production, the whole tool chain should be considered in a holistic approach and it should be clarified how the individual modules can be linked together in an effective way and how robustness and precision can be increased by use of underlying data. To do this, a test framework is needed to benchmark the existing models and approaches against each other in a defined environment. Therefore, it is planned to develop such a framework to enable the user to decide which approach fits to his requirement and to reveal remaining potentials for further research.

Acknowledgements The IGF-projekt 20922N (FlexARob²) of the research association FVP was supported via the AiF within the funding program “Industrielle Gemeinschaftsforschung und –entwicklung (IGF)” by the Federal Ministry of Economic Affairs and Technology (BMWi) due to a decision of the German Parliament.

References

1. International Federation of Robotics. Executive Summary World Robotics 2019 Industrial Robots
2. Martin, C., Leurent, H.: Technology and Innovation for the Future of Production: Accelerating Value Creation. In collaboration with A.T. Kearney, Geneva (2017)
3. Sahbani, A., El Khoury, S., Bidaud, P.: An overview of 3D object grasp synthesis algorithms. *Robotics and Autonomous Systems* **60**(3), 326–336 (2012). <https://doi.org/10.1016/j.robot.2011.07.016>
4. Mahler, J., Goldberg K.: Learning Deep Policies for Robot Bin Picking by Simulating Robust Grasping Sequences. In: Proceedings of the 1st Annual Conference on Robot Learning, pp. 515–524. Proceedings of Machine Learning Research (2017)
5. Kumra, S., Kanan, C.: Robotic grasp detection using deep convolutional neural networks. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 769–776. IEEE, Canada, Vancouver (2017). <https://doi.org/10.1109/IROS.2017.8202237>
6. Du, G., Wang, K., Lian, S., Zhao, K.: Vision-based Robotic Grasping From Object Localization, Object Pose Estimation to Grasp Estimation for Parallel Grippers: A Review. In: Artificial Intelligence Review, Springer, Heidelberg (2020). <https://doi.org/10.1007/s10462-020-09888-5>
7. Rusu, R. B., Blodow, N., Marton, Z. C., Beetz, M.: Close-range Scene Segmentation and Reconstruction of 3D Point Cloud Maps for Mobile Manipulation in Domestic Environments. In: 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1–6. USA, St. Louis (2009). <https://doi.org/10.1109/IROS.2009.5354683>
8. Qu, L., He, S., Zhang, J., Tian, J., Tang, Y., Yang, Q.: RGBD Salient Object Detection via Deep Fusion. *IEEE transactions on image processing* **26**(5), 2274–2285 (2017). <https://doi.org/10.1109/TIP.2017.2682981>
9. Qi, C. R., Chen, X., Litany, O., Guibas, L. J.: ImVoteNet: Boosting 3D Object Detection in Point Clouds with Image Votes. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4403–4412, Seattle, WA, USA, (2020). <https://doi.org/10.1109/CVPR42600.2020.00446>
10. Yang, Z., Sun, Y., Liu, S., Jia, J.: 3DSSD: Point-based 3D Single Stage Object Detector. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 11037–11045, Seattle, WA, USA, (2020). <https://doi.org/10.1109/CVPR42600.2020.01105>
11. Han, L., Zheng, T., Xu, L., Fang, L.: OccuSeg: Occupancy-aware 3D Instance Segmentation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2937–2946. IEEE, Seattle, WA, USA (2020). <https://doi.org/10.1109/CVPR42600.2020.00301>
12. Song, C., Song, J., Huang, Q.: Hybridpose: 6d object pose estimation under hybrid representations. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 431–440. IEEE, Seattle, WA, USA (2020). <https://doi.org/10.1109/CVPR42600.2020.00051>
13. Zeng, A., Song, S., Nießner, M., Fisher, M., Xiao, J., Funkhouser, T.: 3DMatch: Learning Local Geometric Descriptors from RGB-D Reconstructions. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 199–208. IEEE, Honolulu, HI, USA (2017). <https://doi.org/10.1109/CVPR.2017.29>
14. Tian, Z., Shen, C., Chen, H., He, T.: Robust 6d object pose estimation by learning rgb-d features. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 6218–6224, IEEE, Paris, France (2020). <https://doi.org/10.1109/ICRA40945.2020.9197555>
15. Pereira, N., Alexandre, L.A.: MaskedFusion: Mask-based 6D Object Pose Estimation. Preprint (2019). [arXiv:1911.07771](https://arxiv.org/abs/1911.07771)
16. Gonzalez, M., Kacete, A., Murienne, A., Marchand, E.: Yoloff: you only learn offsets for robust 6dof object pose estimation. Preprint (2020). [arXiv :2002.00911](https://arxiv.org/abs/2002.00911)

17. Wang, C., Xu, D., Zhu, Y., Martín-Martín, R., Lu, C., Fei-Fei, L.,Savarese, S.: Densefusion: 6d object pose estimation by iterative dense fusion. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3338–3347, IEEE, Long Beach, CA, USA (2020). <https://doi.org/10.1109/CVPR.2019.00346>
18. Bohg, J., Morales, A., Asfour, T., Kragic, D.: Data-Driven Grasp Synthesis—A Survey. IEEE Transactions on Robotics **30**(2), 289–309 (2014). <https://doi.org/10.1109/TRO.2013.2289018>
19. Mahler, J., Matl, M., Satish, V., Danielczuk, M., DeRose, B., McKinley, S., Goldberg, K.: Learning ambidextrous robot grasping policies. In: Science Robotics, Vol. 4, Issue 26 (2019). <https://doi.org/10.1126/scirobotics.aau4984>
20. Zeng, A., Song, S., Yu, K.-T., Donlon, E., Hogan, F. R., Bauzá, M., Ma, D., Taylor, O., Liu, M., Romo, E., Fazeli, N., Alet, F., Chavan-Dafle, N., Holladay, R., Morona, I., Nair, P. Q., Green, D., Taylor, I., Liu, W., Funkhouser, T., Rodriguez, A.: Robotic Pick-and-Place of Novel Objects in Clutter with Multi-Affordance Grasping and Cross-Domain Image Matching. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 3750–3757, IEEE, Brisbane, Australia (2018). <https://doi.org/10.1109/ICRA.2018.8461044>
21. Zakka, K., Zeng, A., Lee J., Song, S.: Form2Fit: Learning Shape Priors for Generalizable Assembly from Disassembly, In: IEEE International Conference on Robotics and Automation (ICRA), Paris, France, pp. 9404–9410, (2020) <https://doi.org/10.1109/ICRA40945.2020.9196733>

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.