



Opportunities of 3D Machine Learning for Manufacturability Analysis and Component Recognition in the Additive Manufacturing Process Chain

Tobias Nickchen¹(✉), Gregor Engels¹, and Johannes Lohn²

¹ Paderborn University, 33098 Paderborn, Germany
t.nickchen@sicp.uni-paderborn.de

² Protiq GmbH, 32825 Blomberg, Germany
jlohn@protiq.com,
<https://www.uni-paderborn.de/en/home/>,
<https://www.protiq.com/en/>

Abstract. Additive Manufacturing (AM) is one of the manufacturing processes with the highest potentials in the current transformation of the industry. To make use of this potential and to achieve consistent product quality at decreasing costs not only the 3D printers themselves but also the whole process chain has to be automated. Due to the high degree of digitalization and the use of 3D Computer Aided Design (CAD) models within the entire process chain, it is possible to use these information for automation via intelligent data analysis. In this paper, the potential of using 3D Machine Learning (ML) approaches for automation and optimization of sub-processes of the process chain is analyzed. Therefore, we consider the information flow of the 3D models in the process chain of an AM service provider. The potential of using state-of-the-art algorithms from the field of 3D ML for automation of sub-processes like manufacturability analysis, production cost calculation or 3D-component recognition is analyzed and feasibility is examined. For the sub-processes of manufacturability analysis and 3D-component recognition prototype solutions have been implemented and evaluated. For the production cost calculation, only preliminary analyses were carried out, on the basis of which the possible applications of 3D ML algorithms can be estimated. With our analyses, we demonstrate that it is possible to further automate the process chain of AM service providers through the use of 3D ML algorithms.

Keywords: Additive Manufacturing · 3D Machine Learning · Neural Networks · Process chain automation

1 Introduction

Additive Manufacturing (AM) offers enormous potentials for the use of optimized components in many highly technical industries. However, since it is still a relatively young process on an industrial scale, many steps of the process chain that go beyond actual production are characterized by manual work. Since the beginning of the fourth technological revolution, industrial processes have been iteratively optimized. The use of interconnected sensors and cyber-physical-systems leads to intelligent self-adaptive production chains. This enables a more efficient production and a reduction of costs while achieving higher quality [1, 2]. Currently, conventional manufacturing processes cover the majority of industrial production. Only 0.04% of global goods production is additively manufactured [3]. According to the same study, however, 5% is quite within the realm of what is possible in the future, if the AM industry takes advantage of its development opportunities. In order to keep pace with common manufacturing processes such as casting, forging or milling and to enable series production, production costs must be further reduced, consistent quality ensured and the advantage of short time to market further expanded. To achieve this, the potential of process chain automation must be further exploited.

For the analysis of the process chain we worked together with an AM service provider. As the service provider works with 3D printers from the Powder Bed Fusion (PBF) field [4], we also focus on the process chain of this production method. PBF processes for processing polymers and metals are widely used in the AM industry. The actual AM process of the PBF 3D printers themselves is already completely automated.

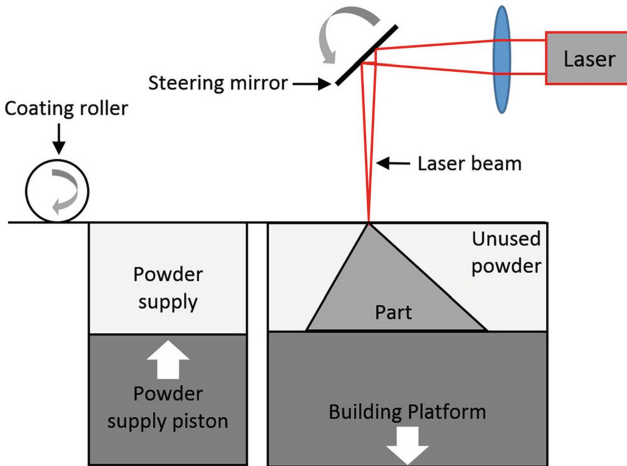


Fig. 1. Structure of PBF 3D printers [5].

Nevertheless, large parts of the upstream and downstream processes are characterized by manual work steps. In the context of our work, we focus on the process steps which are directly linked to the information flow of the 3D models and identify the potentials of using 3D Machine Learning (ML) algorithms for process automation. 3D ML is one of the evolving fields of ML and offers enormous potentials for the analysis of 3D-data.

Our main contribution is the proof of concept that parts of the AM process chain can be automated by using 3D ML techniques. For the steps manufacturability analysis and 3D-component recognition, we already implemented evaluation systems and proved the feasibility of these systems. For the process step of production cost calculation, only preliminary analyses were carried out which underline the potentials for this sub-process.

In Sect. 2, we describe the major parts of the process chain and analyze which sub-processes are suitable for further automation. Afterwards we give an overview of state-of-the-art 3D ML algorithms which can be used for our purposes. In the following Sect. 4, the results of our evaluation studies are presented, followed by a conclusion and the future perspectives.

2 AM process Chain Analysis

In this chapter, we take a deeper look on the AM process chain with its upstream and downstream processes. For getting a detailed insight of all processes, we worked together with an industrial AM service provider and focused on the PBF process chain. All analyses in this paper refer to these processes. We first describe the basic sub-steps of the process chain in Sect. 2.1 and subsequently analyze the automation potentials of the different steps in Sect. 2.2.

The basic structure of PBF 3D printers can be seen in Fig. 1. The main characteristic of PBF is the layer-wise application of powder and subsequent selective melting of the powder using a laser with up to 200 W for polymers and around 400 W for metals [6]. This specific approach leads to the necessity of various additional steps, such as the manufacturability analysis, production cost calculation or 3D-component recognition.

Since we deal with the analysis of the information flow in the process chain, we only cover sub-processes that are directly related to the digital 3D models. Of course, the process chain includes further steps besides those shown in Fig. 2. However, these have been deliberately removed in our graphic, as they have no direct relation to the digital information flow. Subsequently, we work out sub-processes with automation potentials in Sect. 2.2.

2.1 The Process Chain of an AM Service Provider

The process chain of an AM service provider includes many up- and downstream processes besides the actual production. We deal with all steps that take place in the environment of the service provider and therefore start with the data upload. The construction of the Computer Aided Design (CAD) models is on the side

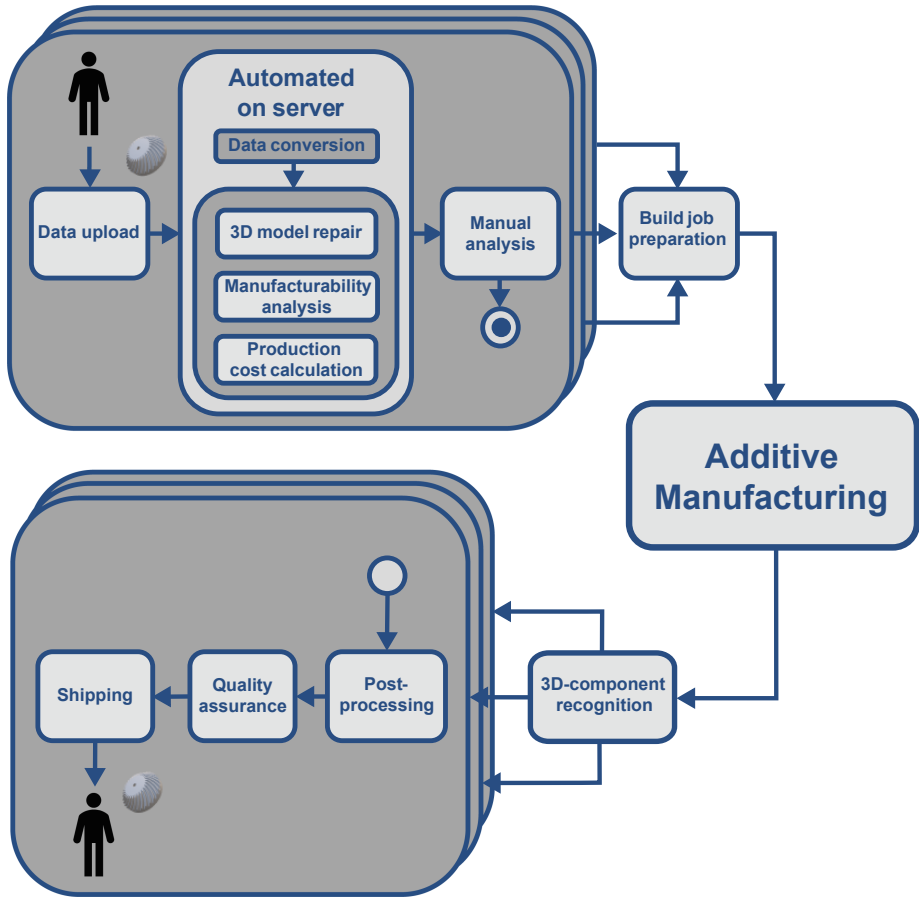


Fig. 2. Process steps in the AM process chain which are directly linked to 3D models.

of the customer and is therefore not included. An overview of the process steps related to the 3D models is given in Fig. 2.

Data Upload. As mentioned before, the process chain begins with an upload of 3D models by a customer. The data can be uploaded in nearly all common 3D data types. Regardless of the uploaded data type, the 3D models are automatically converted to the Standard Tessellation Language (STL) [7] file format. The STL file format is the standard data format used for AM. All following steps are based on the STL models.

Model Repair. The three subsequent steps 3D model repair, manufacturability analysis and production cost calculation are already performed automatically on a server. It is necessary to perform these steps automatically in near real time

so that potential customers can receive a quote for their uploaded 3D models without delay. 3D model repair involves the checking and correction of errors that may occur during the CAD construction or the conversion to STL. For that task, software with ready-to-use functions can be used to repair STL models completely automatically [8].

Manufacturability Analysis. The manufacturability analysis is necessary to verify if an STL model with a given geometry can be manufactured with a specific manufacturing technology. The manufacturability of an object is defined by geometric features like overhangs, bores or channels and process parameters like layer thickness or build orientation [9].

Only parts of these features can currently be checked automatically with software tools. Problematic here is above all that the existing definitions of manufacturability can only be applied to existing models to a limited extent. Possible guidelines for the manufacturability of 3D models were considered by various researchers in the last years [10, 11]. Most of these design guidelines are based on standard geometries like cylinders or cuboids. To use these guidelines, 3D models must be approximated by combinations of the standard geometries. The guidelines can then be applied to the approximated 3D models. A major problem is that complex 3D models can often only be approximated very imprecisely by standard geometries. Therefore, only the guideline of minimum wall thickness is currently automatically tested by a software tool since this feature is not based on standard geometries. At the AM service provider, a more detailed check of the 3D models is therefore performed manually by AM experts in the following step.

Production Cost Calculation. The production cost calculation is the last step which has to be performed automatically after the upload to enable real-time pricing. Based on the 3D models geometry and operational parameters, the production cost of 3D models has to be calculated. The costs of a component are influenced by all processes in the process chain. From the preliminary analyses to the manual build job preparation, the actual 3D printing process and the various post-processing steps right through to shipping. Due to the great variety of individual geometries, however, it is hard to exactly predict the costs for some of these steps, e.g. the time necessary for the post-processing steps like sandblasting or support removal. Owing to the complexity, the relationship between the geometry of the 3D model and the amount of work required for the post-processing steps cannot easily be described mathematically. Therefore, the costs of these procedures can only be approximated with current software solutions.

Build Job Preparation. The build job preparation is split into manual work steps carried out by AM engineers work steps executed by software tools. Single CAD models must be combined to build jobs. Depending on the requirements of

the final product, e.g. different build orientations of a 3D model in the construction space are necessary. Automated functions are in principle available for calculation of a good orientation and position of the 3D models in the build chamber [8, 12]. Nevertheless, especially the orientation has a strong influence on component properties such as surface quality or mechanical characteristics. Therefore, objects produced in automatically calculated orientations do not always meet the specifications a customer expects. On the basis of interviews with the experts of the AM service provider, it has become clear that especially the orientation of the 3D models is currently still chosen manually by AM engineers in order to guarantee the optimal quality of the components to be produced.

Additive Manufacturing. After the build job preparation, the physical process steps begin. As mentioned in Sect. 1, the AM process itself is already completely automated. Depending on process and material, production is followed by various technical finishing steps such as powder removal or sandblasting. These processes run apart from the digital information flow. The physical production and the virtual information flow converge again in the 3D-component recognition step.

3D-Component Recognition. In PBF processes, different components are manufactured together in one batch. Especially with the PBF 3D printers for polymer processing, up to 100 different components are often produced in one build job. After production, they must be assigned to the appropriate digital 3D model again, in order to be able to continue with the subsequent post-processing steps. This is still a manual process which is time consuming and expensive.

Post Processing. After the individual components have been separated again, all digital information is available in order to be able to carry out post-processing steps such as surface treatment and to subsequently check whether all customer criteria have been met in the quality control process step. If this is the case, the process chain can be completed with the dispatch of the components to the respective customer.

2.2 Automation Potentials in the AM Process Chain

In this section, we analyze the optimization potential of the process steps described in Sect. 2.1 to decide which process steps are suitable for automation using 3D ML algorithms. The process steps manufacturability analysis, build job preparation, 3D-component recognition and post-processing are potentially of interest for optimization because they are characterized by manual work. Furthermore, the step production cost calculation can be considered. Although this step is automated, it still offers further potential for optimization to improve the current approximation of the calculation of real production costs. In our work, we do not consider build job preparation any further because we believe that the

expertise required in this step is difficult to replace with intelligent algorithms. Additionally, the step post-processing will neither be considered in the context of this paper, since intelligent data analysis is not relevant for that process.

Manufacturability Analysis. In Sect. 2.1, we explained that current manufacturability criteria are represented by design guidelines which are based on standard geometries. With increasing complexity of components, the guidelines are no longer sufficient to represent the real limitations of processability. Since it is difficult to describe the restrictions by concrete mathematical representations, we believe it is more goal-oriented to investigate the possibility of generating conclusions about the manufacturability directly from existing production data. Intelligent systems are capable of independently learning the features that are decisive for manufacturability. Therefore, we see the potential of automation with 3D ML algorithms for this process step.

According to AM experts of the AM service provider, errors occurring in the production can be triggered by many different causes. Partly it is filigree details in the 3D models, partly larger connected features that lead to faulty production. Therefore, a solution for this application has to be able to capture and process both fine details and global correlations within the 3D models.

Production Cost Calculation. As described in Sect. 2.1, it is difficult to define the mathematical relationships between the geometry of a 3D model and the actual costs for 3D printing itself and the pre- and post-processing steps. The cost of a component depends on many different factors, such as the objects volume and geometric complexity or the orientation in the build chamber. These factors in turn influence direct cost factors such as construction time, material consumption or the amount of work required for the various finishing steps such as sand blasting and surface preparation or in metal processing the removal of support structures. Our research and discussions with the AM service provider have shown that in particular sand blasting, surface preparation and support removal in metal processing are a major cost factor as they are mostly performed manually. Due to the great geometric individuality of the 3D models, it is difficult to calculate the exact amount of work required for the manual post-processing steps. The amount of work and thus the costs are strongly dependent on the geometric complexity. Complex geometries can contain many different features like free-form surfaces, small openings or complex internal structures. It is practically impossible to create a mathematical formula with hand-generated features to calculate the correlation between these features and the costs for post-processing.

At this point we see clear opportunities for optimization. Data-driven systems are able to learn to extract the relevant features. The big advantage of Deep Learning (DL) models is that they are able to generate so called Deep Features from raw inputs on their own. This enables them to learn the complex mathematical relationships that exist here on the basis of their own automatically generated features. By collecting data such as the manual processing time

for post-processing of produced objects during real production, a 3D ML system can then learn the relationship between the geometry and the processing time and thus the processing costs.

By determining the costs for individual post-processing steps more precisely, the total costs for an object can be significantly improved compared to the estimation currently in use.

3D-Component Recognition. For manually assigning physical components to digital 3D models, an employee visually compares the properties of the objects with the 3D models. Due to the increasing number of produced objects, manual sorting is today already connected with an extremely high effort. Therefore, we see a high potential for cost savings through an automated solution. Since in AM completely new components have to be detected in real time every day, conventional computer vision approaches reach their limits. In contrast, 3D ML applications can adapt to the daily changing components. During our work on this topic a first commercial approach recently emerged for this task (AM Flow [13]). According to their data, they achieve a detection rate of 95%. To the best of our knowledge, their system has never been evaluated on a standardized, publicly available data set. In our opinion it is necessary to achieve a detection rate of almost 100% for complete automation. Therefore, a further optimization with 3D ML is possible for the step of component recognition.

3 3D Machine Learning

For the process steps of manufacturability analysis, production cost calculation and 3D-component recognition explained in Sect. 2.2, we have examined the possibility of using 3D ML approaches to automate that steps. To find the most suitable solution for our task, we compared several state-of-the-art 3D ML approaches.

Existing 3D ML algorithms for popular tasks like 3D object recognition, segmentation or tracking are using either directly 3D data or 2D projections like images. Both variants contain different positive and negative aspects. By sensors like 3D scanners, 3D data in the form of point clouds, meshes or Red, Green, Blue, Depth (RGB-D) data can be generated. 3D representations offer the advantage of displaying the scanned data in great detail but have the disadvantage of being very memory intensive and therefore place high demands on the processing hard- and software. On the other hand, 2D projections in the form of images offer the advantage of being less memory intensive but have the disadvantage of information loss.

In order to find the best solution, we figured out the positive and negative aspects of the different approaches and decided which approach is most useful for our purposes.

3.1 Image-Based

Image-based approaches like RotationNet [14] or Group-view Convolutional Neural Network (GVCNN) [15] have the advantage of less memory consumption, faster training times and lower hardware requirements compared to approaches using 3D data. For frequently used benchmark tasks such as the classification of the Princeton ModelNet40 dataset [16], they achieve classification rates of up to 97% which is state of the art and exceeds 3D-data-based approaches.

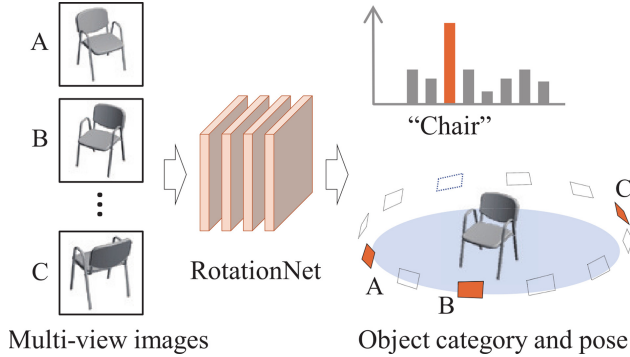


Fig. 3. Data representation of RotationNet [14]

In most state-of-the-art image-based approaches the basic drawback of losing information about the 3D objects by using 2D projections is counteracted by using multi-view representations of the 3D models. For each 3D object of the data set, multiple images are generated by using different view points around the 3D model (see Fig. 3). These groups of images are then used as collective input data for the model, giving more information to the model what leads to the state-of-the-art results in classification of ModelNet40 [14, 15]. However, differentiating 3D models from ModelNet40 differs from our issue because filigree details are relatively irrelevant. Therefore, it must be proven that the image-based approaches are also capable of differentiating very similar objects.

3.2 3D-Data-Based

Approaches based on 3D data can be assigned to the two sub-classes point-cloud-based and voxel-based. Voxels are the three-dimensional equivalent of pixels in two-dimensional space.

Point-Cloud-Based. Point-cloud-based approaches like PointNet++, Relation-shape Convolutional Neural Network (CNN) [17, 18] or Linked Dynamic Graph CNN [19] directly work on 3D point clouds which can be generated by different types of sensors (Fig. 4).

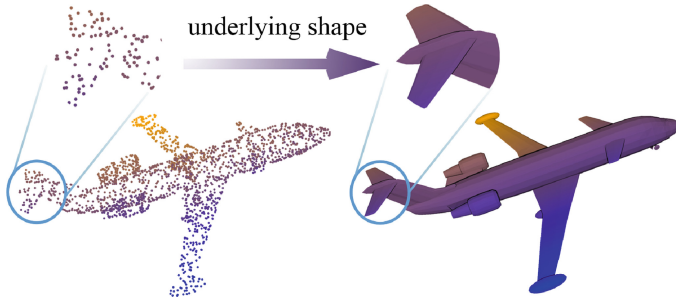


Fig. 4. Data representation of relation-shape CNN [18]

Since point clouds are unstructured data sets with possibly varying point densities in different parts of the cloud, convolutional filters which are often used for image processing can not be used. For being capable of handling the varying resolutions, point-cloud-based approaches learn features in multiple hierarchical scales. However, the possibility to capture filigree and global features at the same time can only be realized in theory because it is limited by the size of Graphics Processing Units (GPUs). In practice, a maximum of 5000 points is usually used which is not sufficient to display all details of complex objects. For the analysis of fine details of 3D models, point-cloud-based approaches reach their limits.

Voxel-Based. Similar to point clouds, so called voxels capture three dimensional information. The difference to point clouds is that the data is stored structured in a three-dimensional grid. Therefore, common mathematical operators like convolution used in Convolutional Neural Networks (CNNs) can be applied on the 3D data. This enables the same behaviour of voxel-based systems in the three-dimensional as the conventional CNNs in the two-dimensional space. The main drawback similar to point-cloud-based approaches is the high usage of computational memory with growing resolution of the voxel grid. Developers of the approaches VoxNet, OctNet [20, 21] or Spatial-hashing-based Convolutional Neural Network (HCNN) [22] have thus tried to optimize the data structure to enable a high resolution. Wang et al. [22] represent the state of the art and reach a resolution of 512^3 voxel using high end GPUs. Therefore, voxel-based approaches are best suited to capture filigree details in 3D models (Fig. 5).

3.3 Approach Selection

Based on the requirements of the process steps manufacturability analysis, production cost calculation and 3D-component recognition and the characteristics of the different types of approaches, the following potentials for process automation arise. Since both local and global features must be considered in the sub-processes of manufacturability analysis and production cost calculation, the use

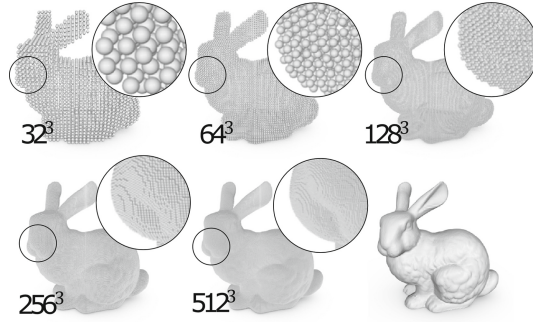


Fig. 5. Data representation of HCNN [22]

of a voxel-based approach is reasonable for these sub-processes. The longer training time is negligible, since no recurring training process is necessary. For 3D-component recognition all described types of algorithms offer positive arguments which argue for a use. Therefore, we evaluate all described algorithms for that task.

4 Evaluation

In order to prove that the described algorithms are suitable for solving the tasks of manufacturability analysis, production cost calculation and 3D-component recognition, an evaluation must be carried out. To the best of our knowledge there are currently no benchmark data sets available for the problems we are investigating. Therefore, we generated our own data sets based on the Thingi10K data set [23] which contains 10000 3D models from the AM domain and trained the algorithms with subsets of that data. The subsets were adapted to the respective problem definition and are oriented towards real production data.

4.1 Manufacturability Analysis

To verify the basic usability of the HCNN approach for manufacturability analysis, we had to generate a labeled data set. Since manual labeling of thousands of 3D models is extremely time-consuming, we decided to use the wall thickness tool which is currently used at the AM service provider. With the help of the tool and the Thingi10K data set, a labeled data set with about 5000 3D models has been created where the 3D models are divided into the categories “manufacturable” and “non-manufacturable” with respect to the feature “minimum wall thickness”. Labeling this data set, enables us to verify the ability to recognize filigree details in 3D models. Some example 3D models can be seen in Fig. 6.

This data set has been used to train the HCNN model which is presented in Sect. 3.2 with a resolution of 512^3 voxels. The HCNN model is able to achieve an accuracy of 94% correctly classified 3D models in first studies. By applying



Fig. 6. Example 3D models with colorization of thin areas. Red areas are beneath a threshold value of 0.5 mm.

parameter tuning, this value can be increased even further. The studies have shown that a voxel-based model has the ability to learn high-resolution features from 3D models. Especially, it has shown that it is possible to recognize the wall thickness feature despite strong variance in the exact expression of the feature.

Therefore, the next step of our work will be to back up the generated results with further data sets. We want to use the data generated during the production of the AM service provider to generate additional data sets. With that information we can create labeled data sets regarding other geometric features like overhangs, bore holes or channels described by Bikas et al. [9].

4.2 3D-Component Recognition

For 3D-component recognition six different evaluation data sets have been generated based on the Thingi10K data set in order to evaluate the performance of the described algorithms for the recognition of physical components. These data sets contain between 10 and 100 different 3D models and were composed in a way that they represent typical build jobs of the currently common industrial PBF 3D printers for polymer processing. Three data sets contain random compositions of 3D models from the AM domain, the other three explicitly very similar 3D models (Fig. 7). The data sets are available for public download in order to offer other researchers the possibility of comparison¹.

To prove that the approaches we are using are applicable for a real application, we need to adapt it to a possible recognition station setup. After printing, the objects will be separated on a conveyor belt and passed into a scanning area consisting of multiple 2D- or 3D-sensors installed in elevated view-points like

¹ https://gpstd.s3.eu-central-1.amazonaws.com/Raw_datasets.zip.

shown in Fig. 3. The biggest unknown in this system is the orientation in which a component is sensed by the sensors. Therefore, the creation of training data for the 3D ML algorithms is a deciding factor.

Depending on the algorithm, virtual images, point clouds or voxel representations have been generated from the 3D models, which have then been used as training data for the systems. For the creation of these input representations we implemented an algorithm which generates physically sound training data. This means that only virtual sensor views are used, which can also occur in the described setup with real sensors. Our evaluation has shown that this method for training data generation has a major influence compared to randomly generated training data.

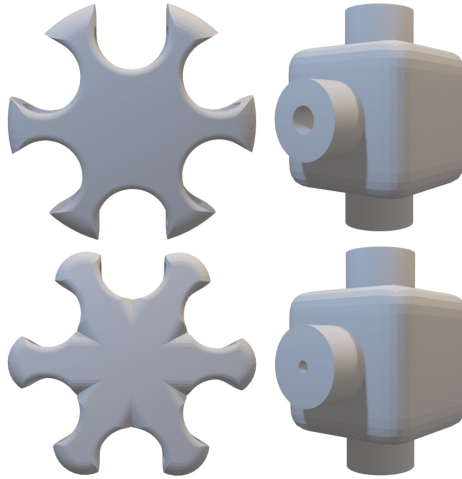


Fig. 7. Example of 3D models with high similarity.

The data sets have been used to train the approaches [14, 15, 17–19, 22] presented in Sect. 3.1. For the data sets with up to 30 different randomly chosen 3D models all approaches reached an accuracy of more than 99%. For higher numbers of different 3D models the accuracy of the image-based approaches decreases to values around 93%. The accuracy of the point-cloud-based approaches can still reach 96% and the voxel-based approach is close to 99%. This confirms the expectations that image-based approaches in particular reach their limits, especially when there are strong similarities between individual objects.

The evaluation experiments have shown that Deep Neural Network (DNN)-based approaches are able to recognize physical objects after production with high accuracy. Especially the generation of physically sound training data brings us one step closer to the goal of a 100% recognition rate and thus to the possibility of complete automation. In the next step, we will evaluate the approaches in the production process of the service provider to move from artificially generated data sets to real production data.

5 Conclusion

In the context of our work we have shown that the process chain of AM offers various aspects that can be optimized with the use of 3D ML algorithms. The studies have shown that state-of-the-art 3D ML algorithms are capable of analyzing 3D models in high detail with regard to various issues. For the sub-processes of manufacturability analysis and 3D-component recognition we implemented prototypes which have shown that these two tasks can be optimized using the proposed 3D ML algorithms. For the production cost calculation we have carried out preliminary studies. These have shown that using 3D ML algorithms for a more precise calculation of the production costs is promising. In our future work we are going to verify this assumption by using the algorithms with real production data.

Our results show that there are different potentials for further automating the process chain of AM. In our research we will continue to build on the studies described in this paper and apply the applications in real production.

Acknowledgements. We gratefully acknowledge the funding of this project by computing time provided by the Paderborn Center for Parallel Computing (PC2). A special thanks belongs to the Phoenix Contact Foundation for supporting this research project.

References

1. Rüßmann, M., Lorenz, M., Gerbert, P., Waldner, M., Justus, J., Engel, P., Harnisch, M.: Industry 4.0: the future of productivity and growth in manufacturing industries. *Boston Consult. Group* **9**(1), 54–89 (2015)
2. Liao, Y., Deschamps, F., Loures, E.D.F.R., Ramos, L.F.P.: Past, present and future of industry 4.0-a systematic literature review and research agenda proposal. *Int. J. Prod. Res.* **55**(12), 3609–3629 (2017)
3. Caffrey, T., Wohlers, T., Campbell, R.: Executive summary of the Wohlers report (2016)
4. King, W.E., Anderson, A.T., Ferencz, R.M., Hodge, N.E., Kamath, C., Khairallah, S.A., Rubenchik, A.M.: Laser powder bed fusion additive manufacturing of metals; physics, computational, and materials challenges. *Appl. Phys. Rev.* **2**(4), 041304 (2015)
5. Woesz, A.: Rapid prototyping to produce porous scaffolds with controlled architecture for possible use in bone tissue engineering. In: *Virtual Prototyping & Bio Manufacturing in Medical Applications*, pp. 171–206. Springer (2008)
6. Bikas, H., Stavropoulos, P., Chryssolouris, G.: Additive manufacturing methods and modelling approaches: a critical review. *Int. J. Adv. Manuf. Technol.* **83**(1–4), 389–405 (2016)
7. Ciobota, N.D.: Standard tessellation language in rapid prototyping technology. National Institute of Research and Development for Mechatronics and Measurement Technique, Bucuresti, *The Scientific Bulletin of Valahia University–Materials and Mechanics*, vol. 7 (2012)
8. Materialise Magics Sinter Module. <https://www.materialise.com/en/software/magics/modules>. Accessed 22 Apr 2020

9. Bikas, H., Lianos, A., Stavropoulos, P.: A design framework for additive manufacturing. *Int. J. Adv. Manuf. Technol.* **103**(9–12), 3769–3783 (2019)
10. Adam, G.A., Zimmer, D.: Design for additive manufacturing–element transitions and aggregated structures. *CIRP J. Manuf. Sci. Technol.* **7**(1), 20–28 (2014)
11. Rudolph, J.P., Emmelmann, C.: Analysis of design guidelines for automated order acceptance in additive manufacturing. *Procedia CIRP* **60**, 187–192 (2017)
12. Zwier, M.P., Wits, W.W.: Design for additive manufacturing: automated build orientation selection and optimization. *Procedia CIRP* **55**, 128–133 (2016)
13. Am-Vision: 3D-Part Recognition. <https://am-flow.com/vision/>. Accessed 19 Feb 2020
14. Kanazaki, A., Matsushita, Y., Nishida, Y.: RotationNet: joint object categorization and pose estimation using multiviews from unsupervised viewpoints. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5010–5019 (2018)
15. Feng, Y., Zhang, Z., Zhao, X., Ji, R., Gao, Y.: GVCNN: group-view convolutional neural networks for 3D shape recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 264–272 (2018)
16. Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J.: 3D shapenets: a deep representation for volumetric shapes. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1912–1920 (2015)
17. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: deep hierarchical feature learning on point sets in a metric space. In: *Advances in Neural Information Processing Systems*, pp. 5099–5108 (2017)
18. Liu, Y., Fan, B., Xiang, S., Pan, C.: Relation-shape convolutional neural network for point cloud analysis. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8895–8904 (2019)
19. Zhang, K., Hao, M., Wang, J., de Silva, C.W., Fu, C.: Linked dynamic graph CNN: learning on point cloud via linking hierarchical features. *arXiv preprint [arXiv:1904.10014](https://arxiv.org/abs/1904.10014)* (2019)
20. Maturana, D., Scherer, S.: VoxNet: a 3D convolutional neural network for real-time object recognition. In: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 922–928. IEEE (2015)
21. Riegler, G., Osman Ulusoy, A., Geiger, A.: OctNet: learning deep 3D representations at high resolutions. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3577–3586 (2017)
22. Shao, T., Yang, Y., Weng, Y., Hou, Q., Zhou, K.: H-CNN: spatial hashing based CNN for 3D shape analysis. *IEEE Trans. Vis. Comput. Graph.* **26**, 2403–2416 (2018)
23. Zhou, Q., Jacobson, A.: Thingi10K: a dataset of 10,000 3D-printing models. *arXiv preprint [arXiv:1605.04797](https://arxiv.org/abs/1605.04797)* (2016)