The 8th European Congress on Computational Methods in Applied Sciences and Engineering
ECCOMAS Congress 2022
5–9 June 2022, Oslo, Norway

DATA-DRIVEN GEOMETRIC MODELLING METHODS FOR DIGITAL TWINNING: MANUFACTURING, GEOSPATIAL AND MEDICAL APPLICATIONS

Oliver J. D. Barrowclough[1], **Sverre Briseid**[1], **Tor Dokken**[1], **Konstantinos Gavriil**[1] **and Georg Muntingh**[1]

[1] SINTEF Digital, Pb. 124 Blindern, 0314 Oslo
oliver.barrowclough@sintef.no, sverre.briseid@sintef.no, tor.dokken@sintef.no
konstantinos.gavriil@sintef.no, georg.muntingh@sintef.no

Key words: Digital Twins, Geometric Modelling, Machine Learning, Artificial Intelligence

Abstract. In recent years there has been an explosion of interest in digital twinning in many disciplines, including the manufacturing, geospatial, and medical domains. A core topic of importance in modelling digital twins, is reconstruction of geometric models from raw data. Despite the diversity of requirements in the vast space of digital twin applications, methods for geometric reconstruction can often be transferred between disciplines with only minor modifications. In this paper we present some recent results related to how advances in machine learning over the last decade can be used for data-driven geometric reconstruction in the medical, geospatial and manufacturing domains.

1 INTRODUCTION

Digital twinning is the process of creating a digital twin: a dynamic virtual representation of a physical object or process that can be used as the basis for making decisions. Although the specific definition of the term “digital twin” varies between authors and disciplines (see e.g., [1, 7]), this definition covers the most important aspects. A core topic of importance in digital twinning, is the modelling and reconstruction of geometric data arising from physical objects, and keeping those geometric representations up-to-date. Such modelling can serve several purposes from detecting and monitoring defects and deviations to tracking objects in space and even generation of new geometries. Geometric modelling for digital twin applications can be distinguished from the wider field of geometric modelling by the explicit need for full and reliable automation.

In the context of geometric modelling for digital twinning, there exist many different data modalities. Raw data often comes in the form of gridded raster data or scattered point clouds. Examples of gridded data include the layered images arising from CT or MR scans in the medical setting or successive photographs taking during an additive manufacturing process. It can also include digital elevation models or RGB-D data that complements images with depth values. In contrast, scattered point cloud data often arises from laser or sonar scans. There are also various methods for changing between scattered and gridded representations. Photogrammetry matches features in pairs of separated images to reconstruct the 3D location of the corresponding features, resulting either in a point cloud or a mesh. On the other hand, scattered point clouds are often rasterized or approximated to a structured data format, either for the purposes of long-term storage, or for further processing. While scattered data representations are important and of interest to the general topic of geometric modelling for digital twin applications, we restrict our attention to gridded data sets here. This is mainly due to the unified learning-based approaches that can be exploited when considering gridded data.

While traditional geometric modelling algorithms tend to process data based on heuristics and tuning of explicitly programmed algorithms based on trial and error, a more recent trend is to exploit the vast power of artificial neural networks (ANNs) for extracting geometric information. Coupled with the availability of large datasets and enough computational power, ANNs provide a method for geometry extraction in situations that were either very difficult or impossible with more traditional approaches. The gridded data we focus on in this paper is particularly suitable as the input to a certain type of ANN known as a convolutional neural network (CNN). CNNs provide a powerful and flexible tool for extracting and processing information from gridded datasets, whether that be for the purpose of segmentation, classification or generation.

In this paper we describe several approaches to data-driven geometric modelling based on the availability of large datasets and the power of combining classical geometric algorithms with machine learning based approaches. We also discuss how certain applications can benefit from hybrid modelling, where physics-based modelling techniques complement the data-driven methods to provide fuller understanding of the physical characteristics of the modelled systems.

2 BACKGROUND

Digital twinning provokes a wide range of challenges, considering different datasets for different problems in different domains. Such challenges necessarily demand a wide range of solutions, but some common threads can be found throughout. In this section we highlight some common challenges and general methods that can be used to solve them independent of the specific field the challenges arise from.

2.1 Data diversity

Geometric data captured for the purpose of digital twinning are diverse, typically covering several data modalities with a varying quality and nature. In the setting of manufacturing there is often nominal 'as-designed' data available in the form of CAD models, normally consisting mainly of geometric primitives (planes, cones, cylinders, etc.) and freeform surfaces such as non-uniform rational B-splines (NURBS), but piecewise linear meshes (i.e., triangular or tetrahedral meshes) are also commonplace in industrial data. On the other hand, data captured during the manufacturing process normally comes in the form of points clouds (e.g., from laser scans), or 2D, 3D, or 4D grids (e.g., images, RBG-D grids from cameras with depth sensors, computer tomography (CT) scans, (functional) magnetic resonance imaging ((f)MRI), etc.). When nominal data is available, comparing it to the captured data is a way of ensuring that deviations in the manufacturing process are controlled and that the fabricated object conforms to prescribed tolerances.

In the medical and geospatial settings there is typically no nominal 'ground truth' dataset, but rather a range of data captured at different points in time, with different capture technologies, covering different parts of the domain, and sometimes with gaps in the dataset. This is for instance the case for cardiac CT-scan, in which a single scan is just a snapshot of an organ under constant deformation. Even when labels are available, they might not cover all the data (e.g., individual images in temporal data). Moreover, different experts will often provide different labels, and there is no single correct label. In these cases the labels can be thought of as samples of a probability distribution that needs to be estimated.

The main challenge in these settings is to combine the datasets in a consistent way, detecting changes and separating uncertainties in the dataset arising from noise or outliers from actual trends in the data.

2.2 Tasks

This paper covers the rather wide topic of how to extract information from gridded data, with a particular focus on information related to the geometry or the shapes embedded in the data. In this paper we focus on three specific tasks that arise in a digital twinning setting: classification, segmentation and generation.

2.2.1 Classification

Classification is the process of determining to which class a specific observation in a dataset belongs. Typically classification algorithms are based on features that represent shape or colour information in the data, whether those features are determined manually (as in classical algorithms) or automatically (as in learning-based approaches). Classification is used for purposes such as determining the contents of an image (see Section 3.2), detecting defects in a manufacturing process (see Section 3.3), and categorizing tumors in cancer patients.

2.2.2 Segmentation

The purpose of segmentation is to divide a dataset into groups according to some predefined criteria. In terms of gridded data, we often aim to partition the cells of the grid into different regions according to shared attributes. Segmentation can be expressed as a per-pixel classification problem, or as a high-dimensional regression problem and the particular representation of the segmentation boundary can be either another grid (e.g., inside/outside binary mask), or a geometric representation such as a spline function; see Section 3.1 for an example of this. Segmentation can be used to monitor the change in shape of an object, for example, monitoring heat-induced deviations during a manufacturing process, or modelling tumor growth over time.

2.2.3 Generation

In contrast to segmentation and classification, the purpose of generative algorithms is to reconstruct or fill-in missing data, or in some cases to create new data points in a dataset entirely. One typically wants to reconstruct features that would be expected to be found in the underlying dataset, based on surrounding data and/or prior knowledge of the characteristics of the dataset. For example, if modelling missing regions of a smooth terrain, spline approximation that is constrained to match the boundary of the missing region can provide good results. Learning-based approaches that use large datasets to extract the features that are typically important for reconstruction can also be used to solve generation problems in a general setting; see Section 3.2 for an example.

2.3 Approaches

Addressing the tasks outlined above can involve a wide variety of techniques, including both traditional approaches and more modern approaches based on machine learning. In general, we can categorize such approaches as being manual, semi-automatic or fully automatic, where both the manual and semi-automatic approaches require significant human input. In this paper we focus mainly on fully automatic approaches that are most relevant in the digital twin setting.

2.3.1 Classical image processing

Prior to the advent of modern machine learning techniques (especially deep learning), image processing techniques were a staple in solving segmentation, classification, and generation problems for gridded data. Image processing covers a wide spectrum of techniques such as edge detection, thresholding, erosion, dilation among others. These techniques are often applied in tandem to solve the specific problem at hand.

2.3.2 Curve, surface and volume modelling

Mathematical modelling of curves and surfaces is also an important tool in understanding the geometric trends in gridded data. By modelling the relationships between the dependent and independent variables (i.e., regression/approximation and interpolation), these methods can provide continuous representations of a model, filling any gaps in the original dataset. Among the plethora of methods that provide such representations are linear regression, least-squares approximation, kriging, inverse distance weighting, etc. Analysis on the reconstructed models can sometimes also reveal features that are difficult to reconstruct from the raw data. Methods such as Fourier transforms are useful for analyzing spatial and/or temporal frequencies embedded in datasets. If it is known that the underlying geometry is smooth, representations such as B-splines can be of particular interest, providing representations with a guaranteed level of continuity.

2.3.3 Machine learning and convolutional neural networks

Machine learning is a broad research field with a long history. While many of the key ideas have been around for several decades, a tipping point arrived about a decade ago when improved algorithms, powerful computational hardware, and the availability of large, open datasets made deep learning applications available to a broader audience. This point has been called “the advent of deep learning”.

Deep learning makes use of deep ANNs, which chain multiple neural network layers with non-linear activation functions together to efficiently represent non-linear functions in high dimensions. ANNs provide a flexible set of approaches for addressing tasks such as classification, segmentation and generation. In this paper we focus on the use of convolutional neural networks (CNNs), which comprise a subset of ANNs that specifically target gridded data. They do this by learning the weights of convolutional filters that 'slide' across the input grid to produce another grid as output; more details can be found in [10]. CNNs are typically deep neural networks consisting of several layers, sometimes first containing contractive layers that encode features in channels. These features can then be used together with fully connected layers in classification algorithms or together with expansive convolutional layers to solve segmentation problems. CNNs can be used in both supervised and unsupervised settings and can be combined with the Generative Adversarial Network (GAN) framework for training generative models that can be used for completing missing data or generating new datasets from a certain distribution.

2.4 Representations

There is no such thing as a single best representation that covers all the purposes a geometric model is made for. In most cases, the best one can ask for is a high degree of interoperability: the ability to efficiently switch between representations with minimal loss of information. Typically geometric data can be categorized into three types, with relative strengths and weaknesses:

- **Explicit representations** model curves and surfaces as functions:

$$\begin{aligned} &\{(x, y) : y = f(x), x \in \Omega \subset \mathbb{R}\}, \text{ for planar curves in } \mathbb{R}^2, \\ &\{(x, y, z) : z = f(x, y), (x, y) \in \Omega \subset \mathbb{R}^2\}, \text{ for surfaces in } \mathbb{R}^3. \end{aligned} \tag{1}$$

  This is the typical representation used for digital elevation models, where each point in the domain has a corresponding single elevation value. Explicit representations are simple and easy to evaluate, but cannot represent more complex geometric features such as overhangs.

- **Parametric representations** provide more modelling flexibility by introducing independent parameters and defining separate functions in each coordinate direction:

$$\begin{aligned} &\{(x, y) = (f_1(t), f_2(t)) : t \in \Omega \subset \mathbb{R}\}, \text{ for curves in } \mathbb{R}^2, \\ &\{(x, y, z) = (f_1(s, t), f_2(s, t), f_3(s, t)) : (s, t) \in \Omega \subset \mathbb{R}^2\} \text{ for surfaces in } \mathbb{R}^3. \end{aligned} \tag{2}$$

  Parametric representations provide flexibility for representing complex geometric behaviour and are typically the preferred representation in a design setting, but lack the ability to easily model topological changes in a model. When modelling scattered data, parametric approximations also require a choice of parametrization which is subjective, and often needs to be tuned to the specific data types.

- **Implicit representations** model geometries by the zero-set of a higher dimensional function:

$$\begin{aligned} &\{(x, y) : f(x, y) = 0\} \text{ for curves in } \mathbb{R}^2, \\ &\{(x, y, z) : f(x, y, z) = 0\} \text{ for surfaces in } \mathbb{R}^2. \end{aligned} \tag{3}$$

  Implicit representations are slower to evaluate, but provide effortless transitions between topological changes in a model; an aspect that is important when modelling data with potentially complex topologies such as organs in a CT-scan.

It is important to use suitable representations to compactly model the data, ensuring that the digital twin remains accurate and responsive to real-time interrogation and visualization. In addition to the type of representation, the specific basis functions used also play and important role in the quality of the model. The careful choice of basis functions also helps keep the amount of data to a reasonable size. In this paper we focus on the following building blocks:

- **Polynomials** provide global representations where all degrees of freedom can add to the precision in any point. Although polynomials can theoretically model continuous data with arbitrary precision, they typically become unwieldy at high degrees.
- **Splines** are piecewise polynomials where the continuity between polynomial pieces is prescribed in some way, allowing the polynomial degree to be kept low. They are typically used in situations where the data exhibits some underlying degree of smoothness. Splines exhibit local support, have good theoretical approximation properties and can be defined on both regular and irregular domains. In this paper we focus on tensor-product splines, whose coefficients can be stored in a grid structure, as they can directly utilize the advantages of CNNs with little or no adaptation.
- **Pixels and voxels** can be considered as piecewise constant splines where each cell evaluates to a single value, independent of where in the cell it is evaluated. They typically provide better representations for discontinuous data, or in situations where smooth and rough data are combined.

Although we focus on gridded data here, meshes are also an important representation of geometric data, particularly for the purposes of visualization and simulation. Interoperability between gridded data and mesh representations is provided by lossy algorithms such as marching cubes, and in the other direction by rasterization.

One issue of note in the gridded data setting is that the curse of dimensionality is very much present. While modern hardware can be used to train CNNs on high-resolution 2D data such as images, high-resolution data in 3D remains a challenge and training directly on 4D data remains out-of-scope except for relatively low resolution or sparsely represented data [5].

3 DIGITAL TWIN APPLICATIONS

Digital twins are emerging in many different domains, but geometric modelling is a core topic that is required in most digital twin settings. Digital twins are typically live assets that need to be kept up-to-date in real-time, and thus generally don't have scope for continual manual input and thereby rely on fully automatic methods. In this section we mention some of our recent results in the domains of medical, geospatial and manufacturing digital twinning and discuss some of the commonalities based on the background described in Section 2.

3.1 Digital twinning in the medical domain

In the medical domain, the digital twin paradigm is becoming more and more widespread, despite a number of ethical concerns it raises [4]. In the general setting medical digital twins aim to support personalized medicine by providing up-to-date digital representations of the status of a patient. This can be targeted towards specific parts of the anatomy, such as tumors or the

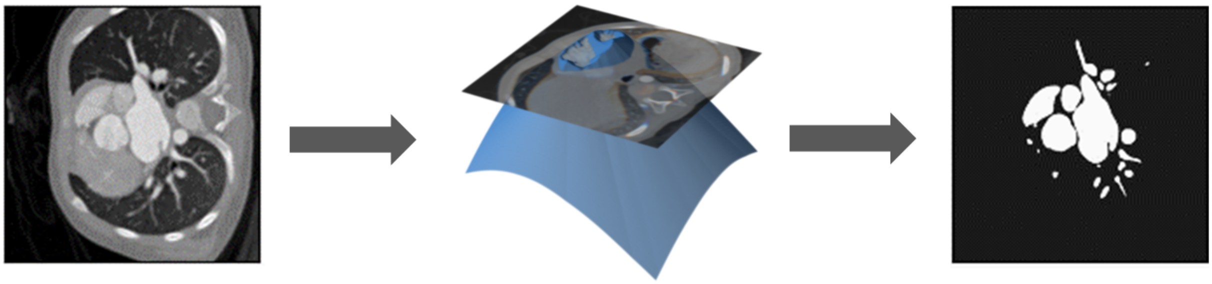

Figure 1: Medical reconstruction workflow: the CNN transforms the input CT images to tensor-product spline coefficients which provide a smooth implicit representation of the segmentation boundary. This can then be sampled at any desired resolution. Middle image extracted from [2], used under Creative Commons license, `https://creativecommons.org/licenses/by/4.0/`.

heart, or can target the body in its entirety, with more and more data being added as and when it becomes available.

In this section we describe how the results presented in [2] fit into the framework described in Section 2. In that paper, the goal was to segment the cardiac blood pool from successive slices of CT images of hearts. The CT data was available in DICOM format, including both the CT-volumes and manually segmented masks for the considered class. The main requirement in this setting is to faithfully represent the connectivity of the blood vessels and chambers, so that surgeons can examine them to get a better understanding of the anatomy prior to surgery. In some cases such digital representations are 3D printed for physical examination, and in others they are presented in augmented reality settings. Another requirement is to provide a smooth representation of the output geometry that reflects the smooth nature of the physical organ. To meet these requirements in a simple way we chose to use implicit representations based on tensor-product splines. Implicit representations allow changes in topology to be made simply by varying the function locally. Splines provide compact representations of the naturally smooth data, which can later be sampled at any desired resolution. Figure 1 shows the workflow of going from an input CT image, to an implicit spline function and then to the binary output. In this context we use a CNN, specifically an adaptation of U-net [9], to create the tensor-product spline coefficients from the input image, as in the first arrow of the figure.

Initially this approach was implemented on a slice-by-slice basis, but similar results were obtained for networks directly operating on 3D voxel grids. Slice-based networks are unaware of information in adjacent slices, which can lead to reconstruction inconsistencies in the slice direction, and 3D networks have the potential to overcome this issue. However, while CT slices typically have a constant resolution (e.g., 512×512), one challenge for 3D networks is to handle a varying number of slices. This can be resolved by either resizing CT volumes to a fixed volumetric resolution, or by tiling the volumes and training a CNN on a fixed tile size and stitching the predictions together. The first approach can lead to loss of detail and introduce anisotropy, while the second approach can limit simultaneous awareness of distant features.

Although this work was performed on CT images, the concepts are general and can be applied both in the wider medical setting (for example to live reconstruction from ultrasound images), and to other problems entirely (see Section 3.3 for an example).

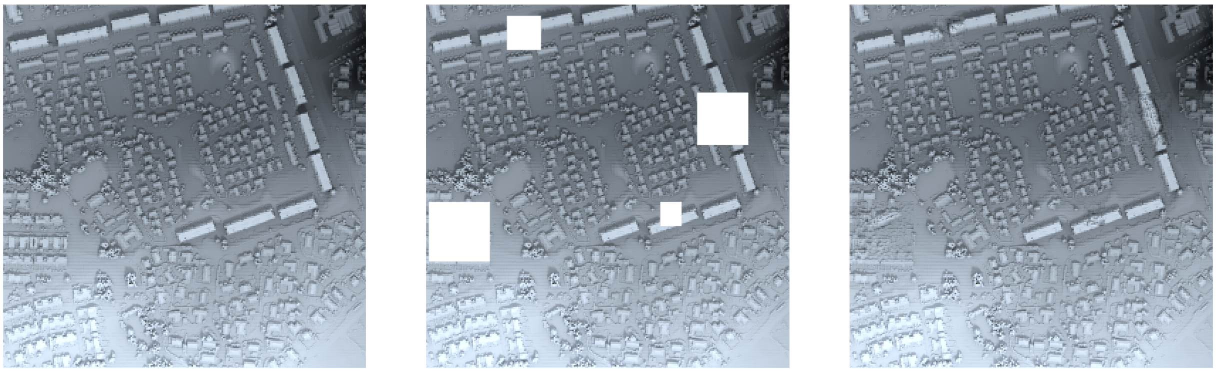

Figure 2: Void filling in the urban setting. Left: original data from the test data set, middle: removed portion in context, right: reconstruction with the data-driven approach for filling voids in DEMs presented in [6]

3.2 Digital twinning in the geospatial domain

In the geospatial domain, digital twins can be used for the purposes of modelling pollution and weather in urban settings, modelling the effects of fertilizer runoff in the agricultural sector and monitoring the dynamics of ecosystems for preserving biodiversity, to name just a few. Specific approaches of interest in this setting include computational fluid dynamics for modelling flows, data fusion for combining disparate geospatial datasets, and change detection for modelling changes in topography over time. Although the majority of raw geospatial data is captured in the form of point clouds, it is generally converted to a gridded format for the purposes of processing, storage and exchange. A common format for the gridded representation is GeoTIFF, which allows georeferencing information to be stored alongside the raster data. We have considered two applications in the geospatial setting that make use of data-driven geometric modelling, described in more detail in [6] and [3].

The former is an approach for filling missing data in digital elevation models (DEMs) by training a generative CNN that uses contextual information about the surrounding terrain to complete the missing data. We train two different models, one targeting digital surface models (DSMs) in the urban setting and one targeting digital terrain models (DTMs) in the rural setting. The difference between DSMs and DTMs is that while DTMs only represent the underlying terrain, DSMs represent the height of the highest static object connected to the ground, whether that is a building, a tree or another structure. Since the characteristics of the data are very different in these two settings, it is natural to train different models to tackle the different problems, despite the algorithmic approach being identical in both settings. The approach is based on a method for image inpainting that uses GANs: a method to train a generative model in conjunction with a discriminator that judges the output of the generator [11]. In [6] we transferred this approach to the problem of filling gaps in digital elevation models based on the surrounding terrain. In Figure 2 we present an example of the approach in the very challenging setting of urban DSMs. The example is taken from a test data set separate to the one on which the model was trained. As can be seen in the reconstructed image on the right, although some artefacts remain, the method captures the general trends in the data with the size and angle of buildings modelled relatively well. This is made possible through the use of attention

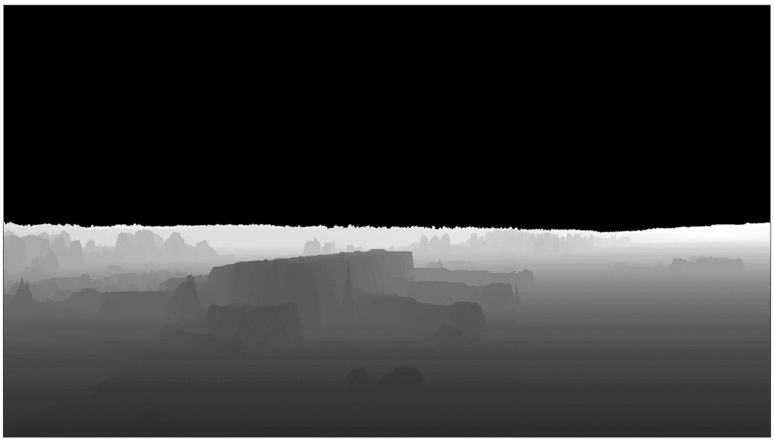

Figure 3: Depth map of a calibrated camera at Sundsvall airport, Sweden. The taxiway, terminal building and multiple tree lines can be seen from the increasing intensity of the pixels (black values represent infinite distance, i.e., sky). Depth maps based on elevation data, © Lantmäteriet, Sweden.

mechanisms, which point to similar features elsewhere in the image to aid reconstruction. In order to improve the reconstructions, the approach could in principle be combined with other geographic data such as cadastral data, which could be stored in other channels of the grid to infer more detailed information on the missing region. The approach could also include, e.g., data on the presence of species, historical flooding data or traffic flow information to be used in a predictive capacity for solving problems such as those related to modelling of biodiversity mentioned above, by learning features of the interrelations between the various layers.

The second geospatial application we have considered combines video data from multiple cameras with depth values for each pixel to determine the 3D location of objects moving on the surface of an airport [3]. This process uses a CNN-based object detection approach known as YOLO (v3) to simultaneously infer the location and class probability of an object via regression [8]. We determine the unknown parameters of the cameras (e.g., direction, field of view, etc.) by matching features in the video stream with those present in the 3D model. The depth values can then be created by rendering a digital elevation model with the depth buffer active, and saving the depth values in a separate raster file, see Figure 3. Combining the object detections over time provides tracking information, which can in turn be used with the depth values to determine the 3D position, speed and direction of objects moving on the airport surface.

3.3 Manufacturing digital twins

In manufacturing, the requirements for digital twins are predominantly geared towards accuracy and reliability. There is typically a need to monitor manufacturing processes to ensure conformance to quality criteria.

One particular use case considered in the Change2Twin project (`https://www.change2twin.eu`) aimed to model the physical deformations that occur in laser-based powder bed fusion (L-PBF) additive manufacturing processes, based on in situ imagery. One of the main drivers behind additive manufacturing is the ability to fabricate complex objects, so there is a need to accurately reconstruct the intricate geometries that can occur. Based on the in situ imagery,

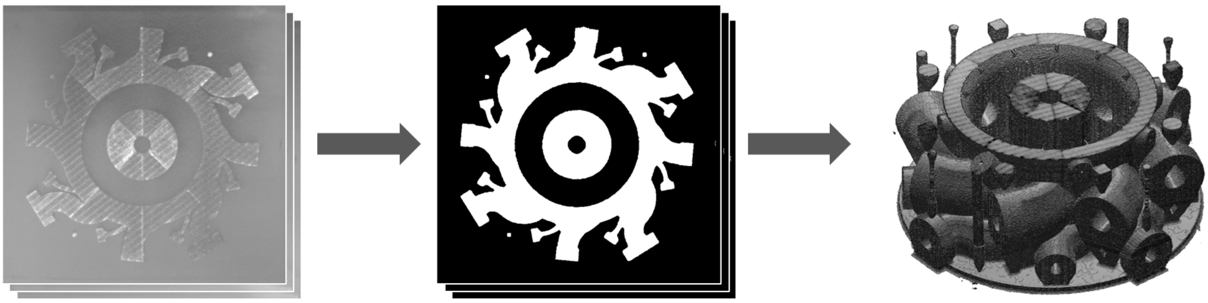

Figure 4: Additive manufacturing reconstruction workflow: from raw images to binary masks to textured voxel model. Data courtesy Additive Industries B.V.

the main challenge is to segment the parts of the image that correspond to melted and unmelted powder. As can be seen in Figure 4, the workflow of this problem reflects the workflow for medical image segmentation presented in Section 3.1, but there are three major differences:

- The segmentation should be as geometrically accurate as possible, to check for deviations from the 'as-designed' CAD model. In the medical segmentation setting this is not the case, as a CT-scan typically corresponds to a snapshot of an organ that is under constant deformation due to internal and external factors — as mentioned, the correct topology is more important in that case.

- Manufacturing data tends to be proprietary, so it is more challenging to get hold of the large amounts of data required to train deep CNNs, which often contain hundreds of thousands or even millions of weights. In addition, due to this being a novel application, there did not exist any manually segmented datasets on which to train deep learning models. In the medical setting, manual segmentation by radiologist is the classical approach so there already exist large databases of ground truth data that can be made available.

- Although CAD models are dominantly made of smooth geometric primitives and freeform surfaces, they almost always contain sharp corners where smooth parts of the model intersect. These models are therefore not well suited to the use of globally smooth representations like the splines used in the medical setting.

The first challenge in this context was therefore to create binary masks that segment the pixels in the image, and that could eventually be used as ground truth for training a deep CNN, once enough data was available. After amalgamation of the slices, the result of this process is therefore a binary voxel grid, which can be textured with the original imagery to provide a 3D model that can be visualized both on the surface and internally.

In order to avoid costly manual segmentation, we developed an approach based on classical image processing techniques that used the variation in intensity of the melted and unmelted powder to segment the images. The workflow consisted of canny edge detection, binary thresholding and a contour finding algorithm with a subsequent filling step. For some of the models an initial gamma correction step was needed in order to improve the contrast and brightness distribution, resulting in more stable algorithms in the subsequent steps. The parameters of the various algorithms were manually tuned to give good results with a test suite of different models originating from the same setup.

Although the tuning is done manually for this specific dataset, the result is an algorithm that is fully automatic. The algorithm can be applied to each slice individually, although less noisy results can be achieved by including a weighted average over several slices adjacent to the slice of interest. This improvement in the uniformity of the results comes at the slight expense of the ability to model very fine geometric features in the model such as thin channels.

In the long term, this approach should probably be complemented by a certain level of manual correction to improve the ground truth data before being used as a training data set for a neural network-based approach. The image processing algorithms are many times more compute intensive than CNN-based approaches of corresponding resolution and are therefore slower (although still much faster than a single layer of the manufacturing process). The approach described in this section can also be combined with classification approaches for detecting different types of defects that can occur, in order for process controllers to intervene at an early stage if the process becomes unstable. It is possible to visualize detected defects together with the 3D-model to provide a complete digital twin environment for monitoring the process in real-time.

Although this approach provides results that are relatively robust, as well as being visually appealing, there are some details that need to be improved for it to be used in production. One issue is that some of the deformations occur during the cooling of the material, and this cooling continues after the next layer of powder has been deposited, meaning there is no information about this contained in the data we have available. In order to account for this in future versions it is therefore necessary either to include physics-based modelling of the cooling process (which is relatively well understood [12]) in the pipeline, or to develop a more thorough data-collection pipeline that also includes data from after the process is complete (e.g., CT-scan data of the final part, in which the effects of residual stresses will be present). In the latter case, by learning correlations between the various in-process and post-process data types, it may, in the long term, be possible to infer the CT-scans directly from the in situ data.

4 CONCLUSIONS

In this paper we discuss how convolutional neural networks provide a comprehensive tool for solving geometric modelling problems in a variety of fields where gridded data is present. The problems are characterized by their need to be deployed in digital twin settings, which typically means the methods need to be fully automatic. Although data-driven methods can provide solutions that were unthinkable only a decade ago, there is still the need to acknowledge that in some cases the data does not encompass the full dynamics of the system to be modelled. It is therefore important to consider integration with physic-based approaches, in cases where the physics is well understood, or to ensure that the data contains sufficient information about the object or process to be modelled faithfully.

ACKNOWLEDGEMENTS

We would like to thank Additive Industries B. V. for their support and for providing data for the development of the manufacturing digital twinning solution presented in Section 3.3.

The research leading to these results has been partially funded by the European Union's

Horizon 2020 Research and Innovation Programme under Grant Agreement numbers 951956 (Change2Twin) and 874470 (SESAR PJ05-W2), as well as by an IKTPLUSS grant from the Research Council of Norway, project number 270922 (ANALYST).

REFERENCES

[1] AIAA Digital Engineering Integration Committee. Digital Twin: Definition & Value. *American Institute of Aeronautics and Astronautics, Inc.*, 2020.

[2] O.J.D. Barrowclough, G. Muntingh, V. Nainamalai & I. Stangeby. Binary segmentation of medical images using implicit spline representations and deep learning. *Computer Aided Geometric Design*, 85, 101972, 2021.

[3] O.J.D. Barrowclough, S. Briseid, G. Muntingh & T. Viksand. Real-time processing of high resolution video and 3D model-based tracking in remote tower operations. *SN Computer Science* 1, Article number: 296, 2020.

[4] K. Bruynseels, F. Santoni de Sio & J. Van den Hoven. Digital twins in health care: ethical implications of an emerging engineering paradigm. *Frontiers in genetics*, 31, 2018.

[5] C. Choy, J. Gwak, S. Savarese. 4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3075–3084, 2019.

[6] K. Gavriil, G. Muntingh & O.J.D. Barrowclough. Void filling of digital elevation models with deep generative models. *IEEE Geoscience and Remote Sensing Letters*, 16(10), pp. 1645–1649, 2019.

[7] A. Rasheed, O. San, and T. Kvamsdal. Digital twin: Values, challenges and enablers from a modeling perspective. *IEEE Access* 8: pp. 21980–22012, 2020.

[8] J. Redmon & A. Farhadi. Yolov3: An incremental improvement. *arXiv preprint*, arXiv:1804.02767, 2018.

[9] O. Ronneberger, P. Fischer & T. Brox. U-net: Convolutional networks for biomedical image segmentation. *International Conference on Medical image computing and computer-assisted intervention*. Springer, Cham, 2015.

[10] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke & A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.

[11] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, & T.S. Huang. Generative image inpainting with contextual attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5505–5514, 2018.

[12] T.I. Zohdi. Modeling and simulation of cooling-induced residual stresses in heated particulate mixture depositions in additive manufacturing. *Computational Mechanics*, 56(4), pp. 613–630, 2015.