



Configuration and Enablement of Vision Sensor Solutions Through a Combined Simulation Based Process Chain

Johann Gierecker , Daniel Schoepflin , Ole Schmedemann 
and Thorsten Schüppstuhl 

Abstract

Machine vision solutions can perform within a wide range of applications and are commonly used to verify the operation of production systems. They offer the potential to automatically record assembly states and derive information, but simultaneously require a high effort of planning, configuration and implementation. This generally leads to an iterative, expert based implementation with long process times and sets major barriers for many companies. Furthermore the implementation is task specific and needs to be repeated with every variation of product, environment or process. Therefore a novel concept of a simulation-based process chain for both—configuration and enablement—of machine vision systems is presented in this paper. It combines related work of sensor planning algorithms with new methods of training data generation and detailed task specific analysis for assembly applications.

Keywords

Synthetic data • Machine vision • Sensor planning • Object recognition

J. Gierecker (✉) · D. Schoepflin · O. Schmedemann · T. Schüppstuhl
Hamburg University of Technology, Institute for Aircraft Production Technology, Denickestraße
17, 21073 Hamburg, Germany
E-mail: johann.gierecker@tuhh.de
URL: <https://www.tuhh.de/ifpt>

© 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_26

1 Introduction

Increasing complexity and variety of production processes enhances the demand for process control systems to reduce downtime, guarantee sufficient quality and avoid rejects. Where manufacturing processes are frequently automated and therefore already have a high density of information, aircraft assembly is mainly performed manually. For this reason, the feedback of information and assembly progresses is typically done by the worker. In this environment, optical sensor systems offer the possibility to automatically record assembly states and derive the required information without intervening the actual working process. The implementation of those vision sensor solutions can be a challenging and time consuming task. Influences from different fields, such as inspection task, hardware, image processing algorithms and optics, have to be considered which either makes it an iterative trial-and-error process or requires experienced engineers. Supporting or even automating this process through an appropriate software pipeline would provide a great advantage as it could reduce configuration and commissioning time and increase the use of machine vision systems.

Such sensor planning tools are well established for mechanical inspection procedures where inspection sequences are automatically generated from CAD models. This is different for the process of designing a machine vision system, as both—the mechanical setup and the generation of a machine vision program—change task-specifically and correlate with each other. Consequently there is a need for a task-oriented description of objects and environment to ensure a successful sensor planning process.

In a novel approach we aim to re-use the 3D models and semantic task descriptions acquired during this planning phase to generate synthetic AI training data and to enable the vision system application. Creating task-specific AI training data is often necessary as industrial applications feature highly individualistic objects and environments and can hardly be generalized. Since handcrafting data is widely considered a costly and tedious approach, generating synthetic data is becoming increasingly popular. Generating data, however does not alleviate the necessity for domain experts [4, 10] and in turn causes high efforts. However, the necessary analysis and semantic descriptions have high similarity to the ones used for the sensor planning. Re-using this process chain to enable the already planned sensor set-up can benefit the use of vision systems with AI applications.

We therefore aim to introduce a combined pipeline for planning and enabling of a vision system and application. Our conceptual work contributes the following to the fields of vision sensor planning and synthetic AI training data generation in the assembly domain:

- Derivation of a assembly feature and task analysis with respect to a possible visual process inspection
- Formulation of a semantic task description and scene grammar
- Introduction of a sensor planning framework for the calculation of viewpoints based on that task description

- Introduction of a data generation pipeline based on that grammar, for generation of training data with the goal of object recognition.

2 Related Work

This section presents related work in vision sensor planning with regard to industrial applications followed by the recent developments in the generation of training data.

2.1 Vision Sensor Planning

The topic of vision sensor planning is an ongoing research for many years in fields of surface inspection [8], active robot vision [19] or public surveillance [11]. Sensor Planning in industrial application is most commonly classified by the available knowledge about the scene. This divides the publications into *Scene Reconstruction*, *Model-Based Object Recognition* and last *Scene Coverage*, which includes our use case and requires detailed knowledge about objects, positions and environment [12]. Tarabanis [14] published a survey to categorize work in the field of sensor planning for *Scene Coverage Problems* in which even current research can still be classified.

Generate-and-test approaches [9] generate sensor configurations by equally dividing the solution space and evaluating the single configurations based on the task requirements. Cowan [3] shows with a *synthesis* approach that a configuration can be generated by an analytic description of inspection task, sensor parameters and several feature detectability constraints such as visibility, concealment, perspective, field-of-view, resolution or depth-of-field. *Expert systems* [2] describe databases which contain information about successful implemented viewing and illumination system, expressed in several rules to support the user while planning his configuration. The last category is *Sensor Simulation* where the scene is visualized within a framework to render sensor-real data based on configuration generated with either of the presented methods. All these different approaches try to find a set of viewpoints from which a maximized set of feature points can be detected. These feature points highly differ between every task. Where for a use case like object reconstruction the set can be a discrete description of the entire surface [9], the assembly inspection requires a task-specific analysis of the relevant features as it is not purposed to have a visibility of the complete object.

However most works regarding sensor planning assume, that the modeled sensor poses will exactly be executed. Manual influences or deviations in positioning systems often result in pose errors. Scott [18] introduces pose errors in the sensor planning process and suggests methods to minimize these. As we manually transfer the calculated sensor poses into our test setup, the pose error problem is relevant for our use case.

2.2 Training Data Generation

Following the planning phase, the vision application has to be implemented. As the use of AI based solution increase in this field, the need for appropriate training data has risen alike. Due to the time-consuming and expensive nature of manual data acquisition processes, use of synthetic data that is rendered out of 3D models has gained in popularity in recent years [5]. Successful training with synthetic training data was achieved in the fields of autonomous driving [7], picking [16] or identification of household objects [5]. Other industrial applications [15] have used CAD data to utilize Reinforcement Learning of a robotic grasping trajectory. Similar tasks were solved by [1, 6]. The insertion of pegs through object identification trained on synthetic data was shown by [17]. Synthetic Data enabled object recognition of assembly related objects like screws and the like was shown by [20]. These datasets are publicly available, yet not necessarily transferable to every industrial vision system and task, due to unknown environments and objects. Therefore, when designing a vision based application, often new data generation pipelines have to be created alongside.

The creation of such data generation pipelines is in need of defining what is to be displayed, a proper scene grammar [10], and implementation in a toolbox. Later can be provided by data synthesizing tools like NDDS¹ for Unreal Engine, SynthDet² for Unity and similar for Blender,³ the semantic definition of a grammar however has to be done by the user for each problem variation he wants to train. Supporting the user in this process is not sufficiently addressed in recent approaches, but could be an important element in widespread use of AI based vision systems.

Where the presented work in sensor planning lacks a task individual feature definition of what has to be visible, the process of training data generation requires similar information about the object and environment. Consequently we display a simulation based process chain with a combined task semantic for both fields. Furthermore the introduced problem of pose errors can be handled with the use of an AI solution where the calculated pose is part of the variation parameters for training data generation.

3 System Overview

The aim of our work is to combine the presented fields of sensor planning and training data generation and to complete the process chain with a task analysis for the field of assembly inspection. Figure 1 shows the proposed system pipeline which individual steps will be explained in the following sections.

¹https://github.com/NVIDIA/Dataset_Synthesizer, September 2020.

²<https://github.com/Unity-Technologies/SynthDet>, September 2020.

³<https://github.com/921kiyo/3d-dl>, September 2020.

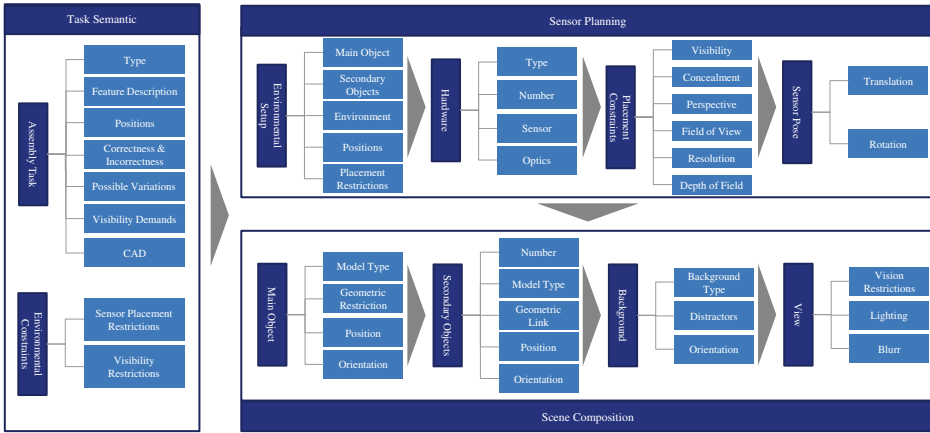


Fig. 1 System overview

3.1 Assembly Feature and Task Analysis

The inspection task of assembly verification is in this work proposed as a problem in object detection, where it is assumed that algorithms need to make decision based on a 2D-image of assembly joints. To verify a robust detection as a result of a automated system configuration, the task has to be analyzed before sensor planning process starts. The resulting task semantic (see Fig. 1) describes all information, which is required by the sensor planning and scene composition. Assembly tasks can differ from positioning of single or multiple objects over connecting them via screws or rivets to welding or soldering. Furthermore each category itself differs in its specifications depending on the geometry. Therefore we describe the parameters, which characterize a successful assembly task and convert them in geometrical features. Figure 2 shows exemplarily how the relevant features (marked red) differ in the category of a bolt connection. To detect a hexagon socket screw, features inside of the head have to be visible from the sensor view, whereas a hexagon cap has its characterizing features on the outside. Depending on the connection type there are additional parts (e.g. washer or nut) on the underside, which may not be visible from the same viewpoint as the features on the topside.

Beside those visibility demands there may also be visibility restrictions inside the environment. These are areas or objects (e.g. humans) which must not be visible within the sensor data. Sensor restrictions are positions where a mounting of sensors is restricted due to interference of the assembly process. Together with CAD-data the scene can be modeled within the sensor planning and scene composition processes.

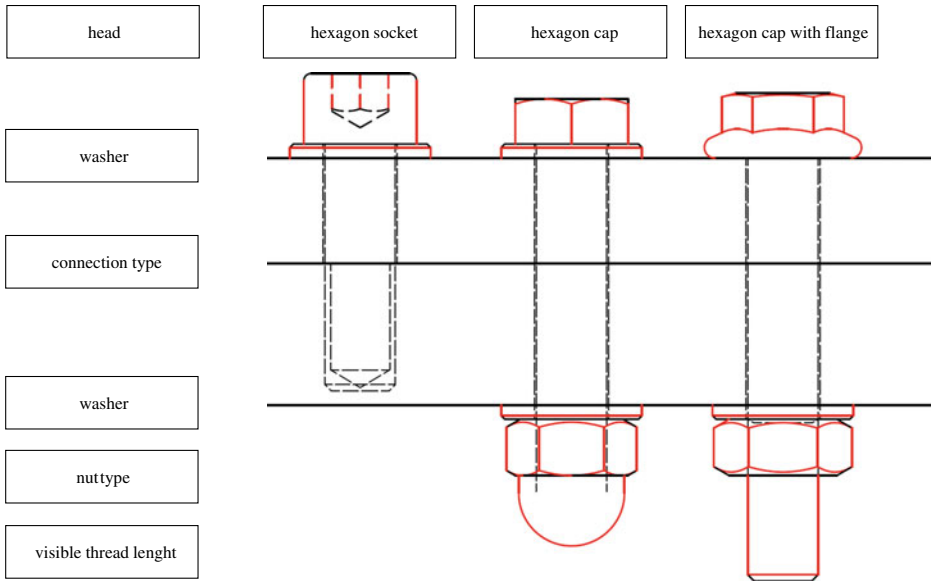


Fig. 2 Feature extraction by the example of a bolt connection

3.2 Calculation of Possible Viewpoints

Calculation of viewpoints and final pose selection as part of the sensor planning process uses mathematical descriptions of feature detectability constraints (see Sect. 2.1) together with sensor parameters and 3D-scene to generate possible poses which satisfy those constraints. Where relevant features for visibility demands are formulated within the task semantic, other optional features can be extracted from the STL-CAD-file. The description of surfaces as triangles allows us to simply generate cell center points and the relating surface normals all over the object. The relationship between the optical axis of the camera and those center points or normals is the base for most features (e.g. visibility, concealment and perspective) within the calculation process.

Where the visibility demands must be fulfilled, the results of the calculations with optional features are important for the rating of viewpoint candidates. The amount of features that satisfy the feature detectability constraints allow us to quantify the *degree of visibility* of the assembly joint. Start poses for sensor planning result from the positioning restrictions of the task semantic. After the calculation of every combination of sensor pose and feature, a final pose is selected based on the amount of features, which satisfy all constraints viewed from this pose. To avoid the expected pose error problems, this final viewpoint has to be slightly varied in the following process of training data generation.

3.3 Scene Composition for Training Data Generation

The rules and formulations describing the possible compositions of a scene can be referred to as the grammar of that scene. In order to implement a pipeline that creates scene variations according to such a grammar a parametrization has to be defined. We will first focus on the more generalized scene grammar and derive the parameter space in the later presented Use-Case.

We define the goal of the training data to enable an assembly process supervision application. Through object recognition this application should state whether the object in the sensor view is assembled correctly or not. However, this is not viable for every assembly type. We discard cases for which measurements e.g. for a slit have to be taken, to determine whether the assembly was done successfully. Our focus is the presence of certain objects e.g. screws or larger components in a view. Through the aforementioned sensor planing, appropriate view poses for detection of the objects within the view are defined.

We distinguish between the main object which is assembled and the secondary objects which are assembled to the main object, which is to be detected. The necessary parameters to define are shown in Fig. 1. The to be rendered scene is built around the main object, whereas the position and the orientation may change with respect to the sensors field of view. According to the type of assembly done, the secondary objects are placed.

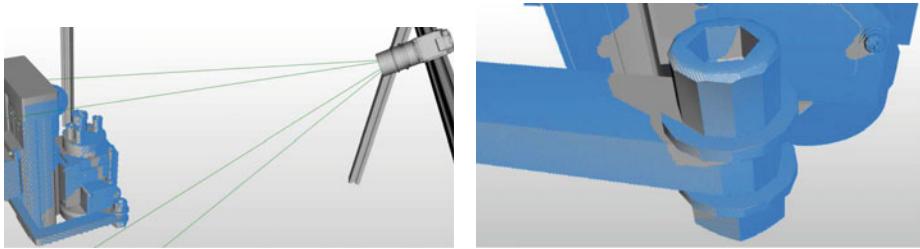
4 Use Case

For reasons of confidentiality the suitable use cases from the project can not be used for the presentation. Therefore we select a tool for scarfing of CFRP structures in aircraft MRO [13] as our main object for the validation of the presented process chain. The relevant joint is a bolt connection, which consists of a hexagon socket srew (M12x35mm) with two washers and a nut. These parts represent the secondary objects. A sensor frame is placed around the object where the camera can be flexibly mounted. The available machine vision hardware is a IDS uEye camera with a 5MP sensor⁴ and a Schneider-Kreuznach lens with a focal length of 12 mm.⁵

Correct assembly is considered when all objects and the correct type of srew is mounted. A correct srew can be expressed by the geometry of its head and its threat length which is visible underneath the nut. This results in visibility demands for the inner lines of the hexagon socket and the thread, which can be described as linear and cylindrical features. The smallest feature of interest is the thickness of the washer, which is has to be considered for the calculation of the resolution constraint.

⁴<https://de.ids-imaging.com/store/ui-5280cp-rev-2.html>, September 2020.

⁵<https://schneiderkreuznach.com/en/industrial-optics/lenses-2-3-c-mount-3-5-mp>, September 2020.



(a) View frustum (green lines) from the final viewpoint (b) Field-of-view (blue areas) and visibility mapped on the objects

Fig. 3 Visualization of the final viewpoint

Table 1 Parametrization and variation of the synthetic training data generation

Object	Parameter	Variation
Main object	Position	10% Translation variation
	Orientation	Rotation z: 10°
	Colour/texture	As CAD model
Secondary object	Screw length	30–60 mm
	Position	Mounted, Thread showing
	Screw nut	Off–on
	2 x washer	Off–on
Background		13 variations
Underground		9 variations
Light	Type	Sun, spot, point, plane
	Intensity	2000–20000

The sensor frame is selected as a positioning restriction and 248 viewpoint candidates are generated equally over the frame. All STL-CAD-files of the four secondary objects add up to 3242 cell normals. Some are directed to the inside, which reduces the amount of cell normals for the calculation to 2590. Figure 3 shows the sensor simulation framework with a visualization of the view frustum from the final viewpoint (Fig. 3a) which supports the user to verify the calculated pose. The field-of-view is mapped on the objects which shows that the features to detect the correct type of screw are visible (Fig. 3b).

We utilize the semantic grammar and view-point definitions to derive a parametrization space for the synthetic training data variations which in turn is based on the parameters of the vision system. Our main object is the scarfing tool, mounted target objects are screw, washers and nut. The quantified values of the parameters as ruled by the grammar are shown in Table 1. This parameter space is then implemented in Blender. We rendered 1600 combinations for correct assembly and 3500 for incorrect assembly. Examples are shown in Fig. 4a, b.

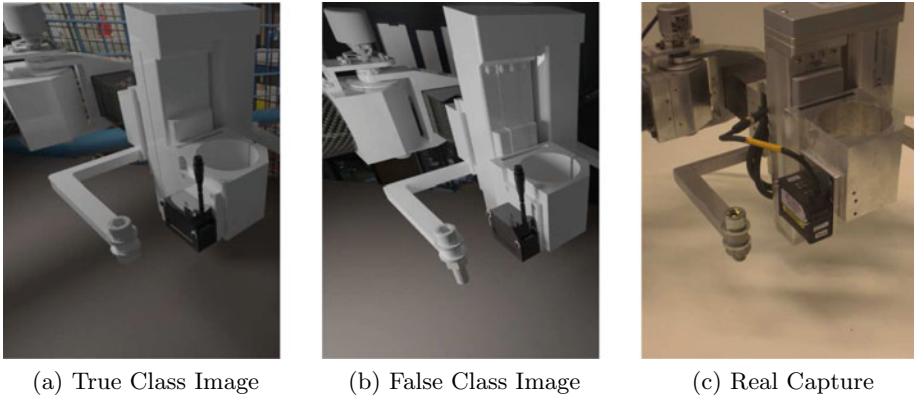


Fig. 4 Examples of rendered training data compared with an real image. One Image represents the True Class with correct assembly, whereas the other image image displays an incorrect screw

4.1 Presentation

We trained a VGG-16 network with our dataset. A binary classification task was trained, with an added top layer of 1024 Dense units, 20% Dropout and a single unit output layer. This was trained for 10 epochs with adam. Afterwards the entire network was trained for 5 epochs with an SGD optimizer and a learning rate of 0.0001. To compensate for the lack of applied Domain Adaption techniques, we fine tune to the real domain with 15 real images of each class. Afterwards the network was tested against real world data, picturing the real scene in various lighting situations and with multiple distractors applied. In sum 164 images were gathered for testing. The confusion matrix of the results can be seen in Table 2. The classification accuracy results to 97%.

4.2 Discussion

With 97% classification accuracy, the aim of enabling a Deep-Learning task can be considered achieved. However, it is to be noted, that the task of object detection and localization

Table 2 Confusion matrix of test results. True class, corresponding to the correctly mounted assemblies and false class indicating the incorrect assemblies

	True _{pred}	False _{pred}
True _{real}	78	1
False _{real}	4	81

are only moderately challenging in this set-up and classification accuracy cannot be the lone indicator towards success of the synthetic training data generation. Our aim was to demonstrate how some of the information and parametrizations that are generated by the process of designing a vision system for a task can be reused to develop a data generation pipeline. This in turn can lead to a quick enablement of a image processing task. No additional analysis had to be done to identify suitable variations of the scene to obtain 3D models or to define the labels of a scene composition. Although the results of the demonstration are promising, additional testing with more complex environments can provide deeper insight into the applicability of the presented method. Suitable domain adaption techniques are to be applied, to improve the networks transfer capability towards the real application data. Further applications, e.g. measurement tasks can be developed. For this adaptations for the synthetic data generation pipeline have to be implemented, to include labeling of appropriate key-points.

5 Conclusion and Future Work

In this work an approach of a combined simulation based process chain for both—configuration and enablement—of a machine vision is presented. It states, that sensor planning and generation of a processing pipeline contribute from a common detailed task and object analysis. The resulting task semantic includes basic geometrical description of the object and environment as well as relevant features for the certain assembly process and environmental constraints which affect the sensor placement. This database is relevant for the sensor planning process and can simultaneously be used for the scene composition as part of AI training data generation. Using a sensor simulation and the example of a bolt connection we show that it is possible to set up a working machine vision solution for assembly verification from CAD data only. To improve the presented process chain and to extend the applicability of this concept, future work includes:

1. **A joint framework** for sensor planning and training data generation would use synergies between both fields and allow an evaluation of viewpoint candidates based on realistic sensor data renderings.
2. **Further visibility constraints**, such as illumination and overexposure, have to be integrated into the framework in order to increase the realism and thus the quality of calculations.

Acknowledgements Research was funded by the Luftfahrtforschungsprogramm LuFo V-3 “Hi-Digit Pro 4.0” and “DEPOT.” The authors wish to thank Sönke Bahr and Sebastian Sauppe (*3D.aero GmbH*) for cooperation during the work on the simulation framework.

Gefördert durch:

aufgrund eines Beschlusses
des Deutschen Bundestages

References

1. Bousmalis, K., et al.: Using Simulation and Domain Adaptation to Improve Efficiency of Deep Robotic Grasping. In: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 4243–4250 (2018)
2. Burla, A., et al. An assistance system for the selection of sensors in multiscale measurement systems. In: Furlong, C., Gorecki, C., Novak, E.L. (eds.) SPIE Optical Engineering + Applications. SPIE, 77910I (2010)
3. Cowan, C., Kovese, P.: Automatic sensor placement from vision task requirements. IEEE Trans Pattern Anal Machine Intelligenz **10**, 407–416 (1988)
4. Dahmen, T., et al.: Digital reality: a model-based approach to supervised learning from synthetic data. In: AI Perspectives 1, Springer, Heidelberg, pp. 1–12 (2019)
5. Hinterstoisser, S., et al.: An Annotation Saved is an Annotation Earned: Using Fully Synthetic Training for Object Detection. In: 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW) (2019)
6. Fang, K.: Multi-task domain adaptation for deep learning of instance grasping from simulation. In: 2018 IEEE International Conference on Robotics and Automation (ICRA,) pp. 3516–3523 (2018)
7. Gaidon, Adrien: VirtualWorlds as Proxy for Multi-object Tracking Analysis. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4340–4349, (2016)
8. Gospodnetic, P., et al.: Flexible Surface Inspection Planning Pipeline. In: 6th International Conference on Control, Automation and Robotics, pp. 644–652 (2020)
9. Jing, W., et al.: (2016) Sampling-based view planning for 3D visual coverage task with Unmanned Aerial Vehicle. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1805–1815 (2016)
10. Kar, A., et al.: Meta-sim: Learning to generate synthetic datasets. Proceedings of the IEEE International **2019**, 4550–4559 (2019)
11. Liu, J., Sridharan, S., Fookes, C.: Recent Advances in Camera Planning for Large Area Surveillance. ACM Comput Surv **49**, 1–37 (2016)
12. Mittal, A., Davis, L.: A General Method for Sensor Planning in Multi-Sensor Systems: Extension to Random Occlusion. Int J Comput Vis **76**, 31–52 (2008)
13. Rodeck, R., Schüppstuhl, T.: Repair of composite structures with a novel human-machine system. In: Proceedings of ISR 2016: 47st International Symposium on Robotics (ISR), pp. 660–666 (2016)
14. Tarabanis, K., Allen, P., Tsai, R.: A survey of sensor planning in computer vision. IEEE Trans Robot Automation **11**, 86–104 (1995)
15. Thomas, G, et al.: Learning Robotic Assembly from CAD. In: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 3524–3531 (2018)
16. Tobin, J., et al.: Domain randomization for transferring deep neural networks from simulation to the real world. In: 17 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, pp. 23–30 (2017)

17. Triyonoputro, J.: Quickly Inserting Pegs into Uncertain Holes using Multi-view Images and Deep Network Trained on Synthetic Data. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), p. 5729–5779 (2019)
18. Scott, W.: Model-based view planning. *Machine Vision and Applications* **20**, 47–69 (2009)
19. Zeng, R., et al.: View planning in robot active vision: A survey of systems, algorithms, and applications. *Comp Visual Media* **6**, 225–245 (2020)
20. Židek, K., et al.: An Automated Training of Deep Learning Networks by 3D Virtual Models for Object Recognition. In: *Symmetry* **11**, (2019)

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.

