Machine Learning Boosted Data-driven Modeling and Simulation of Additive Manufacturing: Process, Structure and Property

by

Zhuo Wang

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Mechanical Sciences and Engineering)
in the University of Michigan-Dearborn
2021

Doctoral Committee:

Assistant Professor Lei Chen, Chair
Assistant Professor Zhen Hu
Professor Pravansu Mohanty
Research Professor Mihaela Banu, University of Michigan

Zhuo Wang

zwg@umich.edu

ORCID iD: 0000-0002-1657-1890

Dedication

To my family

Acknowledgements

I would like to thank:

- my advisor, Dr. Lei Chen, for his advising and support throughout my five-year doctoral study.
- group members (both current and past) in Dr. Chen's research group. I was so lucky to have you to get through this wonderful journey together.
- all of my collaborators. All the research achievements that I have made are definitely team efforts. Without your great contribution, I wouldn't have achieved any of them alone.
- instructors and professors of every course that I took during my graduate study.
- Mississippi State University where I spent the first three years of my Ph.D. Especially, many thanks to Mitchell Memorial Library that opens till midnight.
- High Performance Computing Collaboratory (HPC^2) at Mississippi State University and Advanced Research Computing (ARC) at University of Michigan that produce big simulation data for my research.
- National Science Foundation (NSF), Michigan Institute of Data Science (MIDAS), as well as my department - Mechanical Engineering department at University of Michigan-Dearborn, for the financial support for my study and research.
- fitness centers at Mississippi State University and my community that keep me energetic for carrying out researches.

Table of Contents

List of Tables

List of Figures

List of Appendices

Abstract

Additive manufacturing (AM), which builds a single part directly from a 3D CAD model in a layer-by-layer manner, can fabricate complex component with intricate geometry in a time- and cost-saving manner. AM is thus gaining ever-increasing popularity across many industries. However, accompanied with its unique building manner and benefits thereof are the significantly complicated physics behind the AM process. This fact poses great challenges in modeling and understanding the underlying process-structure-property (P-S-P) relationship, which however is vital to efficient AM process optimization and quality control. With the advancement of machine learning (ML) models and increasing availability of AM-related digital data, ML-based data-driven modeling has recently emerged as a promising approach towards exhaustively exploring and fully understanding AM P-S-P relationship. Nonetheless, many of existing ML-based AM modeling severely under-utilize the powerful ML models by using them as simple regression tools, and largely neglect their distinct advantage in explicitly handling complex-data (e.g., image and sequence) involved data-driven modeling problems and other versatilities.

To further explore and unlock the tremendous potential of ML, this research aims to attack two significant research problems: (1) from the data or pre-data-driven-modeling aspect: can we use ML to improve AM data via ML-assisted data collection, processing and acquirement? (2) from the data driven modeling aspect: can we use ML to build more capable data-driven models, which can act as a full (or maximum) substitute of physics-based model for high-level AM modeling or even realistic AM simulation?

To adequately address the above questions, the current research presents a ML-based data-driven AM modeling framework. It attempts to provide a comprehensive ML-based solution to data-driven modeling and simulation of various physical events throughout the AM lifecycle, from process to structure and property. A variety of ML models, including Gaussian process (GP), multilayer perceptron (MLP), convolutional neural network (CNN), recurrent neural network (RNN) and their variants, are leveraged to handle representative data-driven modeling problems with different quantities of interest (QoI). They include data-driven process modeling (melt pool, temperature field), structure modeling (porosity structure) and property modeling (stress field, stress-strain curve). The results show that this research can break existing limitations of those five data-driven AM modeling in terms of modeling fidelity, accuracy and/or efficiency. It thus well addresses the two research questions that are key in significantly advancing data-driven AM modeling. In addition, although the current research uses five representative physical events in AM as examples, the data-driven methodologies developed should shed light on data-driven modeling of many other physical events in AM and beyond.

Chapter 1 Introduction

1.1 Additive manufacturing

Figure 1.1 Wide application of AM in various industries. (Images downloaded from: www.buggatti.com, www.all3dp.com, www.time.com, www.adidas.com, www.alighntech.com)

Additive manufacturing (AM) or 3D printing, as opposed to conventional subtractive manufacturing process such as forging and milling, enjoys the uniqueness of layer-wise building up complex component or even assemblies as a single part while requiring little follow-on machining. Components with rather intricate geometry [3, 4] can be thus effortlessly fabricated with remarkable time and cost savings, which renders AM potent and increasingly popular in many industries; see Fig. 1.1. Various AM techniques can be classified into seven main categories – material extrusion, powder bed fusion, photopolymerization, material jetting, binder jetting, sheet lamination, and directed energy deposition [5]. Regardless of printing material, feeding fashion and heating sources used for binding raw materials and other distinctions among them, all AM processes share the signature of layer-by-layer incremental/additive building that gives its name.

1.2 Complicated AM process

While the unique building style endows AM unrivaled manufacturing capabilities, those benefits come at a cost of significantly complicated AM process and miscellaneous physics behind. A plethora of physical events, which rarely occur in conventional manufacturing process, exists throughout the AM lifecycle from process to structure and property. For example, the known thermal cycles by layer-wise manufacturing would complicate the phase evolution during AM [6]. Another typical example, as shown in Fig. 1.2a, is that the high-power beam used to consolidate printing materials will result in in-process elongated melt pool featuring steep thermal gradient around, leading to epitaxial grain growth and thus large columnar grain structure during layer-by-layer building [7, 8]. This in turn results in anisotropic mechanical properties that will further adversely affect elongation behavior and fatigue life of final AM products [9]. In addition, improper power or scanning speed of applied heating sources will easily cause various types of porosity in as-built AM parts, such as lack-of-fusion by insufficient melting [10, 11], surface open pore by complex fluid dynamics [12, 13] and keyhole by strong metal evaporation [14], as shown in Fig. 1.2b. Other physical complexities include balling, powder denudation, cyclic heating and remelting, super-fast cooling and non-equilibrium solidification, to name a few.

It is widely accepted that a thorough study of process-structure-property (P-S-P) relationship in AM will hugely facilitate AM process optimization and quality control, thereby benefiting wide adoption of AM techniques. However, the aforementioned complex physics and associated physical phenomena underlying AM process result in extremely non-linear, complicated P-S-P relationship. Therefore, an accurate quantification and thorough understanding of AM P-S-P linkage remain to be challenging.

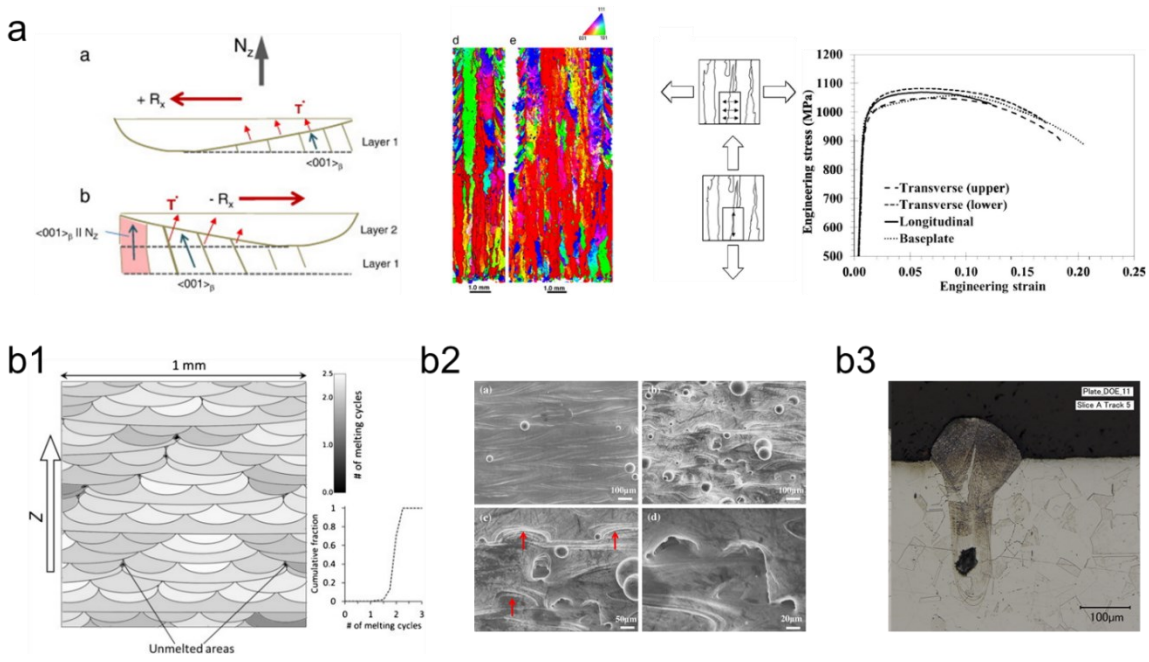

Figure 1.2 The complicated AM process contains various unconventional physical phenomena, making AM P-S-P relationship fairly complex.

1.3 AM process-structure-property relationship study

In spite of the complex P-S-P relationship in AM, numerous experimental and modelling efforts have been invested to advance the fundamental understanding of AM mechanisms. For the experiment-based AM study, different experimental characterization techniques have been implemented to observe and understand various physical activities during AM process even in real-time. Common experimental techniques used in AM include, but not limited to, SEM for powder particle characterization [15], pyrometer and infrared camera for thermal characterization (melt pool and temperature field) [16], high-speed X-ray imaging for porosity monitoring [17], EBSD for post-process grain structure examination [18], SEM for surface morphology observation [19], electronic universal testing machine for mechanical property testing [20]. Some of them are illustrated in Fig. 1.3a-c. Admittedly, moderate success has been achieved in probing underlying physics and AM mechanisms through experimental approaches. The experiment-based AM study,

in general, still severely suffers from its high cost, as well as limited observation resolution and capability. For example, current experimental equipments only allow for observing surface temperature during thermal characterization and final shot of grain structure instead of real-time monitoring.

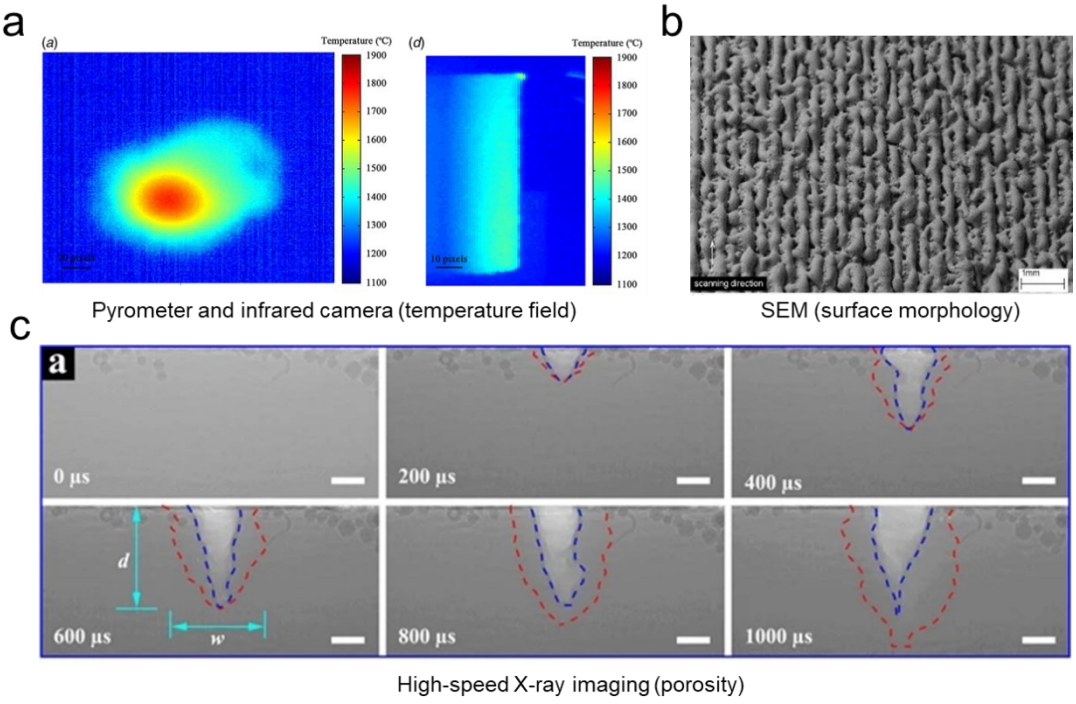

Figure 1.3 Some experimental approaches for studying AM process: (a) pyrometer and infrared camera for temperature field observation; (b) SEM for surface structure examination; (c) high-speed X-ray imaging

On the other hand, modelling-based AM study can serve as an effective alternative to overcome the above shortcomings. Modelling-based AM efforts include physics-based modelling and data-driven modeling. For the physics-based modeling, numerous physical or multi-physical models have been developed for AM modeling and simulation across different time and length scales (as illustrated in Fig. 1.4a [21]), especially in light of the rapid advancement of computation power. Fig. 1.4b provides another typical multi-scale AM physical models for modeling the entire

P-S-P. In this framework, a FEA-based thermal model is used to simulate temperature field evolvement, which are then fed to phase-field based grain growth model to study development of grain microstructure whose mechanical behavior is finally modeled using a fast-Fourier-transform (FFT) elasto-plasticity model [22]. In addition to those, there exists a wide range of physical AM models for modeling various physical events, such as rain-fall model for powder deposition modeling [23], Computational Fluid Dynamics (CFD) model for high-fidelity coupled thermal-&structural-evolution modeling [24], Cellular Automation (CA) for grain structure modeling [25], finite-element (FE) based crystal-plasticity model for mechanical behavior modeling [26], etc. Physics-based modeling can tremendously reduce the time and/or cost and defeat some inherent limitation of a solely experiment-based approach. However, nowadays it is still problematic to carry out large-scale physics-based modeling and simulation, especially multi-scale multi-physical ones that are however inevitable for studying the entire P-S-P, as demonstrated in Fig. 1.4.

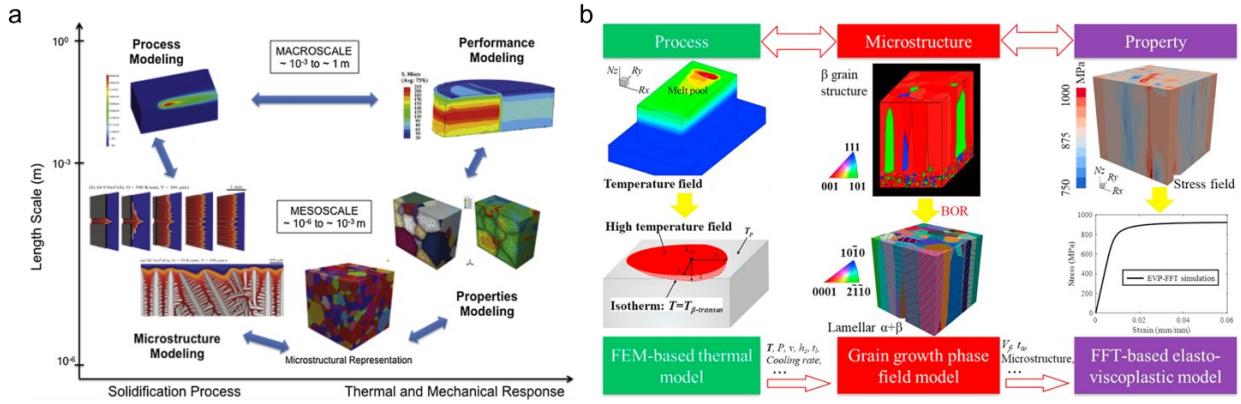

Figure 1.4 Two typical multi-scale physical AM models for studying AM process.

This problem is further compounded by the fact of numerous tunable AM parameters (dozens [27] to as many as 130 [28, 29]) and thus vastness of the AM-parameter design space. Consequently, a physics-based parametric study for exhaustive exploration of P-S-P relationship

over the high-dimensional AM design space is computationally difficult or oftentimes impossible. In that case, data-driven AM modeling comes into the picture, as detailed in the following.

Chapter 2 Background

In the context of the fourth research paradigm of data-intensive discovery [30], research methodologies across science and engineering have seen a shift to data-driven and informatics approaches. This is in part due to the increasing availability of digital scientific data or data deluge [31], both theoretical and experimental, with the fast advancement of computational power and experimental instrumentation. On the other hand, with the revival of artificial intelligence (AI) researches from AI winter [32], the rapid development of machine learning (ML), a disruptive data-driven modeling technique, has especially accelerated such paradigm shift sweeping through many scientific disciplines [33-39]. It should be stressed that, although becoming more and more popular, data-driven modeling is not a standalone approach and, instead, augments physical modeling and experiments by making best use of the generated data thereof. Specifically, data-driven modeling can have different types for diagnostic, descriptive, predictive and prescriptive purposes, respectively [40, 41].

Among others, additive manufacturing (AM) is arguably one of the most affected domains in the age of data. For example, data-driven model has appeared frequently as a key building block of various AM design and management frameworks/strategies, such as design for additive manufacturing (DfAM) [42, 43], digital twin (DT) for AM [44, 45], smart additive manufacturing [46], cloud additive manufacturing [47], etc. While applications of data-driven model in AM are diverse, data-driven predictive modeling is especially useful for studying P-S-P relationship in AM. Fig. 2.1 illustrate the utilization of data-driven model to link various input features and quantity of interest in AM. Therefore, data-driven modeling usually discovers the P-S-P linkage

in a mathematical form (or the predictive function) via regression analysis. More specifically, data-driven predictive modeling allows for automatic exploration of pattern and trend in the AM data, construction of P-S-P relationship over the parameter space and then prediction at unseen points without having to perform new physical modeling or experiments. This data-driven attempt of training a cheap relationship model to replace original physical modeling or experiment is sometimes also called surrogate modeling or metamodeling [48]. Such a data-driven modeling approach is crucial for achieving AM process optimization based on a complete, quantitative understanding of P-S-P relationship.

In what follows, some important data-driven modeling for AM P-S-P relationship study in literature will be briefly reviewed. While there is no widely accepted classification of related data-driven AM modeling, to facilitate discussion, this research broadly classifies them into three types - process modeling, structure modeling and property modeling according to the modeling quantity of interest or output of data-driven modeling; see Fig. 2.1.

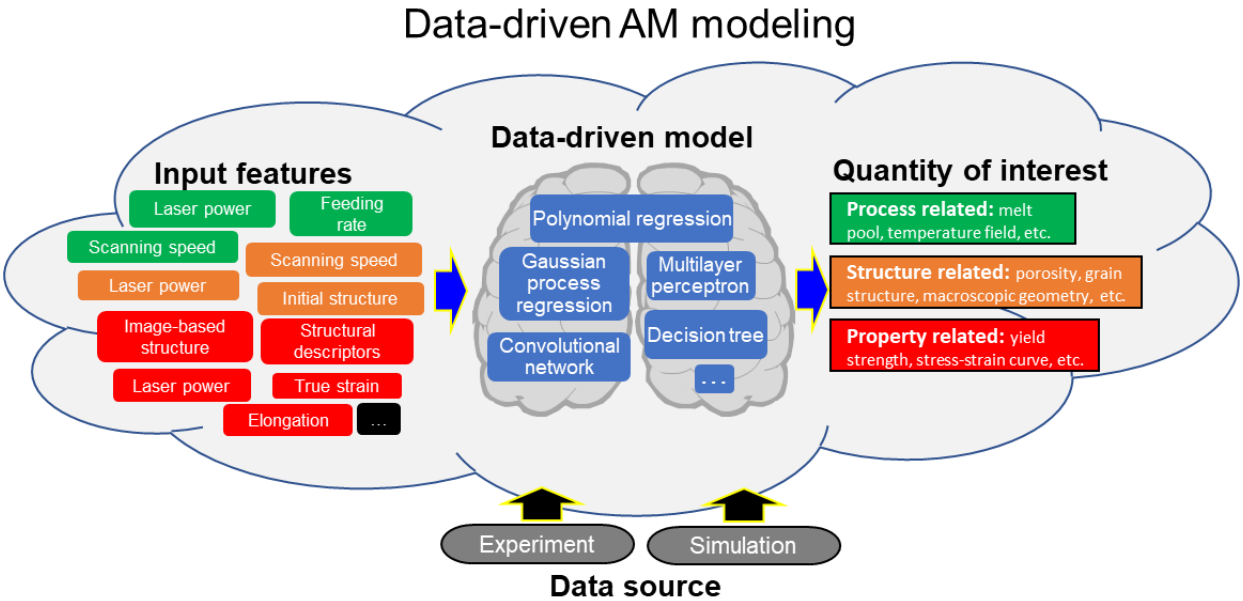

Figure 2.1 Schematic illustration of data-driven AM modeling. Data-driven AM modeling usually consists of four basic components - input features, quantity of interest (output) and data-driven model used to link them, as well as the data source that fuels data-driven modeling.

2.1 Data-driven process modeling in AM

In this research, process modeling is loosely defined as those modeling focused on any physical event occurring before the second stage of structure development. Some typical examples include powder bed deposition modeling (in powder bed-based AM), heat source modeling, melt pool modeling and temperature field development modeling, etc. Most of them are highly related to thermal aspect of AM, which naturally happens before and will affect structure development in next stage.

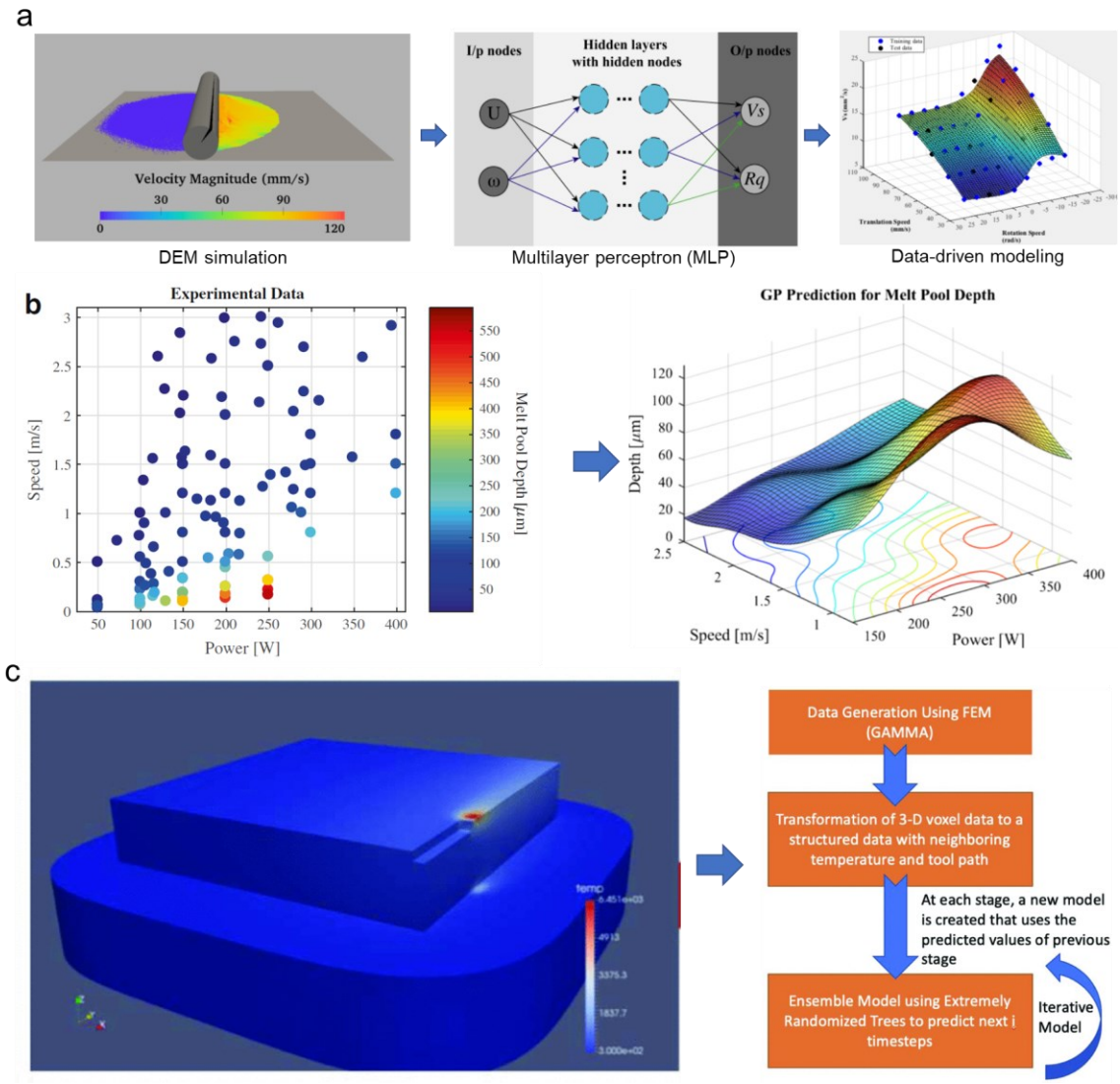

Figure 2.2 Data-driven process modeling of (a) surface roughness of as-deposited powder bed; (b) melt pool depth; (c) voxel-wise temperature during AM building.

Some existing researches on data-driven process modeling are summarized in Fig. 2.2. Zhang et al. [49] has used a multi-layer perceptron (MLP) to build powder bed surface roughness as a function various powder spreading parameters such as spreader speed, with data obtained from repetitive Discrete Element Method (DEM) powder spreading simulations (Fig. 2.2a). Tapia et al. [50] has performed data-driven modeling of melt pool (depth) using Gaussian process regression model, with experimental data from ex-situ measurements. A pool depth map associated with laser power and scanning speed has been thus built (Fig. 2.2b). Paul et al. [51] have performed data-driven modeling of voxel-wise temperature during AM by using extremely randomized trees (ERTs) – an ensemble of bagged decision trees as the regression algorithm (Fig. 2.2c). The real-time temperature of a given voxel was modeled based on data of its historical thermal information and so on.

2.2 Data-driven structure modeling in AM

Structure modeling, as its name indicates, precisely refers to those modeling of structure evolution and development. Structure modeling can be easily identified and differentiated from other types of modeling by its quantity of interest (QoI), which are various structures of AM building across different scales, ranging from dendrites, grain, and porosity to surface structure/roughness and even the bulky shape of the whole AM component.

As shown in Fig. 2.3a, in a multi-stage data-driven model, Gan et al. [52] has built the relationship between dendrite arm spacing and various AM process characteristics using a data-mining model – self-organizing map (SOM). Wang et al. [53] has successfully correlated the developed grain structure, described by mean and variance of grain size ratio, with as many as nine AM parameters by using Gaussian process regression; hundreds of multi-physical simulations were performed to provide enough data for data-driven modeling, as illustrated in Fig. 2.3b.

Similarly, Popova et al. [54] has correlated grain structure, but quantitatively represented by grain chord length distribution, with various AM process parameters using multivariate polynomial regression; see Fig . 2-3c.

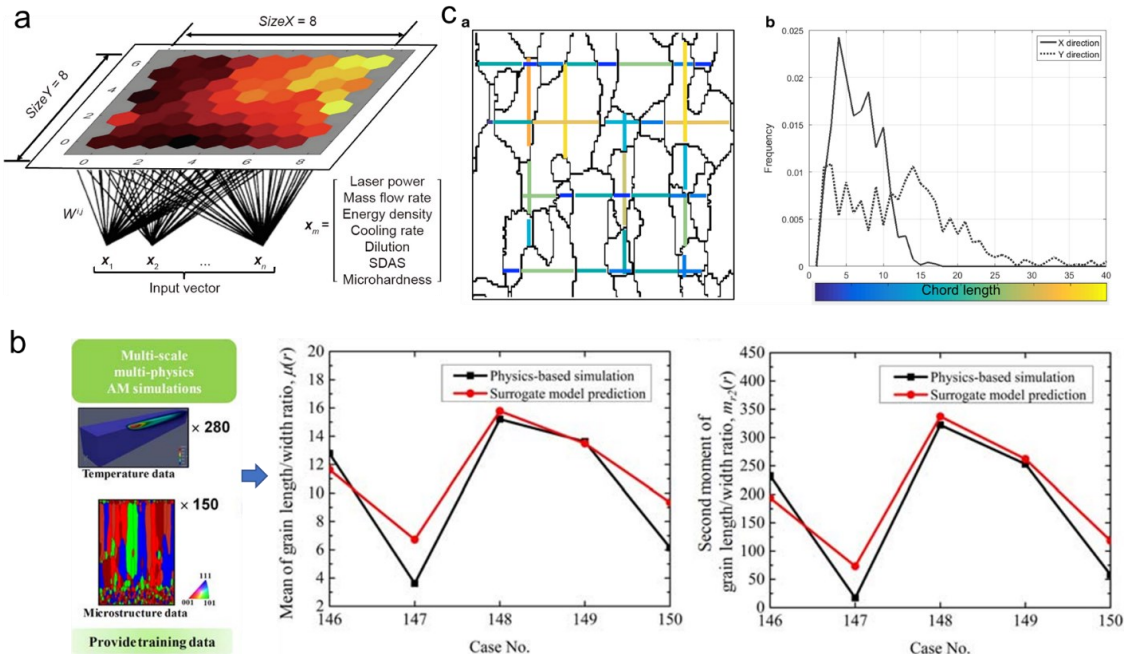

Figure 2.3 Data-driven structure modeling of (a) dendrite arm spacing; (b) grain microstructure represented by mean and variance of grain aspect ratio; (c) grain microstructure represented by chord length distribution.

2.3 Data-driven property modeling in AM

Likewise, property modeling can be clearly defined as those modeling interested in mechanical properties, as implied by its QoI being different mechanical properties of AM parts.

As shown in Fig. 2.4a, Gupta et al. [55] have utilized polynomial functions to construct macroscale performance of inclusion-contained structures (yield strength, strain hardening exponent, localization propensity) as functions of its structure descriptors obtained from n-point spatial correlation and principal component analysis. A large number of FE-based solid-mechanics simulation are performed to provide sufficient data. Garg et al. [56] have adopted three regression

algorithms to build compressive strength of AM parts as functions of five process parameters (Fig. 2.4b). Thirty-two data points based on empirical modelling of the AM process were used for data-driven modeling. Herriott et al. [57] have further take advantage of convolutional network in directly "understanding" structure, thus data-driven modeling of its yield strength without human processing (featurization) of complex structures; see Fig. 2.4c.

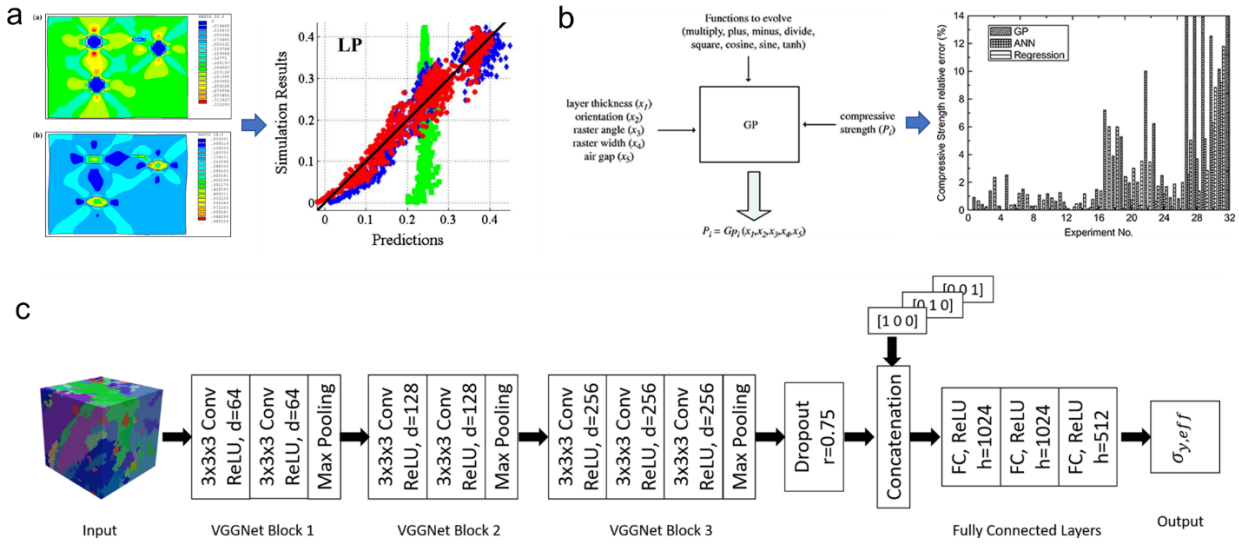

Figure 2.4 Data-driven property modeling of (a) macroscale performance (yield strength, strain hardening exponent, localization propensity); (b) compressive strength of AM parts; (c) yield strength of grain microstructure within AM parts.

2.4 Limitation of existing data-driven modeling in AM

It is admitted that noticeable progress has been recently made in data-driven AM modeling. However, existing applications of ML for data-driven AM modeling still suffers from some critical limitations and thus brings up two important questions:

1) Existing data-driven AM modeling uses ML model merely as a predictive modeling tool for data-driven AM modeling, while overlooking its multi-functionality and enormous versatility. For example, as shown in Fig. 2.5, AM data of experimental images often contain noises requiring further processing that is too laborious for human, whereas some ML techniques are specified for effective image processing. It thus raises the research question – *can we enhance*

data-driven AM modeling by using ML techniques as more than a AM modeling tool and, for example, from the pre-data-driven-modeling aspect of improving AM data via ML-assisted data collection, processing and/or obtainment?

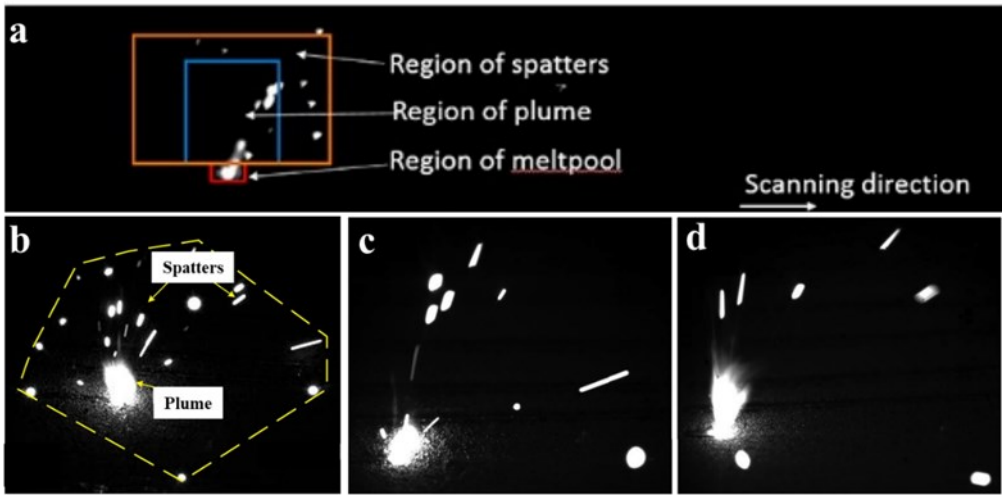

Figure 2.5 A typical noisy AM data – high-speed melt pool image with noise of spattering and plume. Those noise prevents accurately extracting melt pool information (e.g., pool length, width and area) for further data-driven modeling. Pictures adapted from (a) [5] (b) [30], and (c)(d) [28].

2) In many data-driven AM modeling, ML models are used for simple regression analysis that falls within the task of naive curve-fitting and thus the capability of conventional regression techniques. The distinct advantage of ML model in explicitly handling complex-data involved data-driven tasks, such as CNN and GAN for high-dimensional image/field data and RNN for sequential data, is sometimes neglected. For example, for porosity-related data-driven modeling, what one nowadays does is usually using ML to simply build porosity level (or other simple porosity metrics) as a function of AM process parameters; see Fig. 2.6a. It thus poses a serious question – can we use ML model to do realistic simulation of structural/morphological evolution and development of porosity structure (as does a physical model)? For instance, in fluid community, different CNNs with encoder-decoder structure have been popularly used for simulating the evolution and development of flow field [58, 59]. In solid mechanics modeling, CNN [60] and GAN [61] can accurately predict the stress field developed for given solid structure and loading condition.

Admittedly, image-classification-type CNN are now not uncommon in various AM data-driven modeling, but advanced applications of other CNNs, GANs, RNNs and even integrated CNN+RNN for higher-level and higher-quality data-driven AM modeling, are

relatively limitedly explored. The above facts reflect the under-utilization of powerful ML techniques in AM nowadays. Therefore, a more general question raised herein is – *can we take more real advantage of various ML techniques to build data-driven models that can act as a full (or at least maximum) substitute of physics-based models for high-level AM modeling and even realistic AM simulation?*

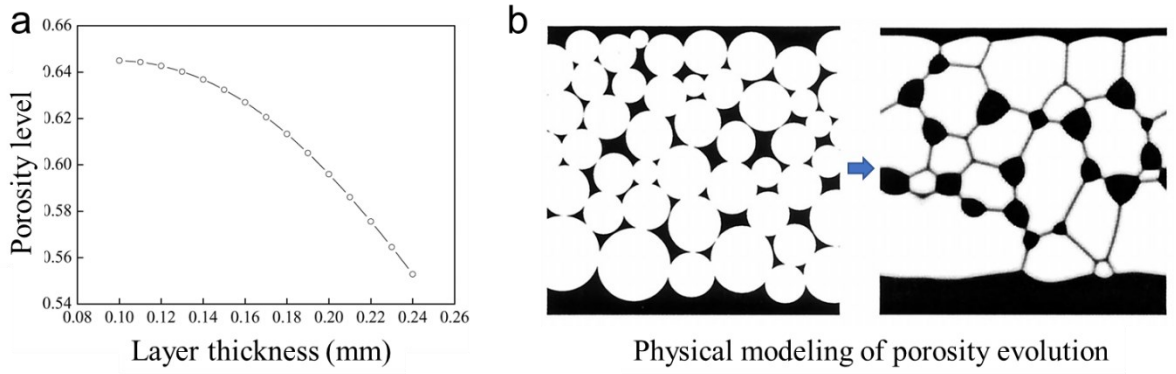

Figure 2.6 (a) Conventional data-driven modeling of porosity (common regression analysis) [1]; (b) Physical modeling of structural evolution of porosity in selective laser sintering (SLS) [2]. A high-level data-driven modeling refers to realistic simulation of porosity evolution, as does a physical model.

Apart from AM, the above two limitations exist in ML applications in some other physical and engineering disciplines. It thus will have a profound, pervasive impact in many domains by breaking those limitations and adequately answering raised questions, which are the purpose of this research.

2.5 Thesis objectives and outline

To explore and unlock the full potential of ML, this research aims to develop a ML-based data-driven AM modeling framework, in which a variety of ML models are used to perform the data-driven modeling and simulation of various representative physical events occurring in three stages of AM (see Fig. 2.7). It would greatly benefit the fast exploration of P-S-P relationship in AM.

The proposed ML-based data-driven AM modeling framework will cover all three stages of AM modeling and focus on a total of five separate data-driven AM modeling tasks, as outlined below:

a) Processing modeling, which includes two modeling tasks of interest - melt pool (area) modeling and 3D temperature field modeling in selective laser melting (SLM);
b) Structure modeling, which contains mesoscale structure modeling (porosity structure development in selective laser sinter);
c) Property modeling, which includes modeling of stress field development and stress-strain curve behavior for lattice structures.

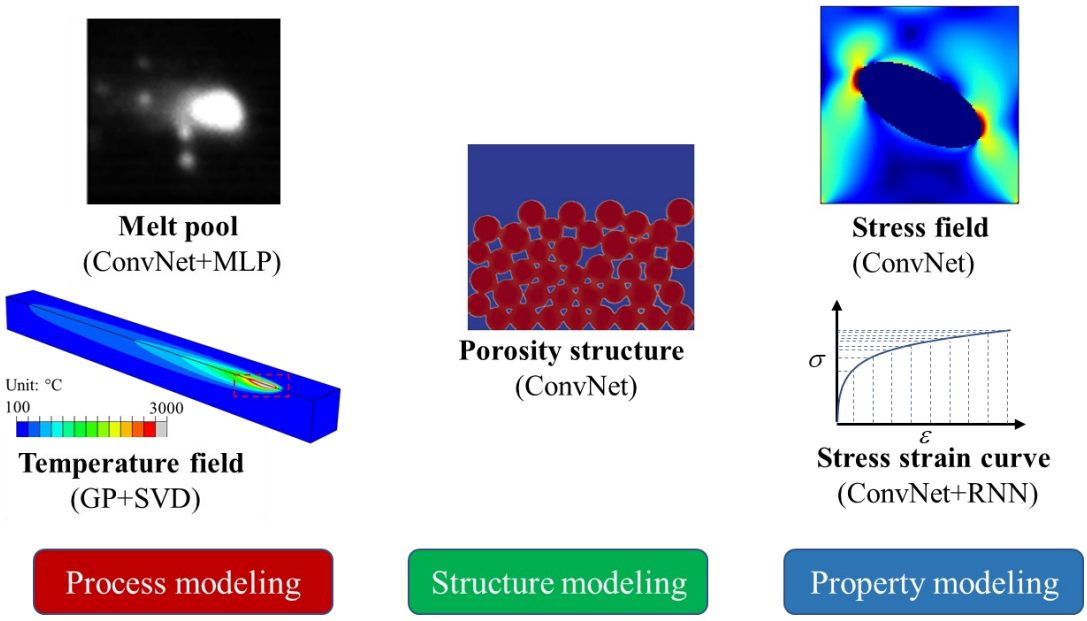

Figure 2.7 The proposed ML-based AM data-driven modeling of various physical events with different quantities of interest. The utilized ML techniques are indicated in the parenthesis: CNN - Convolutional neural network, MLP – Multilayer perceptron, GP – Gaussian process, SVD – Singular value decomposition, RNN – Recurrent neural network.

These modeled physical events involve QoIs featuring distinct properties and characteristics. For example, melt pool size is a simple scalar quantity, temperature field and stress field are high-dimensional quantity with complex pattern or texture, porosity structures are also high-dimensional filed-type quantity but showing simple texture yet strong stochastic characteristics, and stress-strain curve are typical sequence-type quantity. While there are other

physical activities during AM and associated QoIs, those representative quantities studied in this research should well cover different types of quantities possibly encountered in many physical and engineering problems. Thus data-driven modeling of those selected physical events can comprehensively examine and fully explore different ML models for handling various data-driven modeling problems with different flavors. The key idea behind and data-driven methodologies developed can thus broadly shed light on data-driven modeling of a spectrum of physical events in AM and other domains.

The remainder of the dissertation will be organized as follows. Chapter 3, 4 and 5 will present the proposed data-driven modeling of process, structure and property in AM, respectively. Chapter 6 discuss implication of this research on other data-driven modeling in AM and beyond, and present experimental calibration of those data-driven model based solely on simulation data. Finally, Chapter 7 provides conclusions and future work in this field.

Chapter 3 Process modeling

3.1 Data-driven modeling of melt pool

3.1.1 Background

Data-driven melt pool (size) modeling is one of the most widely studied data-driven AM modeling for achieving AM process control. Because the melt pool size, although a simple quantity, is an effective indicator of overall manufacturing quality, intimately associated with the development of columnar grain structure[7], solidification textures at the sub-grain scale [62] and lack-of-fusion porosity [13], etc. A robust data-driven melt pool model is of huge value for optimizing AM process on-the-fly and ensuring part quality. In addition, unlike many other AM quantities, melt pool is readily measurable during AM process by using pyrometer, thermal camera or high-speed camera. This fact facilitates experimental data-based data-driven modeling that can generate actionable insights for AM practice. Therefore, melt pool is a fairly popular QoI in data-driven AM modeling nowadays.

Since melt pool size (e.g., pool width, length, depth or area) is a scalar quantity, data-driven melt pool modeling itself is technically not difficult, falling within the task of common regression analysis. The key challenge and research interest today lie in the rational selection and even design of input features correlated with melt pool development. Most of existing data-driven melt pool modeling [50, 63, 64] just build melt pool dimensions as a simple function of common AM process parameters, typically laser power and scanning speed (Fig. 3.1a-b). Lee et al. [65] have adopted as many as 23 input features, including various process parameters and materials parameters, to

enhance data-driven melt pool modeling. However, those data-driven melt pool modeling by

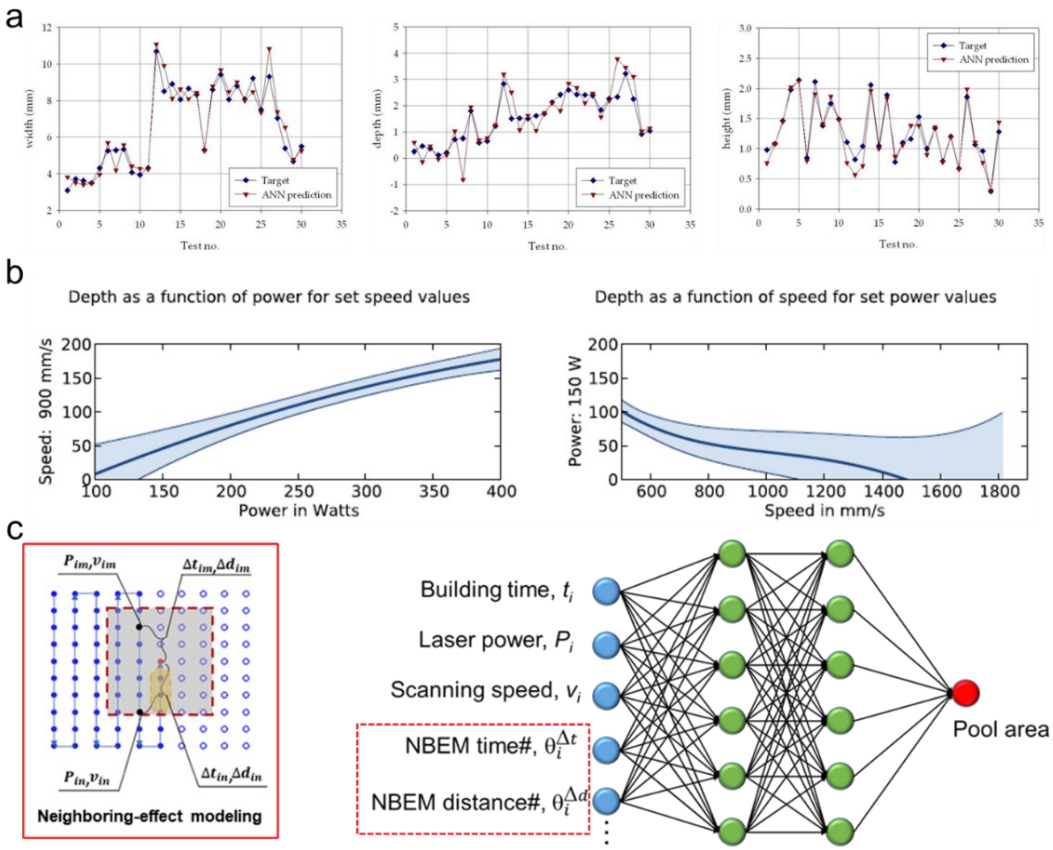

Figure 3.1 A summary of existing data-driven modeling of melt pool in AM

incorporating simple input features, even broadly, cannot adequately account for the complexity of AM process physics, including but not limited to layer-by-layer heat accumulation, inter-track heating effect and, most annoyingly, many aleatory uncertainties (i.e., natural variabilities) present in AM process. Those factors can greatly complicate melt pool dynamics. Regarding this, more advanced input features like cumulative time- and distance-neighboring effect factors [66], have been designed to improve the predictive accuracy (Fig. 3.1c). However, the prior neighboring-effect modeling method (NBEM) still suffers from two drawbacks: (1) ignorance of noise in experimental melt pool image, which can mislead the training of accurate melt pool model; (2)

hand-design of input features, which not only strictly relies on domain expert but also easily lose information of scanning history.

3.1.2 Method

To overcome the above two drawbacks, this research proposes a novel data-driven melt pool predictive model based on machine learning and fueled by massive experimental data from National Institute of Standards and Technology (NIST). As shown in Fig. 3.2, a convolutional neural network (CNN) is first used to pre-process as-received melt pool images, enabling removal of spattering noises and thus extraction of high-quality melt pool data. Then, a melt pool predictive model using multi-layer perceptron (MLP) is trained by incorporating raw, long scanning history as input features, which best accounts for effects (e.g., heating) of printing history on melt pool development.

3.1.2.1 Experiment and raw data

National Institute of Standards and Technology (NIST) has developed Additive Manufacturing Metrology Testbed (AMMT) as a process image acquisition tool to get high-quality experimental data [22][25]. Using AMMT, the melt pool optical signals are maintained within the high-speed camera coaxially and captured following the digital commands (Fig. 3.2a), leading to the generation of numerous melt pool images. The details regarding AMMT acquisition system can be found in [22][24][25].

The raw data include two parts: (1) digital commands and (2) melt pool images. As shown in Fig. 3.2a, the former one contains laser spot coordinates and corresponding laser power and camera ON/OFF (i.e., 2/0 in the command) in sequence. We can visualize the printing path and the fabrication zone based on the commands, as shown in Fig. 3.2b. The printing/scanning path

belongs to the typical zigzag pattern [40]. The red region indicates the fabrication zone where laser power is above zero watts, suggesting a rectangular part. Fig. 14a shows the partial dataset, and the whole dataset for printing one part includes 25051 rows of commands and 1498 captured melt pool images when the camera is ON. Following the experimental procedure in Fig. 3.2, NIST has generated 33 groups of datasets in total, based on AM of 33 sample components. For different groups, different laser powers have been used during fabrication and guaranteed the sufficient variability of data.

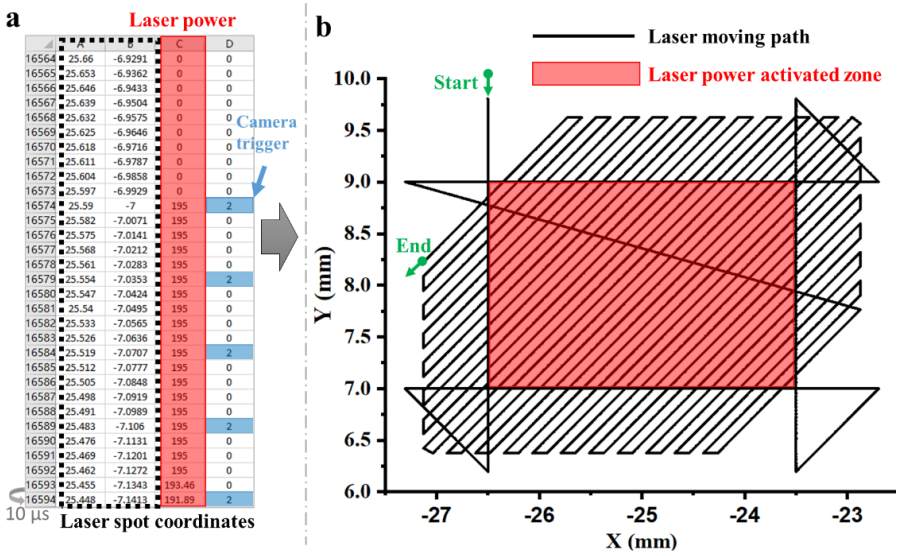

Figure 3.2 (a) Snapshot of AM command file (b) visualization of printing path based on command file.

3.1.2.2 Data denoising by convolutional neural network

Deep convolutional neural network (CNN) is chosen herein among many ML algorithms. As a cornerstone of deep learning, CNN features utilization of multilayer stack of convolutional layers[67] for automatic, hierarchical representation learning; as detailed in Appendix A.2. The four key components of CNN, i.e. local connections, shared weights, pooling and the use of many layers, bring it great advantages in processing data in the form of multiple arrays, such as images (2D) and video or volumetric images (3D) [68].The adopted CNN structure is shown in Fig. 3.3a,

containing two parts: the convolution (green) and deconvolution (blue) parts. The convolution part contains the normalization layers, convolution layers, activation layers with ReLU function, and pooling layers. The deconvolution part includes upsampling layers, deconvolution layers, and output layers. Specifically, this CNN enables complex feature extraction, such as boundaries, textures, and morphologies [38]. A smooth and elliptical-shaped boundary characterizes the real melt pool; by contrast, the noised melt pool presents an irregular boundary due to the interference of spatters as shown. Therefore, CNN will be trained to automatically remove the spatter noise and smooth out the boundary. Prior to performing CNN model training and validating, all the raw melt pool images should go through pre-processing, i.e., image thresholding operation and hue inverting, as indicated by blue dashed arrow in Fig. 3.3a; this processing is used to extract a clear region of interest.

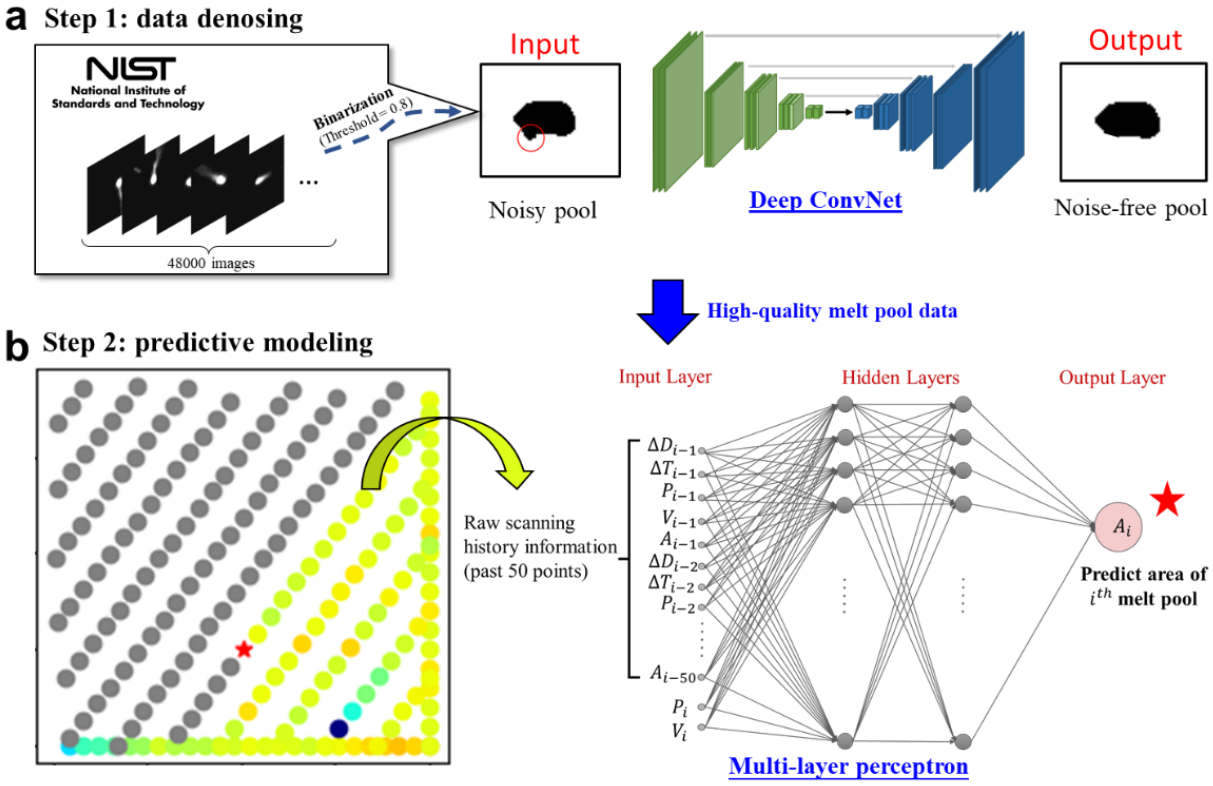

Figure 3.3 Illustration of the proposed framework for data-driven modeling of melt pool.

To successfully train CNN, sufficient input-output pairs are mandatory for CNN to learn feature exaction of real melt pools and removal of the spatter noise. The inputs and outputs are the noised images and corresponding denoised images, respectively. We synthesize 1130 input-output pairs in total, by manually adding spatters to boundary of noise-free melt pools. The synthetic noise is created by mimicking real noise in the NIST data set, including its size, shape, and distributions (i.e., attached or sometimes detached). Data-denoising capability of CNN learned from synthetic noise will be directly transferable to dealing with real noisy images. Noise-free images are then obtained, which generate high-quality melt pool area data for predictive modeling in next step.

3.1.2.3 Predictive modeling by multi-layer perceptron

This research adopts a multi-layer perception (MLP) artificial neural network for melt pool predictive modeling. A detailed introduction to MLP can be found in Appendix A.1. The adopted MLP consists of one input layer, two hidden layers, and one output layer; as shown in Fig. 3.3b. The input layer contains two parts: (1) raw scanning history with respect to the predictive location and (2) laser information about predictive location (i.e., laser conditions used). As indicated by the input layer, the scanning history information involves spatial distance, ΔD, and time distance, ΔT, of preceding points to the predictive point, as well as laser power, P, scanning speed, V, and melt pool area information, A, of previous points. The output layer, i.e., the prediction objective is the melt pool area at predictive location, A_i. By incorporating raw, long scanning history as input features, the proposed predictive model can best accounts for the important inter-track heating and other thermal effects by historical scanning of the current point. The proposed predictive model takes complete advantage of MLP in handling high-dimensional regression problem in conjunction

with large dataset. 6 groups of datasets will be randomly selected for training MLP and the other 27 for testing.

3.1.3 Results

3.1.3.1 Melt pool denoising

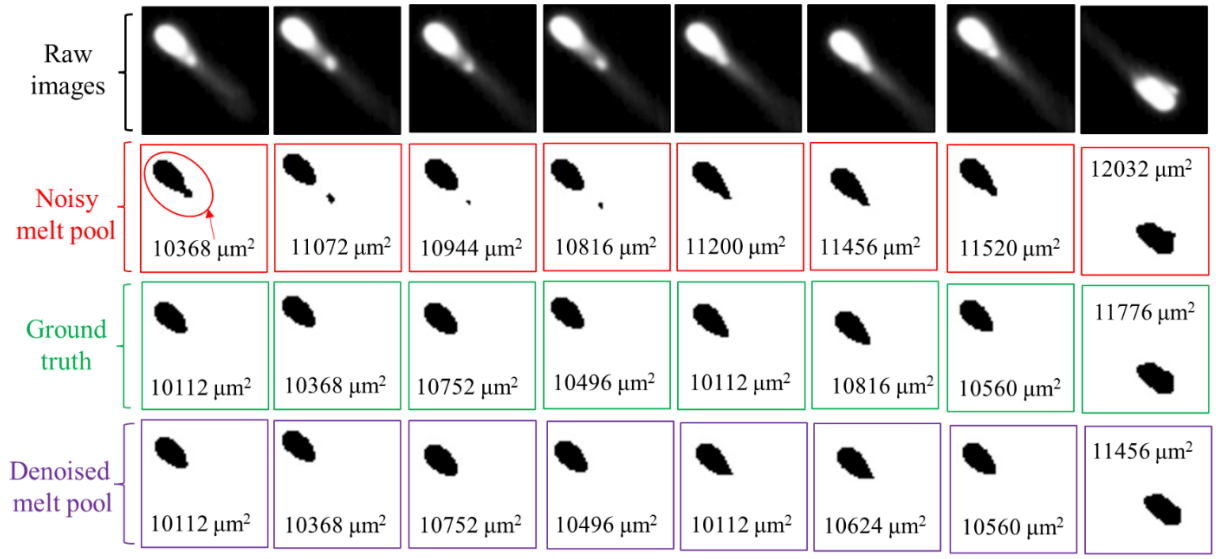

Figure 3.4 Eight testing results to show the effectiveness of CNN in denoising

Fig. 3.4 shows the noisy melt pool image from NIST, denoised image by the trained CNN, and ground truth (manually denoised by human). Area of melt pool is also labeled for quantitative comparison. After denoising processing performed by the CNN, it can be seen that the morphology denoised melt pool closely resembles the ground truth with only a slight difference in few cases. It proves the effectiveness of CNN in eliminating spatter noise. By comparing the calculated area, the CNN result also agrees with the corresponding ground truth. Noteworthy is that the pool area can be reduce up to 15% after the denoising process, especially in cases of detached spattering noise. Minor difference between noisy and noise-free pools would occur when the spatter attaches to the melt pool boundary. However, CNN can still accurately recognize the real melt pool boundaries and remove small spatter noise.

3.1.3.2 Melt pool prediction

Fig. 3.5 compares the predictive performance against the previous NBEM model. The average relative error magnitude (AREM) is widely used to evaluate the prediction performance [22][25][27][48], indicating the average error for all predictions for a testing dataset herein. For rigorous comparison, the NBEM method is implemented using the same MLP model. It can be seen that the proposed model shows an outstanding performance compared with the NBEM model. The AREM value is as low as 2.8 % compared with 14.8 % of the NBEM model. The proposed model reduces the AERM by at least 80% compared with the NBEM model and significantly improves the predictive accuracy. This is largely attributed to the input design, especially the enormous raw information of scanning history. These inputs enable MLP to fully incorporate scanning history as a condition to inform melt pool development at the current location. Also, unlike NBEM and L-NBEM [25], there are no significant AREM fluctuations for all testing datasets, indicative the stability and generalizability of the proposed model.

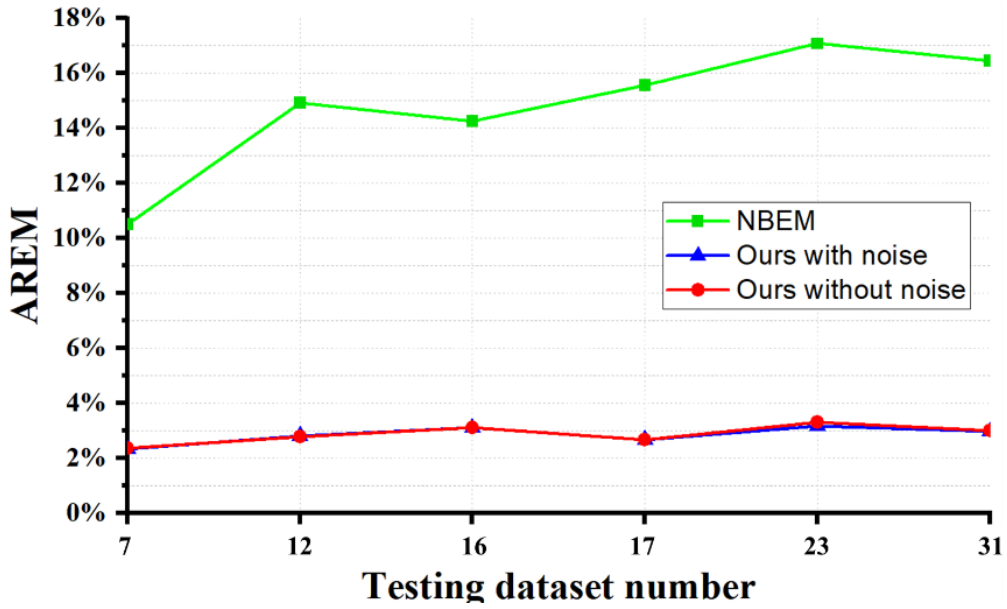

Figure 3.5 Comparison of average relative error magnitude (AREM) in predicting melt pool (area) between NBEM model and our proposed model.

However, it should be admitted that insignificant reduction of AREM is observed after the denoising, as presented in Fig. 3.5. The improvement resulting from data denoising is not apparent herein. This is mainly due to the rather small number and distribution of noise in all datasets. To verify this, we look into the area difference caused by noise. Fig. 3.6 lists the accumulated area difference caused by noise in each dataset. Overall, we can see that the noise causes a relatively small accumulation area difference for 32 datasets, in which only eight datasets have an accumulated area difference ranging from 192 to 5056 μm^2. Even for dataset 1 with the highest accumulation area difference, above 99.1% of raw images are free of noise, as shown in Fig. 3.6b. Overall, the noisy image account for quite a small part in the current case. This is possibly due to the small range of laser power and low energy density used during AM [29][33][49] and the well-designed AMMT system [22][25].

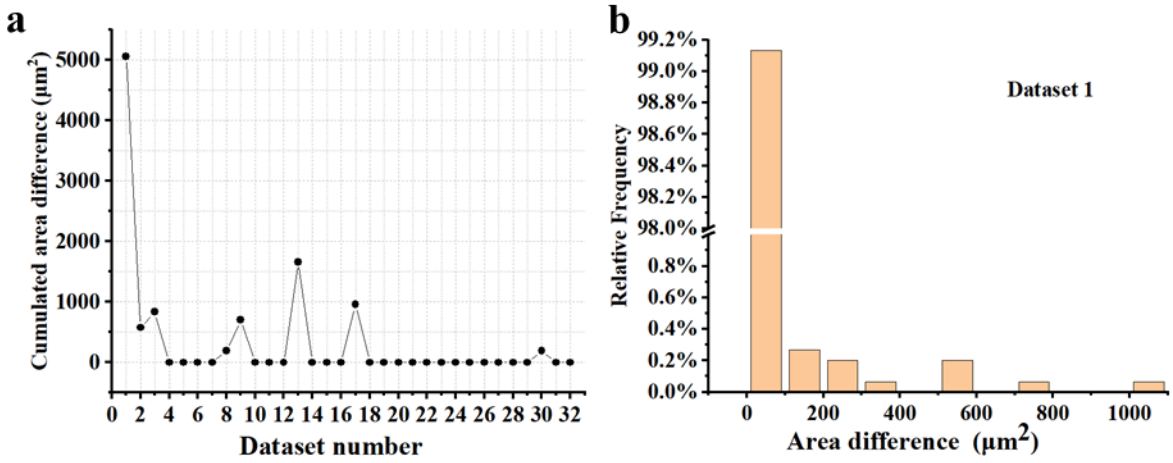

Figure 3.6 (a) cumulated area difference of melt pool between ground truth and noisy melt pool for each dataset, and (b) the relative frequency of melt pool area difference for dataset 1.

3.1.3.3 Noise effect

Despite the fact that little noise exist in current dataset provided by NIST, it is widely observed that the melt pool captured by high-speed camera easily suffers from the interference of spatters and plumes [29][30][31][32]. Therefore, the denoising process is still important in those

cases. To further prove significance of data denoising for improving model performance, we intentionally synthesize melt pool dataset with higher noise level. This allows for comprehensive examination of how the noise deteriorates prediction performance, which in turn demonstrates the importance of the denoising process proposed by this research.

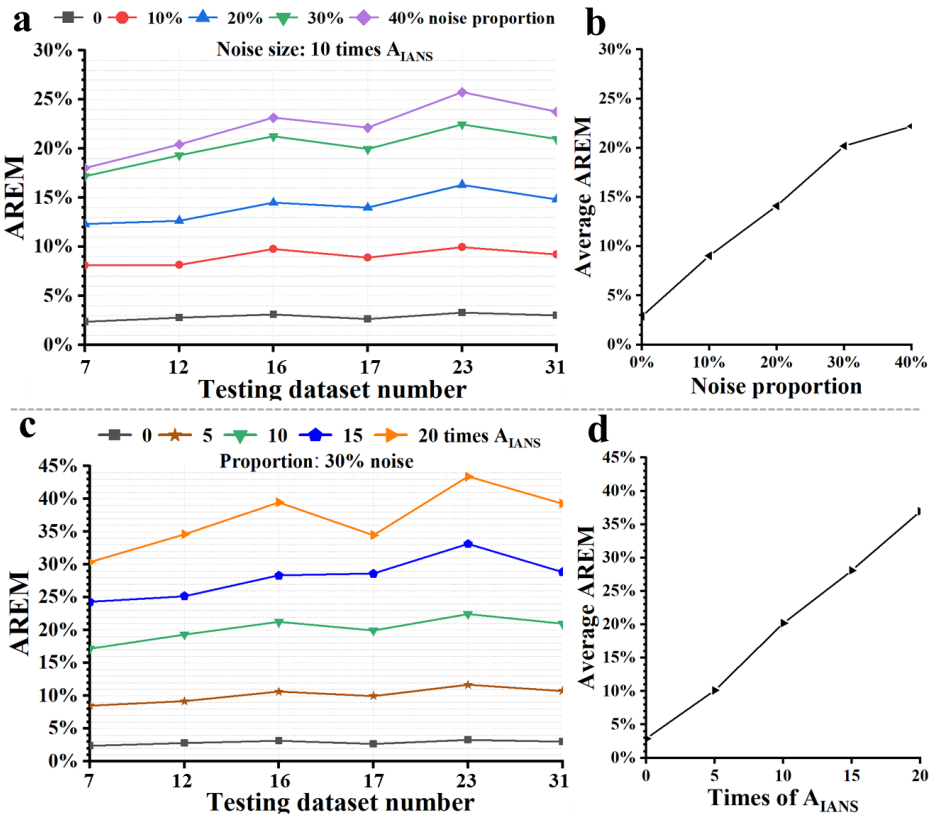

Figure 3.7 Effect of noise on predictive error if without data-denoising.

The synthesizing strategy is based on an even distribution noise: for each dataset, a certain proportion of the raw data (the melt pool area) is made as noisy data by adding a specified noise (noise area). The initial added noise size (A_{IANS}) is 350.89 μm^2, which is 3% of the general melt pool area. Fig. 3.7 presents the AREM variations against noise proportion and noise size. We can see that the increase in noise proportion (0 ~ 40%) with a noise size of 10 times A_{IANS} leads to the substantial rise in the AREM for each testing dataset, as shown in Fig. 10a. Also, the average AREM (Fig. 3.7b) almost linearly increases from 2.88% to 20.195% with 0 ~ 30 % noise

proportion. The slow change is observed by about 2% between 30% ~40% noise proportion. With a fixed proportion of 30 % noise in a dataset, different added noised sizes lead to similar increasing trends as shown in Fig. 3.7c and 3.7d. The average AREM arises from 2.88% to 36.943%, linearly with increasing noise size, i.e., times of original noise size, $A_{IAN.}$ Therefore, a large proportion and size of noise, if contained in melt pool dataset, can significantly affect the predictive performance of the trained ML model.

3.1.4 Discussion

3.1.4.1 Data-driven modeling of 3D melt pool

There is still a lack of capability in predicting 3D melt pools. The AMMT system enables in-situ capturing the 2D surface of the melt pool, however 3D melt pool is still hard to be measured by the existing experimental facilities. A potential way to address the issue is to incorporate the physical model to advance the physical understanding and build a relationship between the 2D melt pool and the 3D counterpart. Following that, integrating the constructed relationship and 2D melt pools is expected to achieve 3D melt pool prediction. The prediction of 3D melt pool can bring new possibilities in further predicting structures and even properties, which are of more practical interests in AM quality evaluation. For example, with the melt pool width and depth information fed from the 3D melt pool predictive model, the probabilistic porosity model [10] can be adopted to predict the lack-of-fusion porosity level within as-built AM components.

3.1.4.2 Real-time melt pool control

The ultimate goal of data-driven melt pool modeling is to realize real-time online closed-loop control of L-PBF processes [22], and achieve smart manufacturing for the AM process. However, it is still quite challenging to accomplish the feedback control in milliseconds for each

laser spot. To reduce the time-consuming iterative calculation for changed laser power [22], an advanced offline design of laser power plays a crucial role. To achieve constant melt pool, the currently developed prediction model can be used to inversely design the AM control factors (e.g., laser power and scanning speed) based on the continuous scanning path coordinates. In that case, the designed printing commands can be directly used for printing a constant melt pool and thus achieve an offline process planning. However, due to the great variability in practical AM [4], more complex control by taking into account uncertainty is required through robust design under uncertainty [69].

3.2 Data-driven modeling of temperature field

3.2.1 Background

Data-driven thermal modeling is a more advanced data-driven process modeling. It gives thermal details of AM process that includes melt pool informed by high-temperature region. Temperature field formed during AM process is closely related to later development of various structures across many scales and, in turn, largely dictate the properties of final AM parts. Thermal modeling is thus of fundamental importance as a starting point of exploring P-S-P relationship, as evidenced by many multi-stage AM studies of dendrite structure [70, 71], grain structure [72, 73], porosity [74] and mechanical property [75].

However, unlike data-driven modeling of melt pool, data-driven modeling of temperature field is challenging due to its high-dimensional and complex nature. In light of this difficulty, there have been much simplified data-driven thermal modeling for AM process. A widely adopted strategy is data-driven thermal modeling of local temperature, instead of the entire temperature field. There exists two such types of data-driven thermal models. The first type focus on a specific, local thermal quantity like peak temperature within the current heating zone (Fig. 3.8a) [76]. The

second type is general to all spatial points; that is, the modeling objective is the real-time temperature or thermal history for any given point within component [51, 77, 78] (Fig. 3.8b-c). In this way temperature field can be implicitly obtained by aggregating separate modeling results of many points [78]. However, there is still a lack of data-driven modeling effort that directly predict the 3D temperature as a whole.

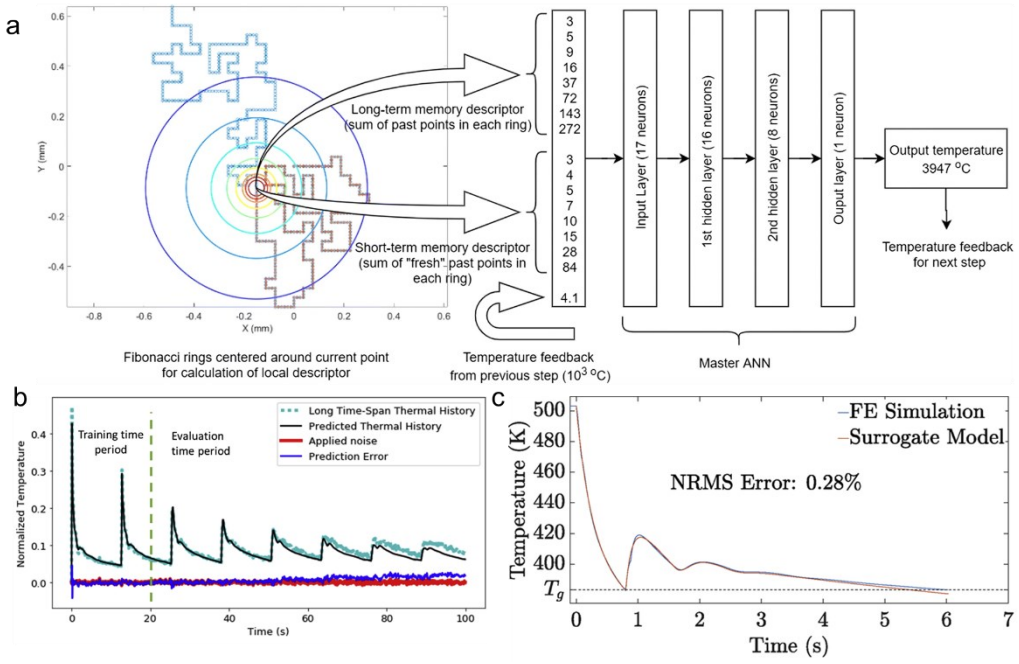

Figure 3.8 A summary of existing data-driven thermal modeling in AM

3.2.2 Method

Instead of modeling the node-/voxel-level temperature, this research aims to predict entire 3D temperature field developed under different AM conditions. This is done by first utilizing singular value decomposition (SVD) [79] for dimension reduction of temperature field data, followed by modeling the relationship between temperature field and AM conditions by using Gaussian process (GP) modeling in the latent space. More details about GP regression can be found in Appendix A.3. Note that, this research focuses on predicting the stable temperature field achieved under various conditions, instead of real-time temperature field evolution.

Fig. 3.9 presents the overall flowchart of the data-driven modeling using the SVD-based GP modeling method. Define the steady temperature field for given AM condition $\mathbf{d}$ as $T(\mathbf{d}, \mathbf{s})$, where $\mathbf{s} \in \Omega_{xyz}$ stands for all the spatial coordinates of the nodes. In this study, AM conditions include laser power, scanning speed, power absorption efficiency, and thermal properties of printing materials. We first generate N_t training points for $\mathbf{d}$ using Latin Hypercube sampling approach [80]. We then perform thermal simulation for each of the training points and obtain the steady temperature field developed under different conditions, $T(\mathbf{d}^{(i)}, \mathbf{s}), i = 1, 2, \cdots, N_t$. After that, we approximate the original simulation data matrix using SVD as follows [81]

$$T(\mathbf{d}^{(i)}, \mathbf{s}) \approx \sum_{j=1}^{m} \gamma_j(i) \boldsymbol{\eta}_j(\mathbf{s}), \forall i = 1, 2, \cdots, N_t \tag{3.1}$$

where $\gamma_j(i), i = 1, 2, \cdots, N_t; j = 1, 2, \cdots, m;$ are the responses in the latent space, m is the number of important features used in SVD, and $\boldsymbol{\eta}_j(\mathbf{s}), j = 1, 2, \cdots, m$ are the important features.

Using the training points of $\mathbf{d}$ and the corresponding latent space response, we then build GP surrogate models in the latent space as

$$\gamma_j = \hat{g}_j(\mathbf{d}), j = 1, 2, \cdots, m \tag{3.2}$$

For any new prediction point $\mathbf{d}$, we have $\gamma_j(i) \approx \hat{g}_j(\mathbf{d}^{(i)}, \boldsymbol{\lambda}^{(i)})$ and thus the surrogate models are used to predict the steady temperature field response at new prediction point as below

$$\hat{T}(\mathbf{d}, \mathbf{s}) \approx \sum_{j=1}^{m} \mu_{\hat{g}_j}(\mathbf{d}) \boldsymbol{\eta}_j(\mathbf{s}), \tag{3.3}$$

where $\mu_{\hat{g}_j}(\mathbf{d})$ is the mean prediction of the j-th latent space surrogate model and $\hat{T}(\mathbf{d}, \mathbf{s})$ is the predicted temperature field response for given $\mathbf{d}$.

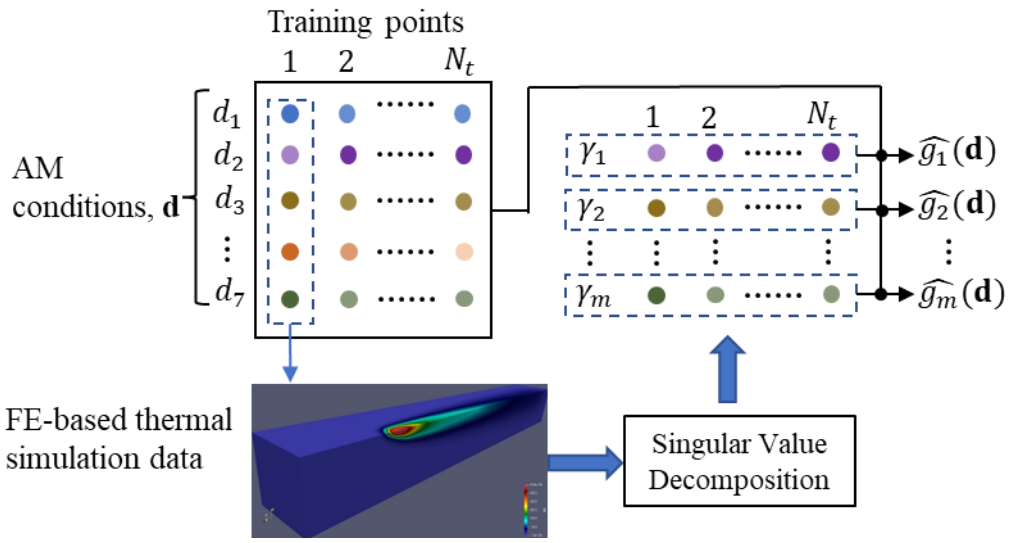

Figure 3.9 Flowchart of data-driven modeling of temperature field.

To train GP model, we have run 280 thermal simulations at generated training points (each training point or case corresponding to a specific combination of AM conditions. Of them the last 20 temperature profile data are set aside for testing; the remainder are training dataset.

3.2.3 Results

3.2.3.1 Temperature field prediction

Figure 3.10 shows partial testing results of the GP-based thermal surrogate model. High-temperature fields resembling teardrops of different size are developed under different conditions. By comparing Fig. 3.10a and b, we find that the thermal surrogate model can predict similar temperature field development as the physical model under various conditions. However, running a single FE simulation usually requires about 50 minutes (based on Xeon CPU E5-2660), while a surrogate model just takes a couple of seconds for each prediction. Fig. 3.10c shows that relatively large prediction errors mainly occur near the electron beam focus. This might be attributed to great sensitivity of thermal response and thus great variability of temperature field in EB focus region.

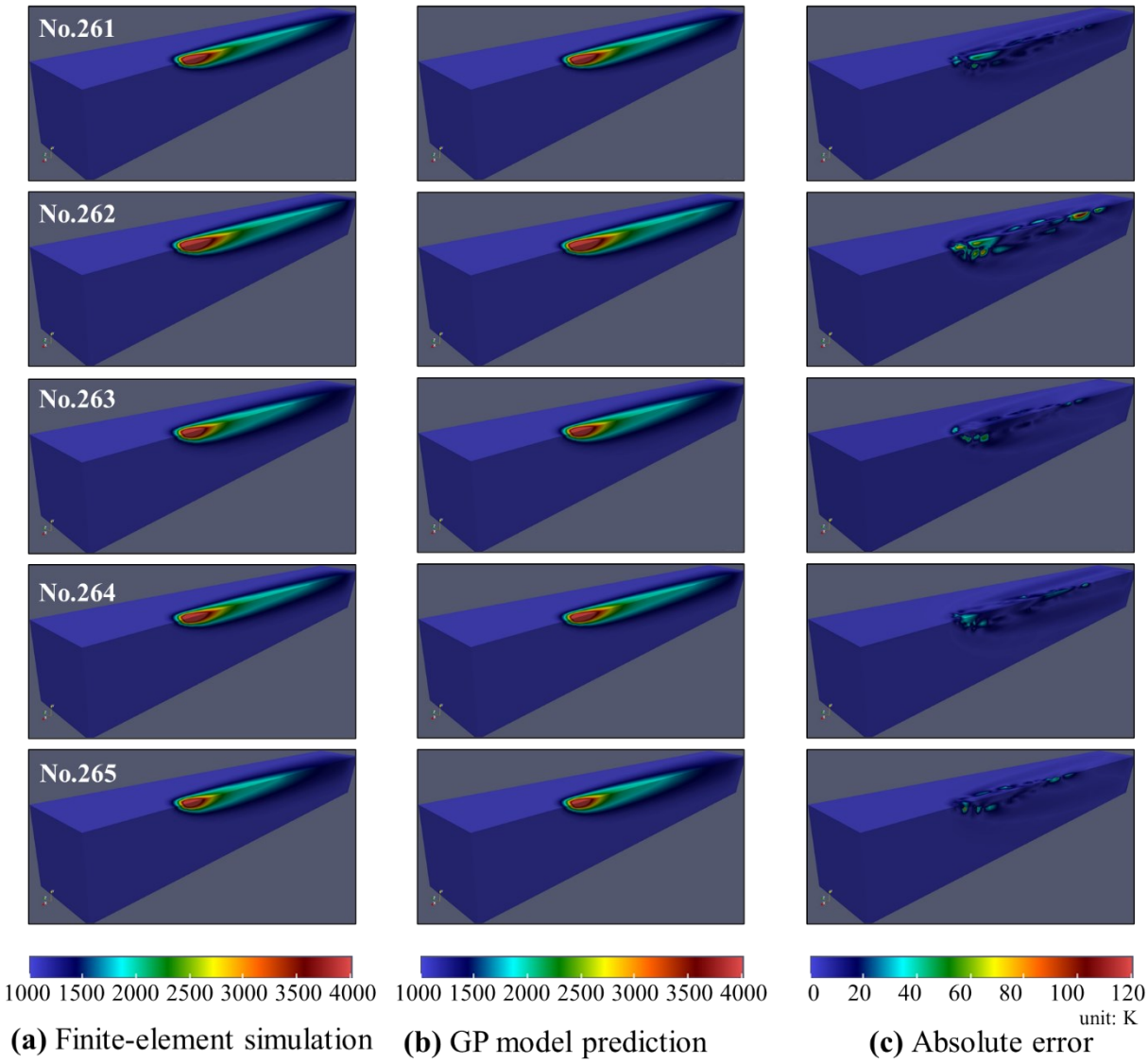

Figure 3.10 Prediction of the steady temperature field developed under different testing AM conditions.

3.2.4 Discussion

3.2.4.1 Application for accelerated structure prediction

As mentioned, obtaining the temperature field is often the foundation of ensuing structure prediction in AM. The trained GP-based thermal surrogate model can quickly provide temperature field information for accelerated structure modeling. Fig. 3.11 illustrates one potential application of the trained thermal model for grain structure modeling [82, 83] in SLM. It is known that grain

structure development is greatly controlled by the living temperature field. For example, grain nucleation at the front of solidification is dictated by cooling rate, and grain growth direction is usually along the local thermal gradient. The predicted temperature field can be thus fed to microstructure simulation model and inform the evolution of grain structure during AM process.

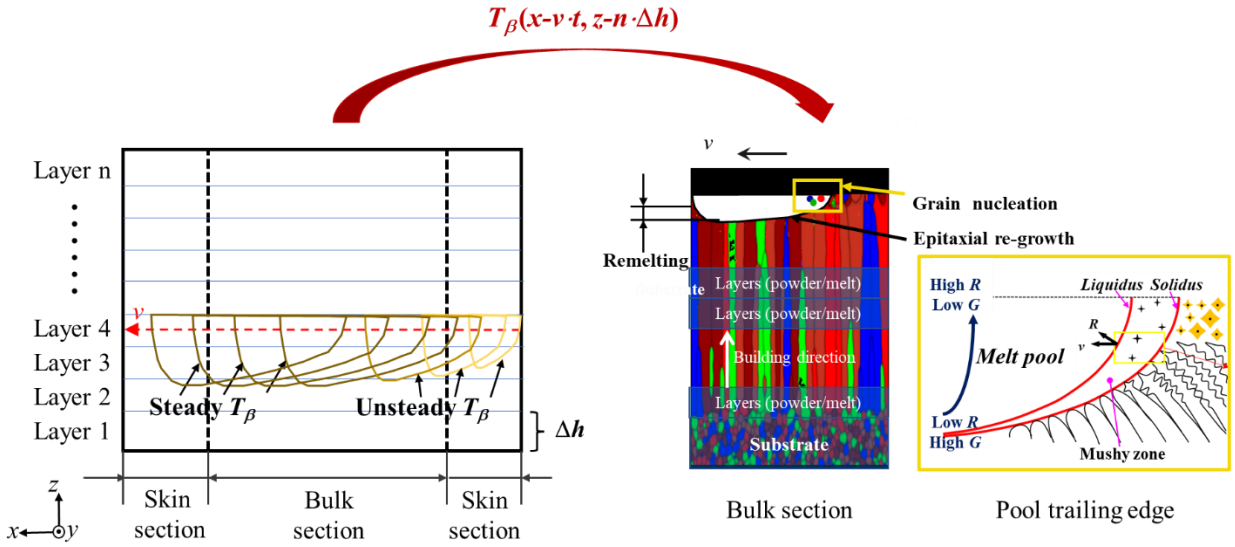

Figure 3.11 Illustration of incorporating the trained GP thermal model for accelerated simulation of grain growth in AM.

Previous researches often depend on the analytical model for fast obtainment of temperature field under different AM conditions. Of them, the most known analytical thermal model is Rosenthal model, which was firstly developed for analytical thermal modeling in fusion welding [34]. Rosenthal [35] has further adopted the Rosenthal equation to predict the quasi steady-state temperature distribution, followed by wide extension to temperature field prediction in AM [84-86]. However, the Rosenthal model is based on multiple assumptions [87]: 1) the thermal properties are independent of temperature; 2) temperature predicted at the origin of the beam is infinite; 3) the maximum temperature does not exceed the liquidus temperatures [88]. On the contrary, the trained GP model can well reproduce the high-fidelity results by the physical simulation and give more reliable temperature field prediction. It thus can serve as an excellent

substitute of the Rosenthal model to speed-up prediction of various thermally-controlled structure development in AM.

Chapter 4 Structure modeling

4.1 Data-driven modeling of porosity evolution

4.1.1 Background

Porosity is one of the most important structural features of AM-fabricated components. Depending on AM process conditions, various types of porosity can be developed for as-built AM parts, such as lack-of-fusion by insufficient melting [10], surface open pore by complex fluid dynamics [12] and keyhole by strong metal evaporation [14], etc. Porosity modeling might be particularly valuable for the selective laser sintering (SLS), a type of AM process featuring sintering mechanisms for binding raw materials. Unlike laser melting, sintering-based binding is a mild process with insignificant melting & solidification phenomenon, which makes porous

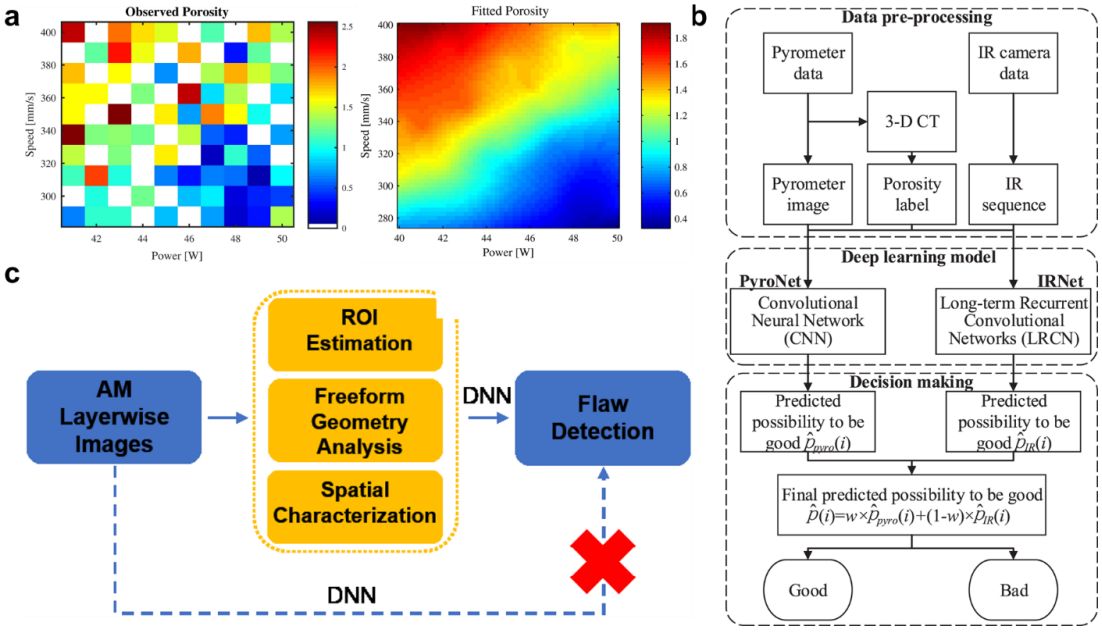

Figure 4.1 A summary of existing data-driven modeling of porosity in AM

structure the prominent feature of SLS components [89]. A data-driven modeling capability of porosity development is urgently necessitated to facilitate process plan and porosity control in SLS process.

Data-driven porosity modeling has been widely studied in AM community. The majority of them concentrate on correlating porosity represented in scalar porosity-related metrics, typically density [1, 90, 91] or relative density [92, 93], with different AM process parameters (Fig. 4.1a). In addition to such common data-driven porosity modeling, there are a few other interesting attempts. Tian et al. [16] have selected some special and complex input features, including pyrometry images and sequential images from infrared camera during AM process, to data-driven predict layer-wise porosity (Fig. 4.1b) and Imani et al. [94] have taken layer-wise surface images from DSLR camera as inputs (Fig. 4.1c). Therefore, existing researches often focus on scalar porosity metrics and rarely use ML for data-driven modeling of morphological evolution and development of porous structures.

4.1.2 Method

To overcome the limitation of existing researches, this study proposes to use CNN to simulate the porosity evolution and development in selective laser sintering (SLS). To account for the laser conditions on porosity development, a novel CNN with multi-input, yNet, is especially proposed for solving this data-driven modeling problem [95]. It can be used for data-driven modeling of condition-dependent field evolvement, e.g., AM condition-controlled porosity evolution herein. Fig. 4.2 illustrates the designed architecture of yNet, which is basically composed of the main encoder-decoder based on deep convolutional network and a branch of multilayer perceptron (MLP). The encoder plays a role as feature extractor, which distills the original high-dimensional field into numbers of information-rich and reduced-size feature maps. MLP acts to

expand the condition parameters into numbers of neurons that generate a high-dimensional embedding vector. The encoder and MLP can greatly alleviate weight mismatch of original inputs, thereby facilitating effective fusion of input signals at their ends. Decoder serves to correctly reconstruct the merged signal back to a meaningful and desired field through the deep deconvolution process. Note that, in the deep deconvolution process we also concatenate feature maps extracted during the early autoencoding process; see gray arrows in Fig. 4.2c. This technique is widely adopted in semantic segmentation networks, such as U-Net [96] and FCN [97], with the aim of improving segmentation details. In the case of field evolvement, this technique is expected to endow yNet the capability of easily capturing tiny or even no field evolvement in partial region. For example, there is no porosity evolution/change at vertical locations far from heat source. It thus requires yNet to efficiently reconstruct feature maps to original porosity structure therein.

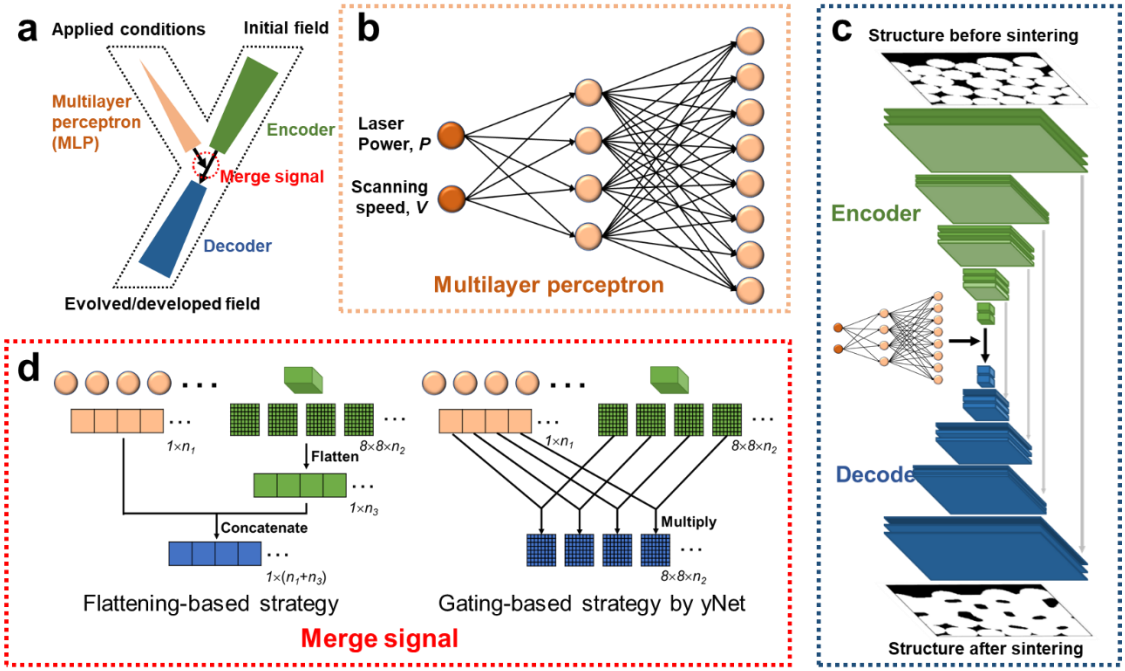

Figure 4.2 The proposed yNet for data-driven modeling of laser condition-dependent porosity development in selective laser sintering.

More specifically, for yNet instantiated in this study, the first layer of MLP is two condition inputs (e.g., laser power and scanning speed in data-driven porosity modeling in SLS process), the

hidden layer is a fully connected layer with 128 neurons and the final layer 256 neurons. For the encoder, each green block represents a combo of Conv+Batch-Normalization[98]+Relu[99] operations and light green block means max-pooling. In this manner, encoder finally yields 256× 8×8 feature maps, which are passed to a dropout layer with rate of 0.5 before merging with MLP signal. The decoder just has a somewhat mirrored topology of encoder, with each blue block representing a combo of Conv+Batch-Normalization+Relu operations and dark blue block up-sampling.

After building up the main architecture, the core problem is the proper manner in which two streams of input signal are merged. In fact, such multiple mixed inputs (a high-dimensional image and scalar parameters) are not uncommon in various ML tasks, although the output is not necessarily image. However, the merging strategies in existing multi-input CNNs can suffer from various drawbacks associated with dense parameters. A common merging strategy [100-102] is to flatten image signal, here represented by its feature maps, into a long vector, thus making it compatible and hence concatenable to the vectorial output of MLP; see Fig. 4.2d. The flattening-based strategy might be effective for developing a simple label or continuous predictor, but will deteriorate the trainability and performance of yNet whose output is image/field. Because consistent signal flow of meaningful feature maps is desired throughout the encoding-decoding process, whereas the flattening operation in those strategies will interrupt the signal flow of feature maps. The merged signal in those manners becomes somewhat meaningless or less interpretable. A flattening-based strategy thus, if workable for image-to-image regression [60], might make yNet more like a brute-force regressor with poor interpretability and low trainability. What's more, flattening feature maps yields a large fully connected layer, which are notoriously parameter-intensive and can degrade the network performance (e.g. DeconvNet [103] versus SegNet [104]).

Regarding the above facts, we propose to properly merge the signal using a one-to-one connection via multiplication inspired by the gating mechanism [105], thus without flattening feature maps. In this way, MLP actually turns as a signal modulator, which takes charge of signal strength of each feature map passed to decoder. The merged signal in this fashion is meaningful, still representing feature maps albeit with changed signal strength. Besides its full interpretability, the effectiveness of the gating-based merging strategy also lies in the fact of those neurons (or basically the applied conditions) interacting with respective feature map and thus the initial field behind in a direct, neat and therefore strong manner. We posit that, through training, MLP can learn to precisely manipulate the initial field represented by high-level feature maps into the developed field (preliminarily in the form of gated feature maps) for any given condition; the decoder then appropriately reconstruct the gated feature maps into the realistic developed field.

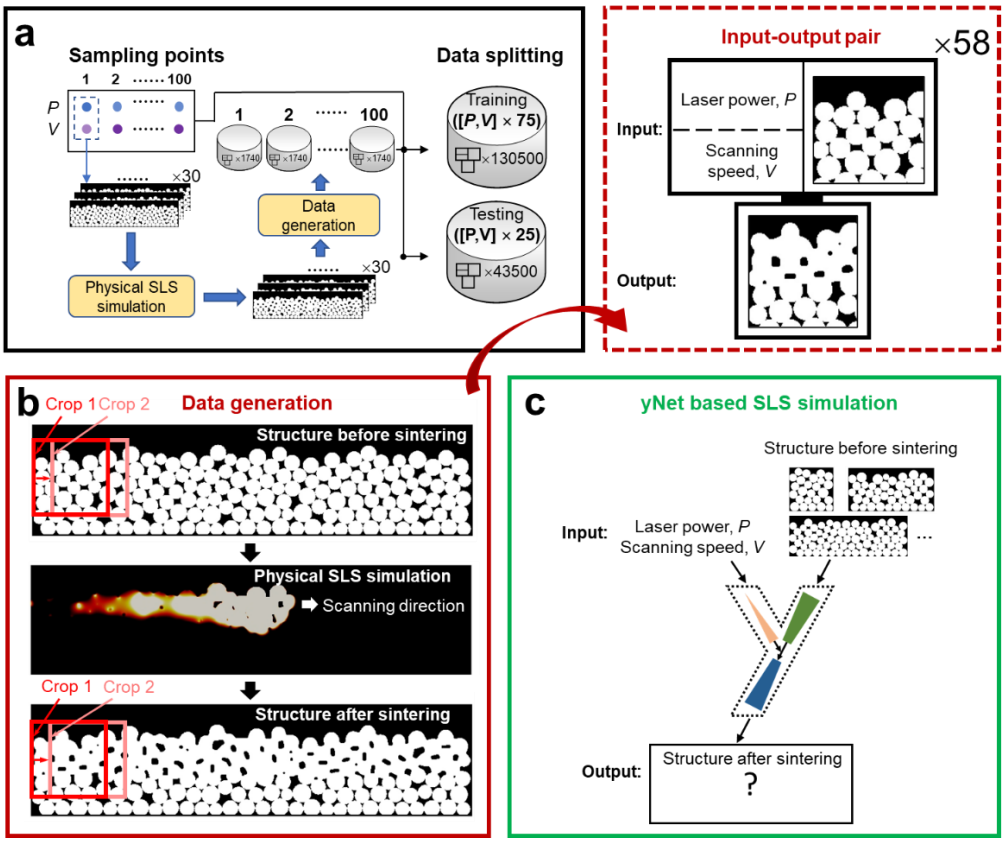

Figure 4.3 The overall procedure for training and testing yNet for porosity evolution simulation in SLS.

To train yNet, we first generate 100 sampling points of [P, V] using Latin Hypercube sampling method [80], and then perform physical SLS simulation of 30 1400 mm-long single-layer of powder beds randomly generated at each point; see Fig. 4.3a. We repeatedly simulate 30 1400mm-long tracks, simply because it is computationally much easier than simulating a single 42000mm track while producing the same amount of data. A natural problem would then arise since we often deal with field of variable and large size in a scientific problem, as is the current case. However, CNN during training favors standard patches of predefined dimensions. We thus adopt the patch-wise data generation scheme as illustrated in Fig. 4.3b, to convert our raw simulation results into data more amenable to yNet. Consequently, we get a dataset of 174000 input-output pairs in total. Each data point consists of four elements - laser power, scanning speed, structures before and after sintering. Therefore, we train a yNet to directly predict the final porosity development from raw powder structure (i.e, one-time prediction). The generated dataset is split into training and testing parts based on laser power and scanning speed sample points, i.e., 75 [P, V] conditions randomly selected for training and 25 for testing; see detailed partition of [P, V] points in Appendix C.1. As demonstrated later, during yNet-based SLS simulation, the trained yNet based on small patches is well scalable to long tracks, i.e., directly taking long track as input after training as shown in Fig. 4.3c.

4.1.3 Results

4.1.3.1 Testing on small standard patches

Fig. 4.4 shows 5 testing results on small standard patches. Note that, a different colormap is used to better visualize the absolute error. It can be seen that yNet results resemble closely to physics-based simulations and only small morphological error exists. As expected, predictive errors occur within the heat affected zone (HAZ), where the porosity evolution takes places. In

addition to visual resemblance in Fig. 4.4, we examine the pixel-wise global accuracy (i.e., percentage of correct pixels) for such two-phase high-contrast fields. The yNet achieves an as high as 99.13% similarity to ground truth in average for all 42750 testing results.

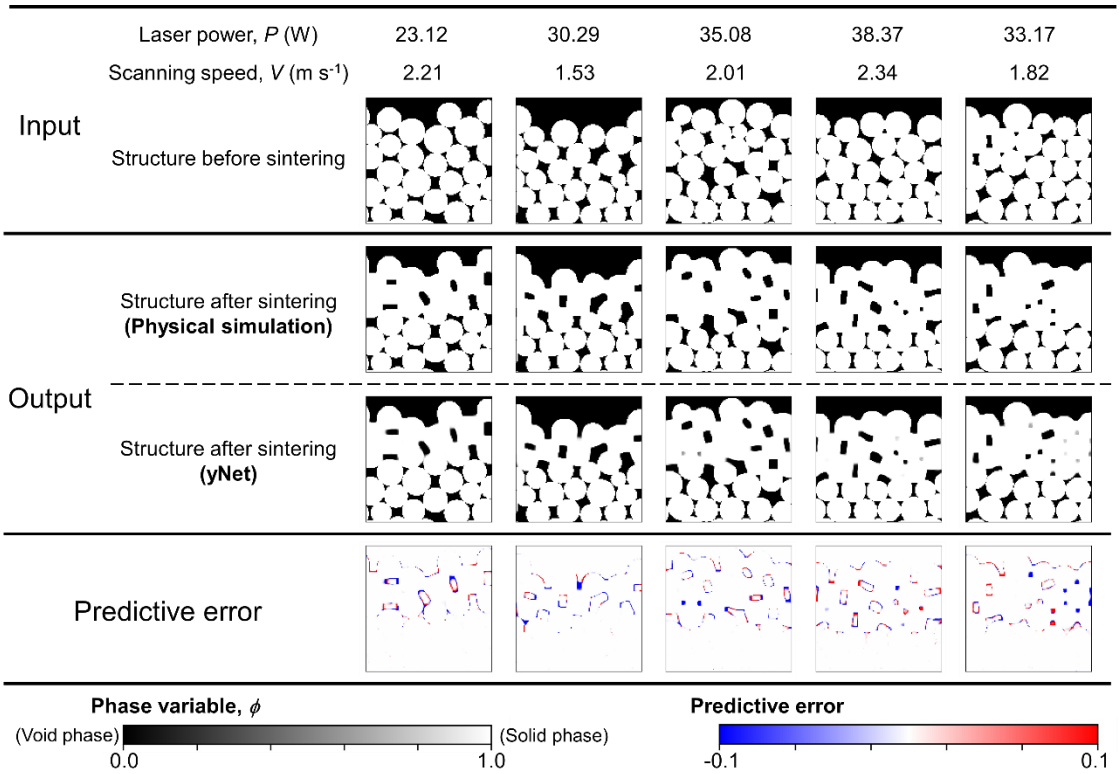

Figure 4.4 Testing of the trained yNet on modeling domain of small standard patches. We randomly selected 5 out of 42750 testing results.

4.1.3.2 Testing on long tracks and spatial extrapolation

The as-trained yNet is scalable to perform large- or even extreme-scale simulation in space, such as multiple layers of long tracks. To validate step-by-step, we first examine the spatial extrapolation of the trained yNet to handling a single-layer long track; see Fig. 4.5a. Fig. 4.5b compares the accuracy of yNet in modeling small patches of original size and long tracks. For different laser power conditions, yNet consistently shows a comparably high accuracy in dealing with long tracks. On average, yNet achieves a pixel-wise accuracy of 99.04% in modeling long tracks, which is close to 99.24% in modeling small patches. The root reason for its scalability to

track of any length is that yNet essentially learns sintering effect associated with laser conditions and independent from track length. The second step is the further validation of sintering multi-layer long tracks that involves inter-layer interaction. By following the workflow in Appendix C.2, we perform sintering simulation of multiple layers under three laser power conditions (P = 20, 25, and 30 W). The overall sintering effect of multiple layers (measured by fraction of unsintered region) as a function of laser power are then obtained for yNet and physics-based model, as shown in Fig. 4.5c. Note that, for unbiased measurement of unsintered regions, unsintered regions are manually labeled and calculated by the third party. We can observe that, by increasing laser power, results by yNet and physics-based model both show a transition from layer-scale unsintered region (see top-left inset) to small scattered unsintered areas (see bottom-left inset) and, eventually, to a

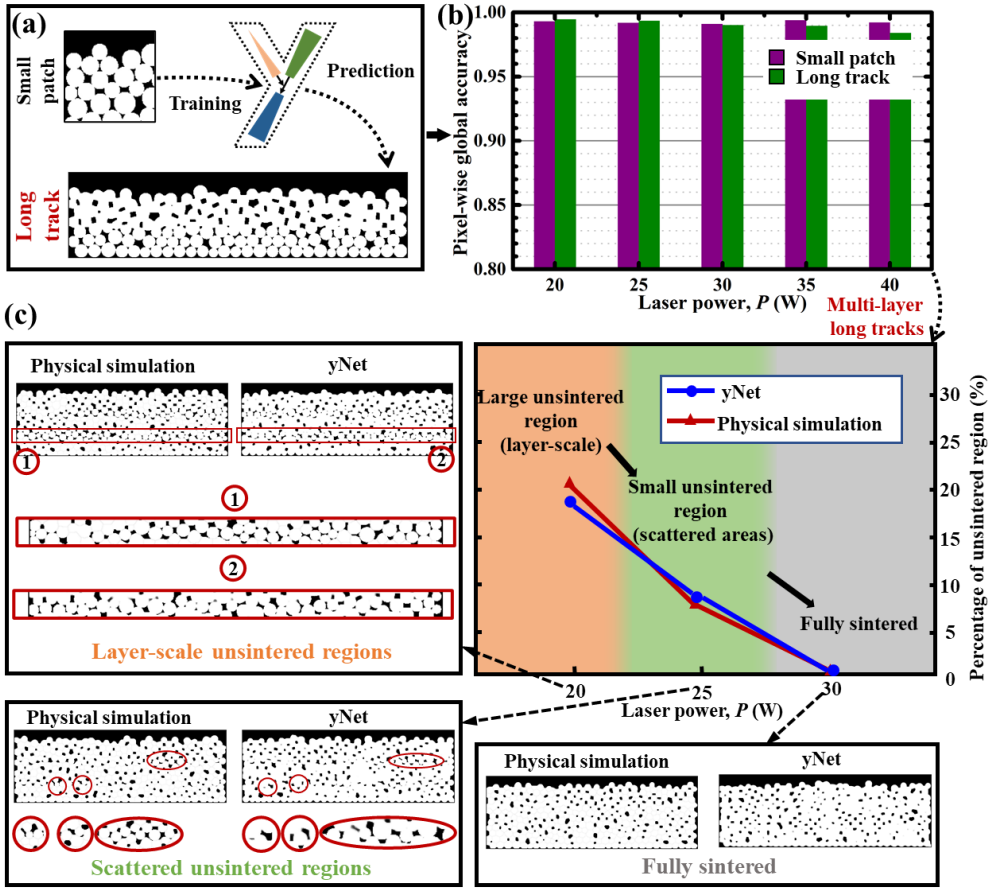

Figure 4.5 Testing of the as-trained yNet on extrapolating prediction for single-layer and multi-layer long tracks.

fully sintered state (see bottom-right inset). The unsintered regions are located in between layers, due to insufficient sintering depth under small laser powers. The quantitative results of measured unsintered regions (see top-right figure) further prove the good agreement between yNet-based and physical multi-layer simulations.

The above-validated spatial scalability can help address the computationally daunting task of full-component SLS simulation. As an illustration, we use yNet to perform multi-layer sintering

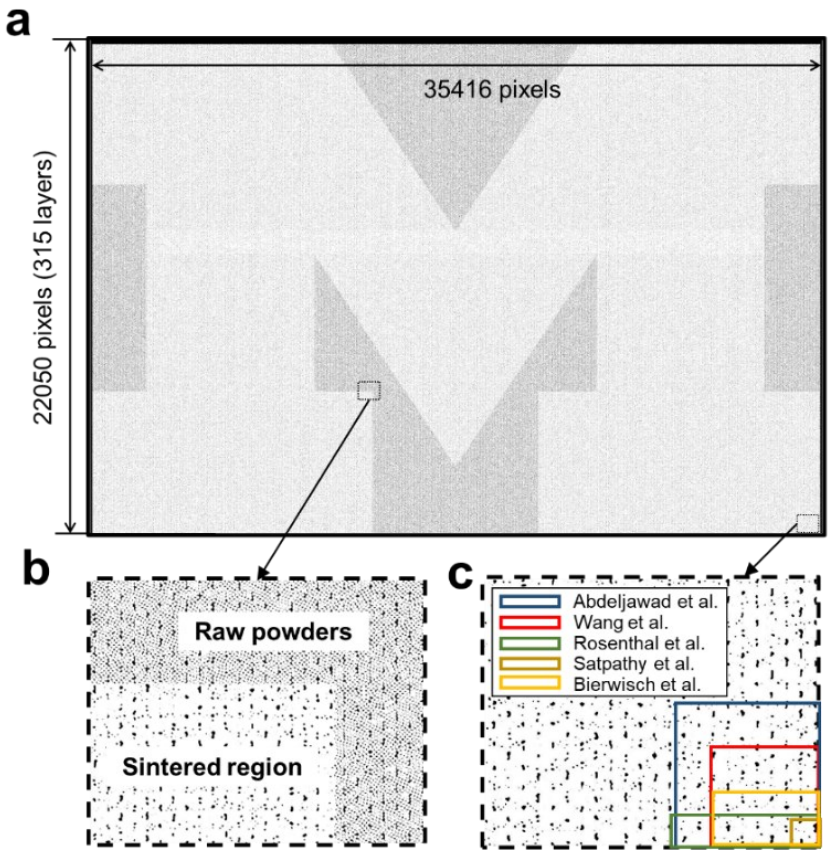

Figure 4.6 Machine learning enabled component-level sintering simulation. The final result is yNet-based porosity simulation of a 315-layer high SLS component - block "M" logo of University of Michigan (35416×22050 pixels).

simulation of a practical 315-layer SLS component with dimensions of 70.8×44.1 mm^2 (or 35416×22050 pixels). As depicted in Appendix C.3, we completely emulate the practical SLS process to perform the full-component simulation layer-by-layer, i.e., generating a new layer of raw powder bed, CAD-model guided selective sintering simulation of the newly added layer, and

repeat the above two steps until completing the whole component. It should be pointed out that computational cost of raw powder bed generation increases exponentially with track length. For efficiency, we simplify powder bed generation during full-component simulation, by depositing a continuous series of mini-powder-beds till completing deposition of the entire layer. This will numerically create large pores regularly observed in between adjunct mini-powder-beds, as shown later. Fig. 4.6a shows the final SLS simulation result. Due to its extremely large size, here we can only present a fraction of the detail of the whole simulation result; see Fig. 4.6b. The periodically observed large pores are attributed to numerical artifacts due to the aforementioned simplified powder-bed generation. Once available, a more efficient powder bed generation model can be integrated with our yNet for better component-level simulation. It is known that AM simulation is often limited to a single layer or even a representative volume element (RVE). A practical simulation of the entire layer-by-layer AM process is rarely achieved so far. Fig. 4.6a represents one of the first achievements of component-level AM simulation to date. Fig. 4.6c summarizes the simulation size of some of existing 2D sintering simulation researches [2, 106-109]. Note that they are compared in terms of dimensionless size in number of pixels. Although prior researches did not report computation time and hardware used, Fig. 4.6c gives a rough yet direct insight into the massively boosted simulation capability by yNet. The simulation size by yNet is several orders of magnitude larger than previous physics-based simulation. yNet enabled component-level simulation would open up enormous possibilities in SLS practices. For example, SLS has been intentionally used for fabricating porous structures with high permeability and bio-compatibility [110, 111]. The trained yNet permits extensive virtual experiments in the pre-production phase, thus greatly facilitating structure design and process planning for fabricating a SLS component with any desired porosity distribution.

4.1.3.3 Comparison of computational efficiency with physics-based model

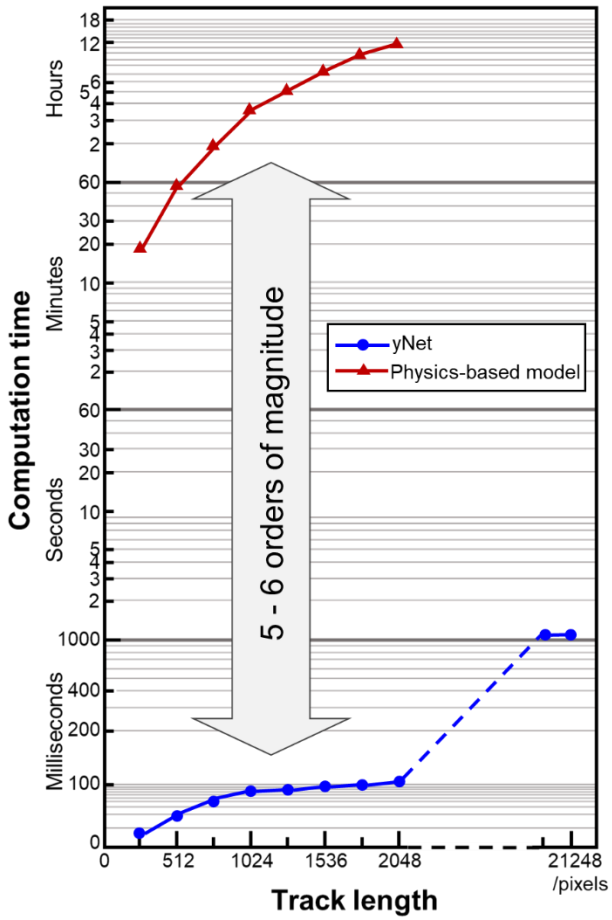

Figure 4.7 Comparison of computation time between yNet and physics-based model.

As discussed early, physics-based SLS simulation especially at the component level is quite computationally demanding. It thus presents as an ideal case to demonstrate the computational superiority of yNet over physics-based model. In this subsection, we look into the computation time of yNet and the original physics-based SLS model. Again, yNet and physics-based model are intentionally tested on the laptop used in the benchmark study in Application 1. Fig. 4.7 shows the computation time as a function of the length of modeled powder bed. We can find that simulation of small tracks like 640-pixel long tracks in Fig. 4.5a is just a matter of tens of milliseconds by using yNet. In striking contrast, the original physics-based model used in this study will take a few hours to complete the same task in the same computational setting. The test for physics-based

model stops at the length of 2048 pixels, because it can be foreseen that by modeling even longer track, the computation time will increase rapidly to days. With such limited computational resources, yNet can however easily handle extremely large layers with a maximum length of ~21248 pixels. Meantime, modeling even this large layer remains to be nearly instant for yNet. In this regarding, the component-level SLS simulation (i.e., hundreds of large layers), which is previously a formidable task even for high-end computing facilities, is now accomplishable in a few minutes with yNet by simply using a laptop.

4.1.4 Discussion

4.1.4.1 Limitations

The current data-driven porosity modeling is based on two assumptions: (1) the sintering effect is largely decided by the laser condition, i.e., consistent along the entire sintered layer for a certain laser condition; (2) in multi-layer cases, inter-layer interaction mainly includes re-sintering of previous layers during scanning the newly added layer. In other words, the current modeling ignores layer-wise heat accumulation and its effect on porosity evolution. Only with the above assumptions, we can train yNet based on dataset of small patches and easily scale it to handle multiple layers of long tracks. Therefore, the present data-driven modeling can be further improved by incorporating layer index as condition parameters. Also, to account for inconsistent sintering effect along the sintered track (e.g., by strong heat dissipation on two sides), we should consider the distance of current sintering location to two sides or scanning time as an additional condition parameter.

Chapter 5 Property modeling

5.1 Data-driven modeling of stress field

5.1.1 Background

Stress field development under external loading is of common interest for structures at many scales, from materials microstructures to mechanical components. Data-driven modeling of stress field development might be especially important for AM, in view of the mainstream application of AM for fabricating complex components. Stress concentration are common issues for components with great geometrical complexity. An accurate yet fast predictive capability of

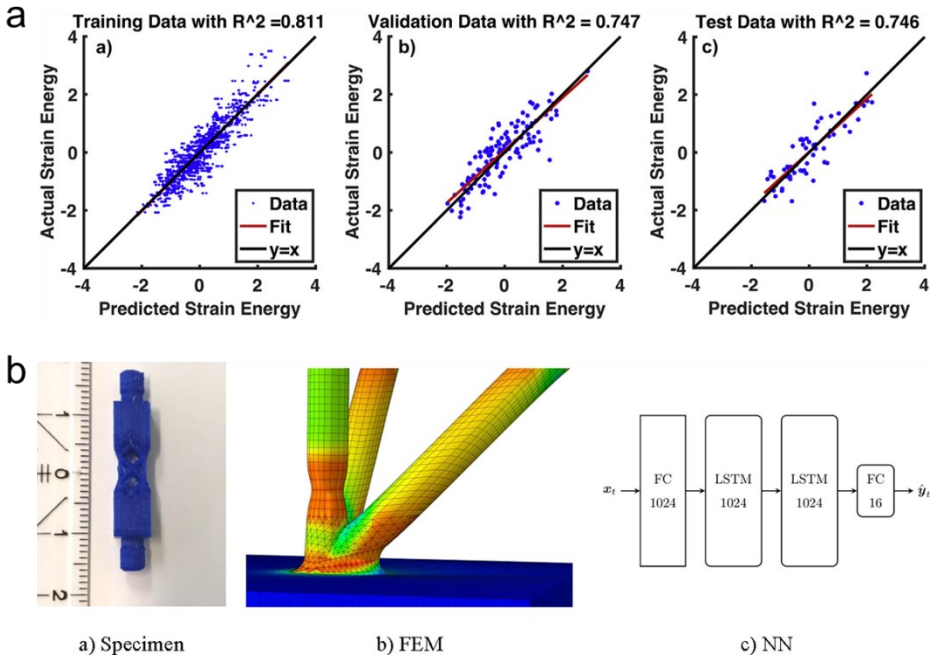

Figure 5.1 A summary of existing data-driven modeling of stress development within AM component.

stress field development is crucial for optimizing structural design and improving mechanical performance of AM components.

Despite its huge significance, there are only limited researches on this topic in AM community. As illustrated in Fig. 5.1a, Kantzos et al. [112] have used convolutional neural network to data-driven model the stress concentrations concerned with surface roughness, which is described by the total strain energy accumulated at the surface. Hoeppe et al. [113] have leveraged recurrent neural network for data-driven modeling the maximum stress in lattice structure at different deformations; see Fig. 5.1b. It can be easily found that a data-driven AM modeling effort on full-field stress development is still lacking.

5.1.2 Method

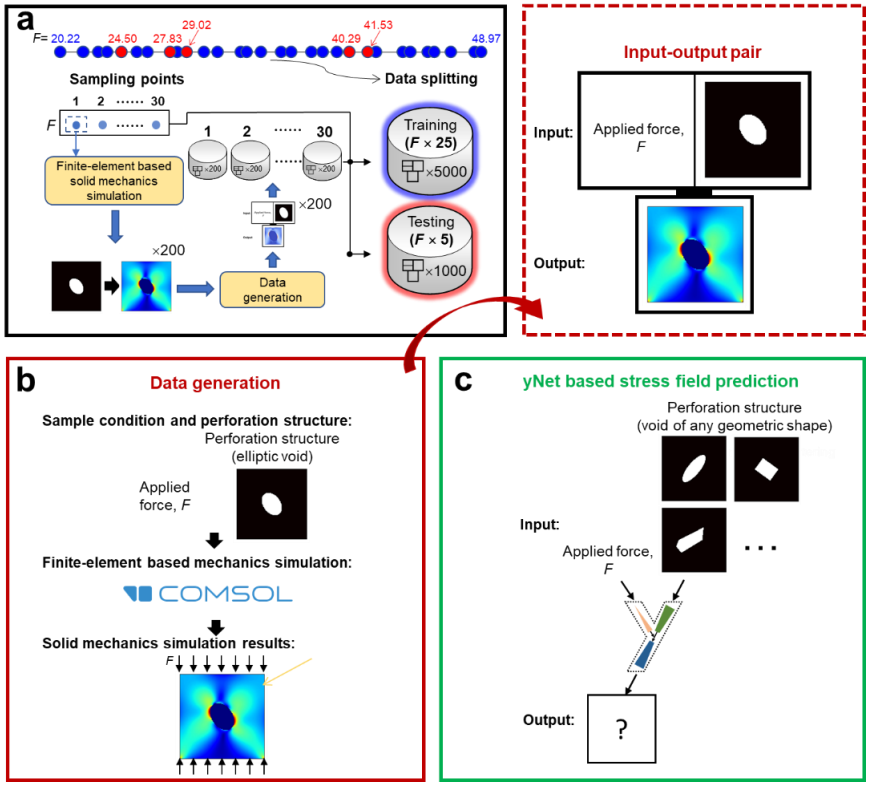

Figure 5.2 The overall procedure for training and testing yNet for predicting stress-field development within mechanical structure.

The stress field development for given structure and applied loading is also a multi-input field-to-field regression problem. In this regard, yNet developed for porosity modeling in Section 5.1.2 can be used herein. For illustration purpose, we trained yNet to predict stress field development within typical perforation structure (i.e., 0.1m×0.1m solid structure with a hole), when subject to compressive loading. The root reason for selecting perforation structure is that perforation structure, although with great simplicity, can serve as the unit cell of many lattice structures.

Fig. 5.2a shows the overall process of training and testing. The applied compressive force, F, is selected as condition parameter of interest in this problem. We firstly generate 30 random F sampling points ranging from 20to 50 MN. This is followed by performing solid mechanics simulations (Fig. 5.2b) with respect to 200 random perforation structures at each F sampling point. We train yNet with a specific type of perforation structures, which all feature elliptic hole albeit with random combination of hole orientation, size and ratio. The modeled materials is Ti6Al4V. As shown, the applied force, perforation structure and corresponding stress distribution result will comprise a data point. The 6000 results in total are split into training and testing data based on applied forces, i.e., 25 groups randomly selected for training and 5 groups for testing. As illustrated in Fig. 5.2c, the trained yNet is expected to predict stress field development for perforation structures with holes of not only elliptical shapes but also other geometries.

5.1.3 Results

5.1.3.1 Testing on perforation structures with elliptical holes

We first routinely test yNet for perforation structures with elliptical holes; see Fig. 5.3. These perforation structures in this testing have the same type of elliptical hole as that in training dataset, but with different hole orientation, size and ratio. The presented 5 results are randomly

selected out of 1000 testing results. As implied by the comparison, the prediction by yNet overall agrees well with FE-based simulation. The RMSE for all 1000 testing results is as small as 11.29 MPa. Therefore, although the input and output images are totally different and display huge changes, yNet can still correctly predict those significant "evolvements". Noteworthy is that, such a data-driven stress field prediction capability also naturally addresses the data-driven prediction of stress hotspot. The prior art of this research area is grain-wise stress-hotspot prediction for grain microstructures [114, 115], which also relies on expert knowledge to hand-design contributing factors to stress hotspot. The yNet trained herein can predict the detailed distribution of loading-dependent stress field, thus allowing for direct pixel-level prediction of stress hotspot by simply thresholding using the critical stress.

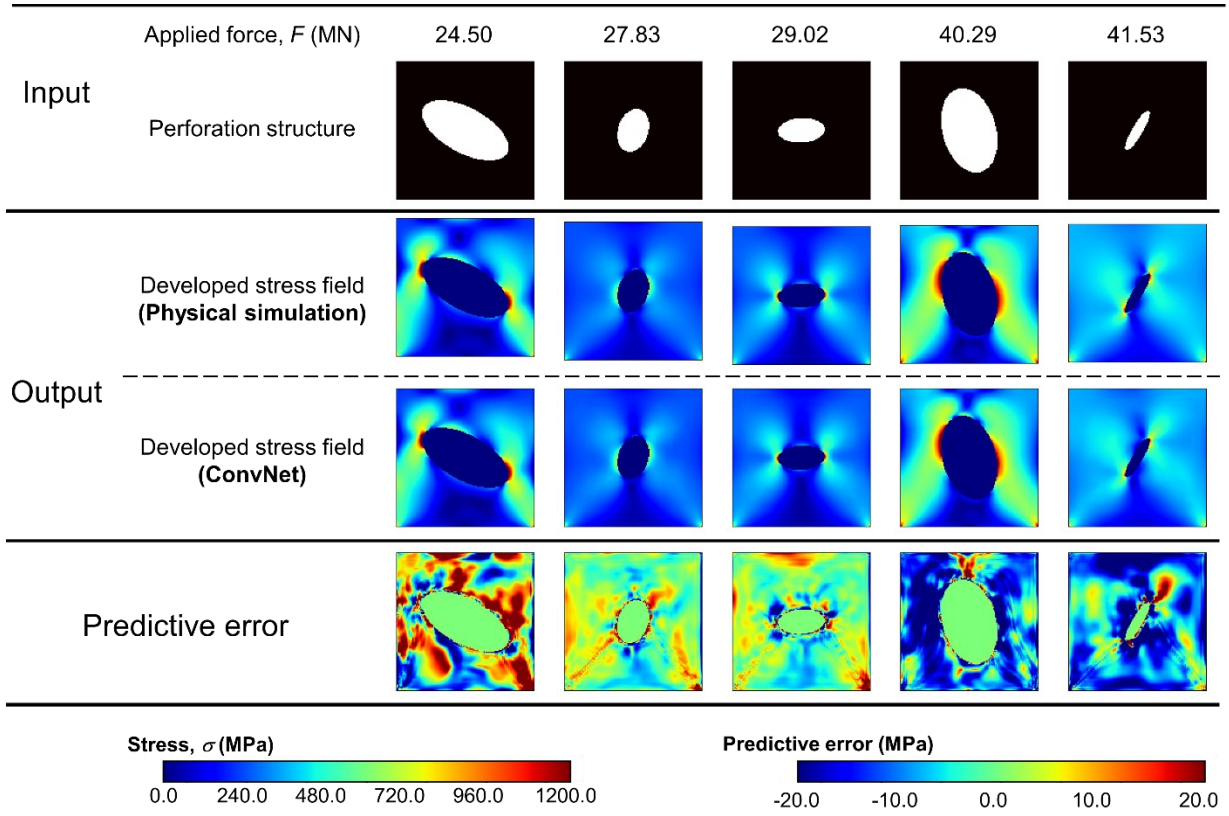

Figure 5.3 Testing of the trained CNN on perforation structures with elliptical holes.

5.1.3.2 Testing on other types of perforation structures

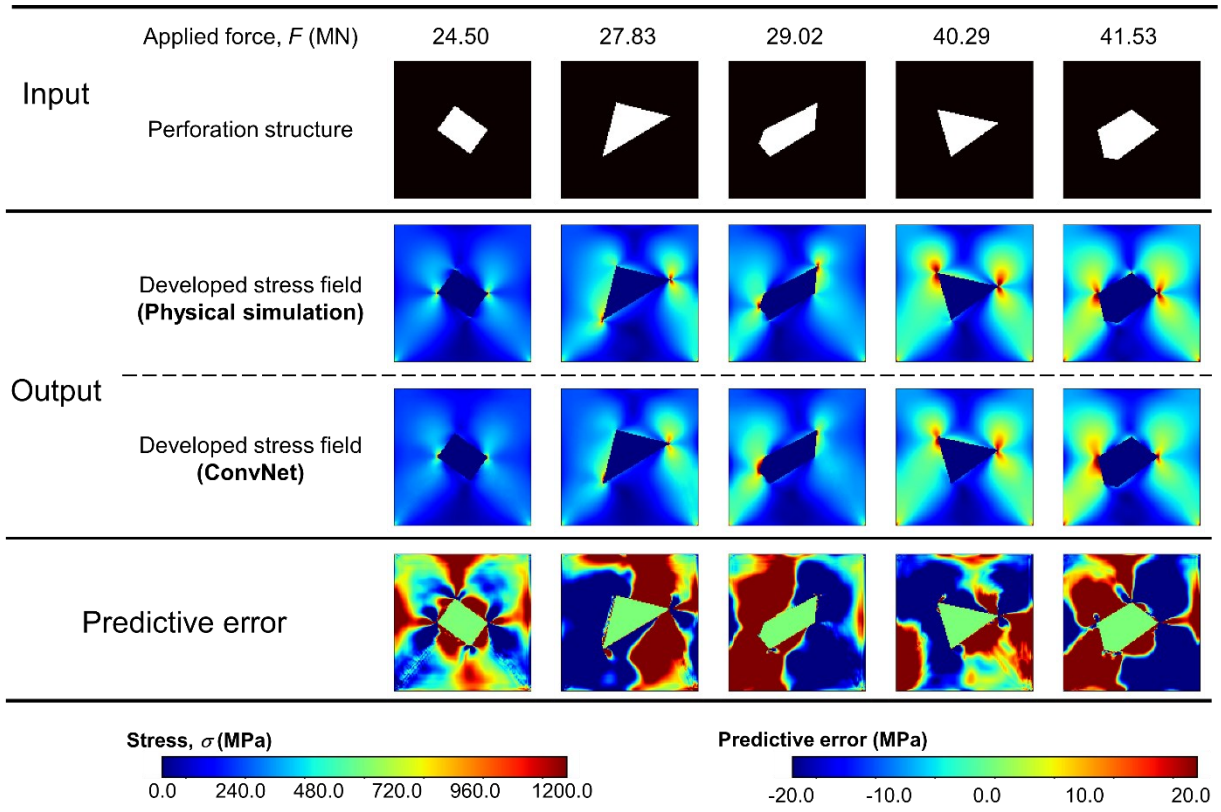

Figure 5.4 Testing of the trained CNN on other types of perforation structures

Besides the routine testing, we further test yNet for perforation structure with other categories of holes, including triangle, rectangle, and polygonal ones. Therefore, in this second-round testing, these perforation structures have completely different types of holes than that in training dataset. Five randomly generated rectangle, triangle and polygonal holes are tested. For the 5 random testing results in Fig. 5.4, the RMSE is 20.97, 50.43, 49.66, 41.33, 51.71 MPa respectively, which are relatively higher that of testing with respect to elliptical holes. Although the predictive performance shows a small decrease, from the comparison of detailed stress distribution, yNet still reasonably predict stress field development. It is thus believed that yNet has learned the basic rule for stress field development. For example, stress concentration tends to take place during sharp geometrical transitions along the loading direction and also initiates from two

bottom corners with increasing loading (associated with fixed boundary at the bottom in the current modeling); the discontinuous geometrical changes transverse to the loading direction will however not serve as significant stress risers. Those rules are fully contained in the training dataset and universally valid for perforation structures of any type. Learning those fundamental rules thus endows yNet great extrapolation capability in coping with different perforation structures.

5.2 Data-driven modeling of stress-strain curve

5.2.1 Background

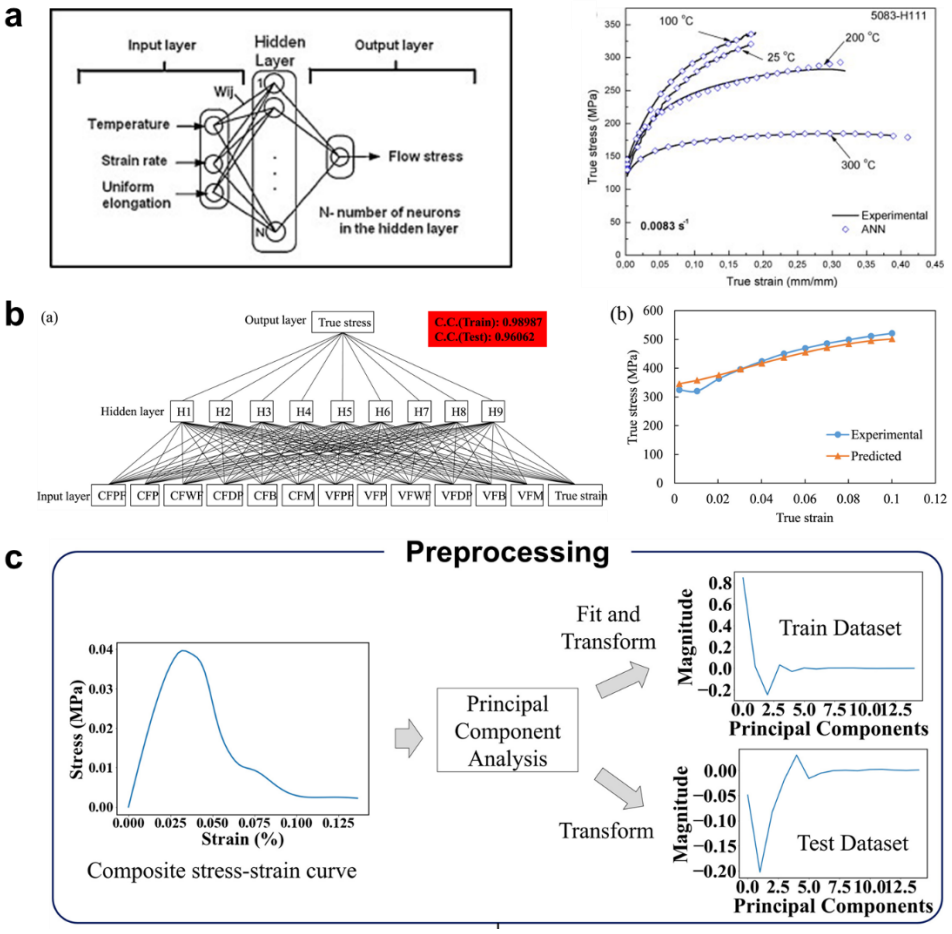

Figure 5.5 A summary of existing data-driven modeling of stress-strain curve.

Stress-strain curve is an informative representation of mechanical property, revealing many scalar properties, such as yield strength, Young's modulus, stiffness, etc. However, instead of modeling the complex stress-strain curve, the bulk of existing data-drive property modeling in AM is centered on those specific scalar properties, such as ultimate strength [56, 116, 117] and yield strength [55, 57, 118]. Those researches either link properties to AM process (parameters) by skipping the structure stage (i.e., building P-P linkage) [56, 116, 117, 119] or build structure-property (S-P) linkage by using structure as input[55, 57, 118].

As mentioned, in contrast with those scalar properties, data-driven modeling of stress-strain curve in AM has been less studied, largely due to the stress-strain curve as a complex quantity. Wang et al. [120] have proposed to use singular value decomposition (SVD) to reduce dimension of stress-strain curve, followed by data-driven stress-strain curve modeling in the latent space. In other fields there are indeed a few research efforts on this theme, and those data-driven modeling methodologies can be easily extended to AM. Most of those researches [121-123] adopted a step-wise modeling method, namely building real-time mechanical response as a function of input features (Fig. 5.5a-b). In this case, elongation or strain as a time-step indicator has to be included as an input. The entire stress-strain curve is then obtained from a series of separate predictions at different time-step inputs, i.e., discrete points in Fig. 5.5a-b. In addition to step-wise method, Yang et al. [124] have used principal component analysis (PCA) for dimension reduction and facilitated data-driven modeling of stress-strain curve as a whole (Fig. 5.5c). This idea is similar to the above-mentioned SVD-based approach, but they have further integrated CNN for microstructure representation modeling, instead of using structural descriptors as input features of the data-driven structure-property model. The existing approaches thus mainly have two limitations: (1) they highly rely on human labor and expert knowledge to hand-design structural descriptors, which

however easily lose information of original input structure; (2) they use conventional PCA to reduce dimension of stress-strain curve, which is relatively laborious and complicated; and/or (3) they fail to model the stress-strain curve in its entirety.

5.2.2 Method

In this subsection, we will show how to use a novel ML-based strategy for stress-strain curve prediction and overcome the above limitations with no efforts. Essentially, we take inspiration from image captioning task in computer science. The objective of image captioning is

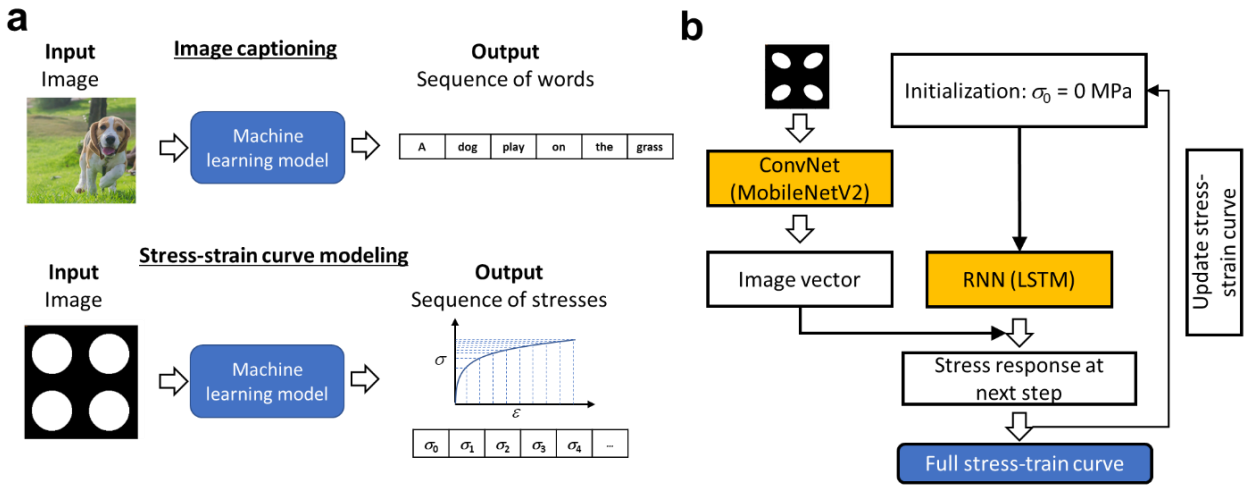

Figure 5.6 (a) Comparison of image captioning and stress-strain curve modeling. (b) Illustration of the proposed hybrid ML model for stress-strain curve modeling.

to train ML model to understand image and then generate reasonable textual description; see Fig. 5.6a. We observe that the heart of stress-strain curve modeling is interestingly close to image captioning. Stress-strain curve can be represented as a sequence of stress responses at a defined interval of strains. Thus, both of these two tasks are to generate sequence based on the image input. In this case, the ML model originally for image captioning can be adapted for data-driven modeling of stress-strain curve. Note that, for illustration purpose, the structure considered herein is also a typical type of perforation structure, which is completely symmetrical. By repeating such

perforation structure in space, we can easily get a full lattice structure. However, the proposed model should be generalized to other types of mechanical structures.

Specifically, we will use a hybrid CNN+RNN (recurrent neural network) to address image understanding and stress-strain curve generation tasks, respectively. There are four different ways to merge CNN with RNN in terms of merging location [125]. In our model, we select to merge the image vector outside of the RNN. The detailed structure of the implemented model is depicted in Fig. 5.6b. The model is realized using Keras [126] with Tensorflow as backend. The whole training and testing process is detailed as below:

1) Preprocessing of image dataset: since 2D images of mechanical structures cannot be directly fed to RNN, we first use pre-trained CNN model to convert each structure image to embedding vector. While a lot of pretrained CNNs are available in Keras, we choose MobileNetV2 [127] due to its light weight. From the official Keras document, MobileNetv2 has the smallest size of only 14MB among all pretrained CNNs (as a comparison the known VGG-16 [128] is 528MB), and it has however comparable image understanding capability as indicated by its Top-1 and Top-5 Accuracy in ImageNet challenge. Moreover, the final hidden (image-embedding) layer of MobileNetV2 is only 1280-dimensional compared to 4096-dimension of VGG-16. The above merits of MobileNetV2 together will allow for efficient generation of information-rich yet memory-friendly image-embedding vectors for the structure images.
2) Obtaining stress-strain curve dataset: Raw datasets are first obtained from physical simulation by using COMSOL Multiphysics [129]. The modeled material is structural steel. Especially, to correctly model the stress-strain curve, the built-in Johnson-Cook material model is used to capture hardening behavior:

$$\sigma_{ys}\left(\varepsilon_{pe}\right)=\left(\sigma_{ys0}+k\varepsilon_{pe}^{n}\right)\left(1+C_{\max}\left(0,\log\left(\frac{\dot{\varepsilon}_{pe}}{\dot{\varepsilon}_{0}}\right)\right)\right)\left(1-f\left(T_{h}\right)\right) \tag{5.1}$$

where σ_{ys0} , k, n, $\dot{\varepsilon}_0$ are materials parameters and $\dot{\varepsilon}_{pe}$ is applied strain rate. We have performed simulation for 200 randomly designed perforation structures.

3) Preprocessing of raw dataset:

   We further properly create input-output pairs for training and testing. Each physical simulation will give us a full stress-strain curve of the modeled mechanical structure, which are described by a sequence of stress response (101-step long) at different deformation stage. A sequence of 101-step stress response can give us 101 input-output pairs, i.e., [σ_0]-[σ_1], [σ_0, σ_1]-[σ_2], ..., [σ_0, σ_1, …, σ_{100}]-[σ_{101}] (here we omit image vector input for notational simplicity). We have generate 200 groups of such datasets, based on 200 randomly designed perforation structures. 180 groups of them will be used for training and 20 for testing.

4) Model training and testing: we train the model for 100 epochs with mean squared error (MSE) function, by using Adam optimizer with default setting of learning rate of 0.001. We dynamically monitor the val_loss and the best weights with smallest val_loss is saved. During testing after training, we initialize the stress input as σ_0 = 0 MPa for all structures.

5.2.3 Results

5.2.3.1 Testing on unused structures

Fig. 5.7 presents five testing results for perforation structures unused during training. It can be found that the ML prediction matches well with the ground truth of physical simulations. For all the 20 testing structures, the proposed ML model achieves a RMSE of 8.12 MPa. Therefore,

regardless of hole configuration of the perforation structure, ML model can accurately reproduce the stress-strain curves. Note that, previous SVD-based method [120] is also able to model the stress-strain curve as a whole. However, SVD basically model the geometry of different stress-strain curve, instead of learning physical rule that govern stress-strain curve development. Also, the complex geometry of stress-strain curve will easily result in information loss during dimension reduction and curve reconstruction by SVD. The proposed model thus may significantly outperform the prior SVD-based model.

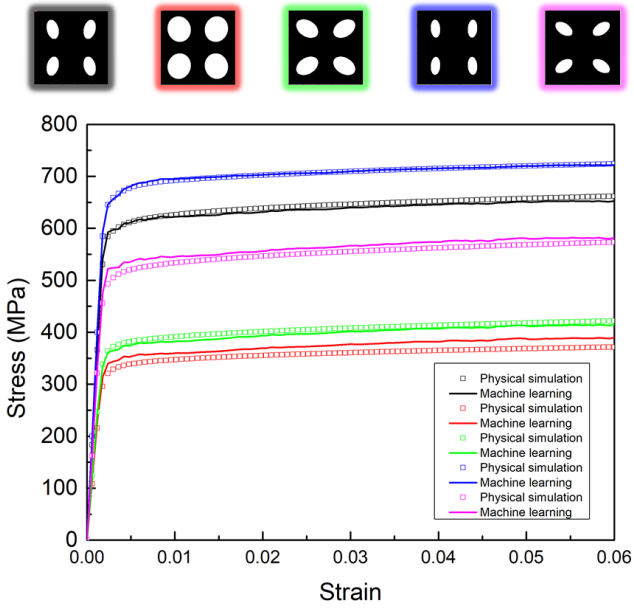

Figure 5.7 Testing of the trained ML model on perforation structures unused in training.

We point out that, although for illustration purpose we use simulation data to train the ML model, the proposed model is essentially experiment oriented. Because stress-strain curve is usually the property quantity that we measure and care about in practice. Once experimental data of stress-strain curve is available, we can train the ML model by following the same procedure. On the contrary, the data-driven stress-field predictive model proposed in Section 5.1 is for simulation use. The detailed stress field data is hardly measurable in experiments. It thus is expected to be trained with simulation data only and reproduce the physical simulation. In this

case, it allows for theoretical study of stress-field response of AM-fabricated structure, while CNN+LSTM proposed in this section permits more practical study of stress-strain curve. Therefore, the two data-driven property models proposed in this research are complementary to each other.

Chapter 6 Discussion

6.1 Implication on other data-driven modeling in AM

The five presented data-driven modeling should well cover many types of data-driven modeling problems encountered in data-driven AM modeling. The developed methodologies are anticipated to have a broad impact on other data-driven modeling in AM. A few examples are discussed as below.

6.1.1 Data-driven modeling of bead geometry

For example, another type of data-driven AM modeling very similar to melt pool modeling is the modeling of bead geometry and size. Melt pool and bead are highly relevant AM quantities, both essentially concerned with the high-temperature molten region, but bead is just more related to molten materials upon solidification. In fact, their geometrical characteristics, e.g., depth and width between melt pool and bead, are sometimes not really distinguished [65, 130]. The only big difference might be that solidification gives rise to a new geometrical quantity - bead height, data-driven modeling of which is important for building precision control along z-axis [131]. The overall significance and philosophy of data-driven bead modeling are thus almost the same to that of data-driven melt pool modeling. Therefore, the presented idea of incorporating scanning history to inform melt pool prediction may be borrowed to improve the data-driven modeling of bead geometry.

6.1.2 Data-driven modeling of grain structure

Data-driven modeling of grain structure is of particular significance for AM process, since complex and unconventional grain structure is one of the signatures of AM-fabricated components [18]. Grain structure development within AM part has always been an active research topic in AM community [132]. Data-driven grain structure modeling will greatly benefit process plan for tailoring grain structure within AM parts. In contrast with many physical modeling and experimental efforts, data-driven grain structure modeling has been surprisingly little explored. Wang et al.[133] have built the formed grain structure, described by the mean and variance of grain aspect ratio, as a function of AM process conditions and other input features. Similarly, Popova et al. [54] have correlated grain structure, but quantitatively represented by principal component (PC) scores of PC analysis on grain chord length distribution, with various AM process parameters. From the few related researches, it can be seen that data-driven grain structure modeling is still at its infancy, focusing on building mapping between simple microstructure descriptors and manufacturing conditions.

Therefore, existing data-driven modeling of grain structure thus faces the same dilemma as that of porosity, namely a lack of explicit modeling of real structure. Regarding this, the proposed multi-input CNN for data-driven porosity modeling can be easily extended to realistically simulate the grain growth; because both of the two data-driven modeling tasks are basically condition-dependent structure evolution.

6.1.3 Data-driven modeling of external shape

The necessity of modeling external shape, from mesoscale cross-sectional shape of struts [134] to macroscopic shape of the whole component [135], stems from geometrical distortion and dimensional inaccuracy in AM. This problem is generally caused by internal stress associated with

the complex thermal behavior of AM process [136], insufficient support for overhangs [134] and limited printing resolution of AM machine [137]. They lead to deviation between designed and fabricated geometry/shape. Accurate modeling of geometrical distortion is critical for correct compensation of geometrical deviation in the computer-aided design (CAD) phase. Admittedly, data-driven modeling of geometrical distortion might be the most developed data-driven AM modeling. For example, unlike data-driven grain structure or porosity modeling, the majority of existing data-driven modeling of geometrical distortion focus on the real geometry, rather than dealing with some simple deviation descriptors (e.g., percentage of shrinkage/expansion). Also, some novel data-driven modeling methodologies based on CNNhave been proposed for explicitly handling the 2D or even 3D component geometry. Most of existing data-driven modeling may be categorized into two groups. The first group adopted localized modeling of points that form the geometry/shape. For instance, Chowdhury et al. [138] have attempted to correlate Cartesian coordinates of nodes between the designed and actual model. However, a preferred way in this group is to describe points and thus their deviation in polar coordinate system (PCS) [134, 135, 139], which facilitates the modeling of complicated deviation mainly as a function of the polar angle. The second group mainly leverages CNN for direct data-driven modeling of the whole shape [137, 140], thus usually enjoying better generalization to various shapes. The core principle behind them is to use image-to-image regression capability of some CNNs, namely those with an encoder-decoder structure, where encoder reads original shape and decoder reconstructs encoding results to the shape after distortion.

However, the existing CNN-based modeling simply adopted encoder-decoder architecture originally for image segmentation, which are unable to incorporate the effect of AM conditions on geometrical distortion. The trained CNN is strictly applicable to the specific AM condition used

for training. Again, the proposed multi-input CNN for data-driven porosity model can be used to overcome this limitation. It can build functional relationship between multi-input (i.e., AM conditions and designed geometry) and fabricated geometry. This capability will allow us to train a single data-driven model applicable to predicting geometrical deviation under various AM conditions.

6.2 Implication on data-driven modeling beyond AM

This section will further show a detailed example of extending the proposed model to data-driven modeling in other domains.

6.2.1 Data-driven modeling of fluid dynamics

The proposed yNet for data-driven modeling of porosity in AM is a generally useful regression tool. To demonstrate this, in this section we present its application to fluid dynamics simulation. Specifically, we choose a classical fluid flow problem – incompressible, viscous flow shedding around a cylinder. In this case, the spatiotemporal evolution of viscous fluid field is governed by:

$$\frac{\partial \omega}{\partial t} = \nabla \times \left(\boldsymbol{v} \times \omega \right) + \nu \nabla^2 \omega \tag{6.1}$$

where $\omega = \nabla \times \boldsymbol{v}$ is the vorticity, $\boldsymbol{v}$ is the velocity of the fluid, ν is the kinematic viscosity of the fluid. Variables in Eq. (6.1) can be rewritten in scaled units with respect to unit length L and unit fluid velocity V, namely, $x = x'L$, $\boldsymbol{v} = \boldsymbol{v}'V$, $t = t'\frac{L}{V}$, $\omega = \omega'\frac{V}{L}$ Consequently, Eq. (6.1) can be rewritten as:

$$\frac{\partial \omega'}{\partial t'} = \nabla' \times \left(\boldsymbol{v}' \times \omega' \right) + \frac{1}{Re} \nabla'^2 \omega' \tag{6.2}$$

where $Re = \frac{LV}{\nu}$ is the Reynolds number as a function of kinematic viscosity, unit length and unit fluid velocity. From Eq. (6.2), we can find that Reynolds number is the dominating condition that dictates the flow behavior. Therefore, in the following research, we will train a single yNet capable of simulating fluid flow evolution over a range of Reynolds numbers. The trained yNet will be tested on: (1) predicting flow field development for *Re* conditions unused in the training datasets; (2) simulating dynamic flow field evolution via recursive predictions; (3) temporal extrapolation for predication, i.e., extending the dynamic simulation beyond the original time domain in training data.

6.2.1.1 Training and test procedure

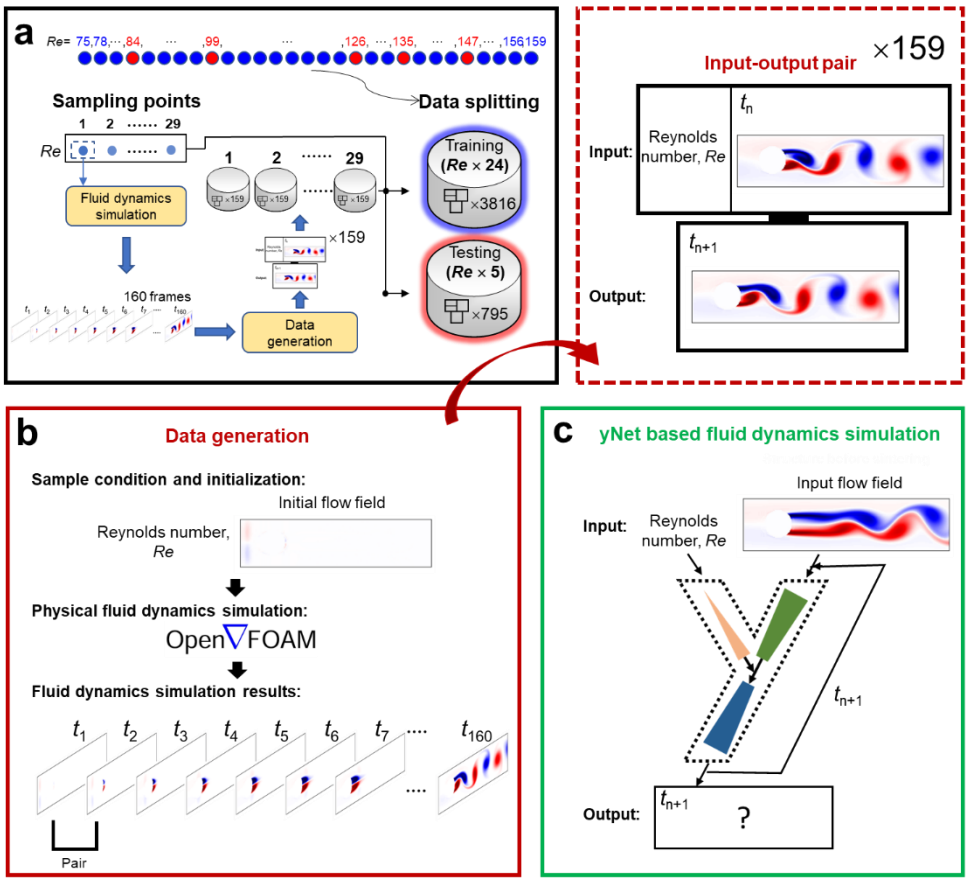

Figure 6.1 Illustration of training and testing of yNet for fluid dynamics simulation.

Fig. 6.1 illustrates the overall procedure of training an yNet applicable to a range of Reynolds numbers. Vorticity field is selected as flow field of interest and Reynolds number as the condition parameter, but we can easily train yNet for other flow fields in the same way. To achieve simulation of dynamic field evolution, yNet acts to predict next-step flow field based on the current flow field and Reynolds number, *Re*. As such, long-term flow field evolution simulation can be performed by yNet through multi-step recursive predictions as shown in Fig. 6.1c.

To provide dataset, physical simulations are performed at 29 equispaced Reynolds numbers (i.e., *Re* = 75, 78, 81, … ,159), where 24 cases would be randomly chosen for training and the other 5 for testing; see data splitting in Fig. 6.1a. For all those physical simulations, extraction of neighboring snapshots is performed to generate one-step-distance data pairs, as illustrated in Fig. 6.1b. A physical simulation will output a total of 160 frames, indicative of 159 data pairs obtained at each Reynolds number sampling point or 4611 for the entire dataset. They are split into training and testing datasets based on Reynolds numbers as mentioned early. Training process is fully monitored and the weights with the smallest testing loss after reaching convergence are used to generate results in the next subsection.

6.2.1.2 Testing under unused Re conditions

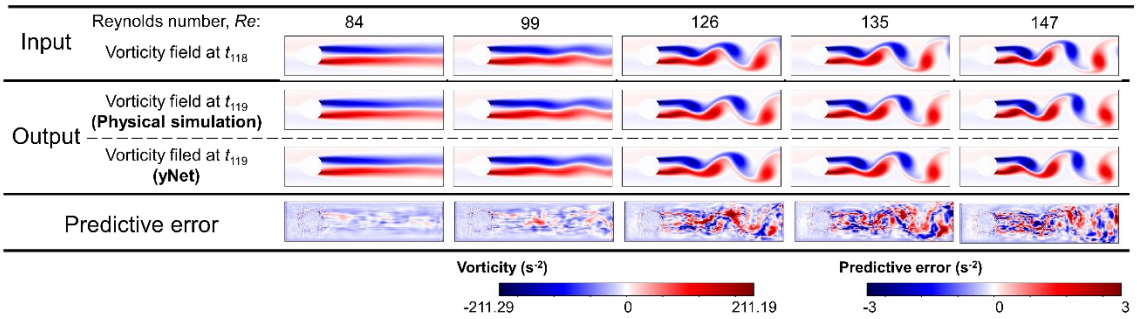

Figure 6.2 Testing results of yNet under different Re conditions.

Fig. 6.2 presents a set of testing results, which show yNet in predicting next-step flow field development under different Re testing conditions. As clearly indicated by the predictive error,

yNet can accurately predict the next-step flow field in all cases, with minor difference to the original physic-based simulation. For all the 795 testing results, yNet achieve a negligibly small root mean squared error (RMSE) of 0.588 s^{-2}. It represents one of the first data-driven modeling that directly incorporates Reynolds number. Previous strategies for considering Reynolds number are usually to train a neural network that takes as inputs a series of historical flow fields [141-143], which as a whole embed corresponding *Re* information; see LSTM+CNN model. Thus, the neural network can predict the next step based on the evolving trend of consecutive flow fields, without being informed by any additional inputs like Reynold number. yNet however explicitly incorporates the important condition of Reynolds number, and thus possess two major advantages over previous strategies. First, yNet is as conceptually simple as a basic CNN, without the complexity of integrating sequence-modeling purposed LSTM model [141, 142]. Second, yNet is widely applicable to other condition-dependent fluid flow simulation problems. For instance, besides modeling the dynamic flow field evolution process, another task of great interest is to predict the development of flow field for different bluff body shapes [144, 145]. This is essentially a different mapping problem between distinct field quantities, i.e., from structural field (bluff body structure) to flow field. For such regression tasks beyond standard field evolvement, the aforementioned LSTM+CNN model apparently will no longer work. We have to explicitly incorporate effects of condition parameters on the flow field development, by using our proposed multi-input CNN.

6.2.1.3 Testing on dynamic fluid simulation and temporal extrapolation

Fig. 6.3a visualizes a representative testing result of predicting multi-step evolution, under the testing condition of *Re* = 84, through recursive inference as depicted earlier in Fig. 6.1c. In the dynamic simulation, we start from No. 128 step, which is around the wake of flow under *Re* = 84.

We presented snapshots at some time points to quickly show the dynamic evolution process. Noteworthy is that predictive errors can accumulate after many steps of recursive predictions. However, compared to the ground truth of physics-based simulation, yNet well reproduces the long-term dynamic flow field evolution, suggestive of rather small error during recursive prediction at every step. The predictive error (i.e., RMSE) over multi-step predictions for different *Re* testing conditions is summarized in the left half of Fig. 6.3c. The quantitative result reveals that predictive error does accumulate as recursive prediction proceeds, but remains within a small level, i.e., $< 12.0\ s^{-2}$ for different testing conditions.

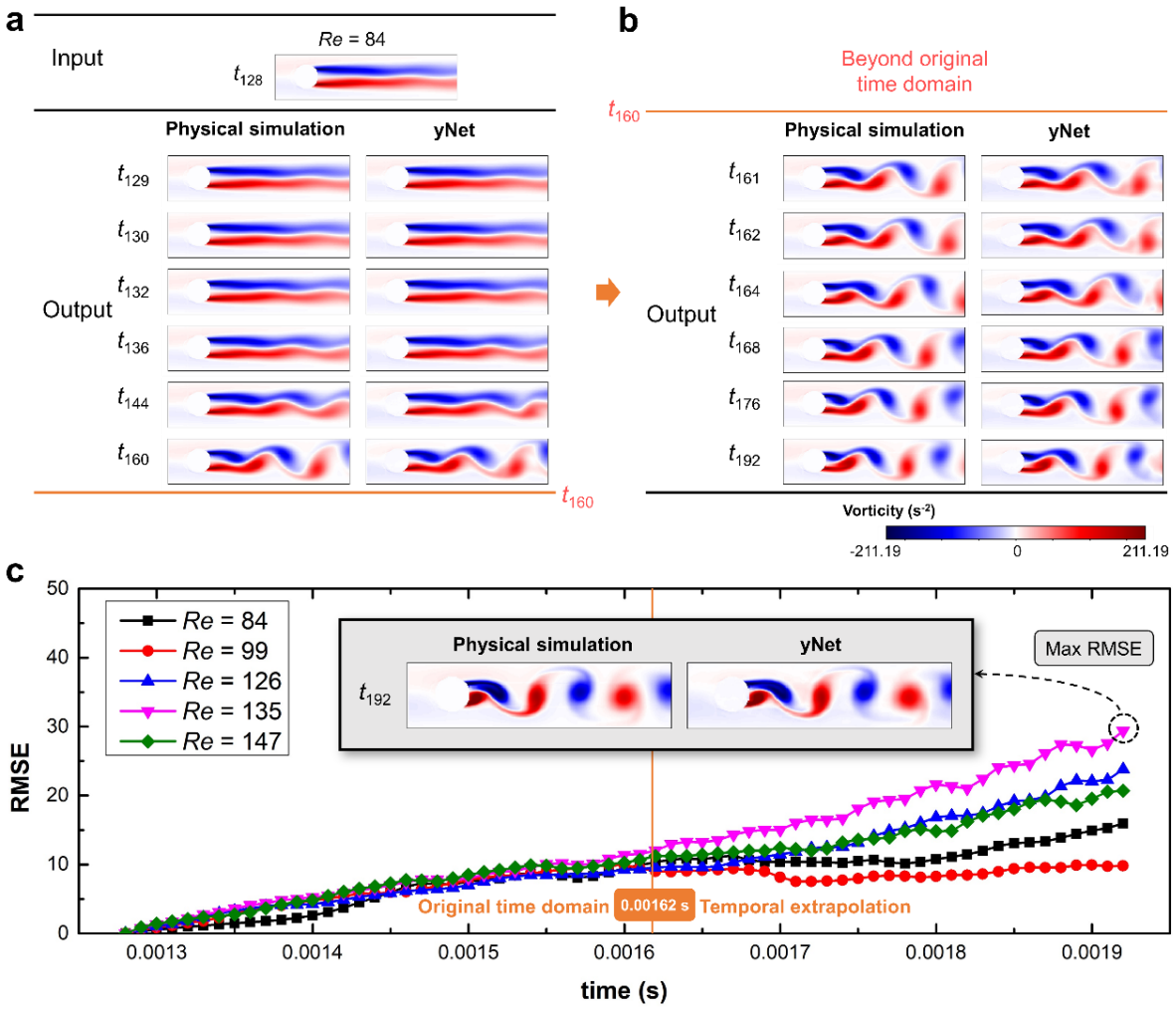

Figure 6.3 Testing results of the trained yNet for dynamic fluid simulation. (RMSE: root mean squared error)

In addition, attempts have been made to predict fluid dynamics towards the further future, i.e., beyond the time domain in training data; see the visualized example for Re = 84 in Fig. 41b. Correspondingly, for validation purpose we also extend the physical simulation for another 32 frames. Despite the big challenge associated with temporal extrapolation, predictive error continues to display gradual accumulation without a sudden rise, as shown in the right half of Fig. 6.3c. Notably, predictive error can finally reach a maximum RMSE of 29.37 s^{-2} for Re = 135. However, such a predictive error is acceptable, still prviding good prediction of flow field as clearly shown by the inset in Fig. 6.3c. The above results indicate the successful learning of underlying evolution dynamics by yNet, which supports its temporal scalability to reasonably predicting flow field evolvement in far future.

6.2.1.4 Comparison with conventional multi-input CNN

Fig. 6.4a and b compare the computational accuracy and efficiency, respectively, between yNet and the conventional flattening-based multi-input CNN. For fair comparison, both networks are implemented with the exact same encoder and decoder parts, but using gating and flattening based signal-merging strategies respectively. Also, since there are a few variants of flattening based strategy, we implement one by directly concatenating Re to the flattening result. As testing loss curves in Fig. 6.4a indicate, after reaching convergence of training (> 60 epochs), yNet in general achieves comparable testing accuracy (MSE = 1.10×10^{-6}) to that (MSE = 1.16×10^{-6}) of the conventional multi-input CNN. The inset in Fig. 6.4a further shows almost the same prediction by the two trained models at a randomly selected testing data point. However, yNet can substantially reduce the number of model parameters by one order of magnitude and thus enjoys various computational benefits against the conventional CNN, such as significantly reduced memory consumption, shorter training time and faster inference speed; see Fig. 6.4b.

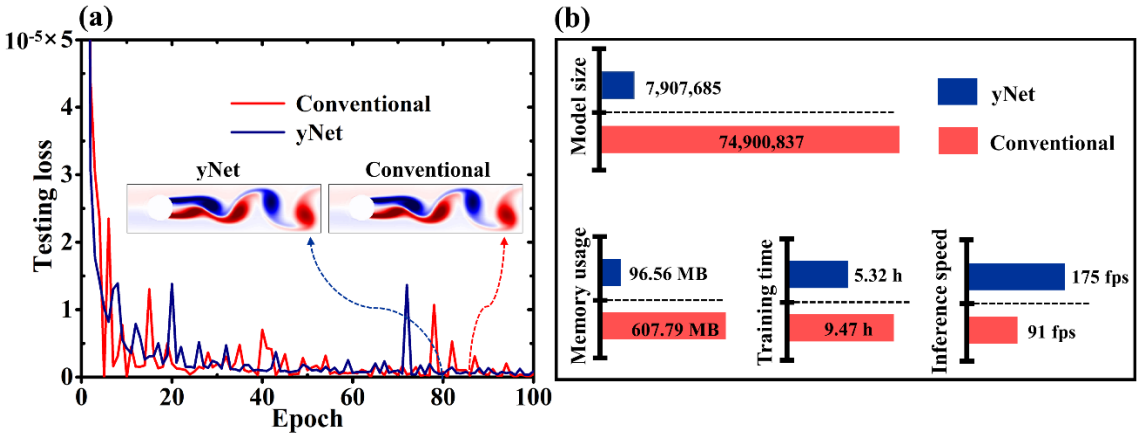

Figure 6.4 Comparison of yNet with conventional multi-input CNN. (a) testing loss curves of yNet and conventional multi-input CNN; (b) comparison of model size (i.e., total number of parameters) and three other performance metrics between yNet and conventional multi-input CNN. The performance is measured purposely based on a laptop (Intel Core i7-7500U CPU, NVIDIA GeForce GTX 950M GPU, 16G RAM). Note that memory usage is measured with a single unit input, instead of a batch.

The above merits of yNet are attributed to gating-based signal merging strategy, which permits effective interaction between two input channels while inducing only a minimal number of model parameters. The root reason for the effectiveness of directly manipulating feature maps might be further revealed by drawing an analogy to the conventional data compression technique - proper orthogonal decomposition (POD). It is known that POD is close to a basic feedforward neural network with certain constraints [33, 146]. In fact, there are also interesting parallels between POD and CNN. POD is a linear-theory based feature extraction technique. Through POD, a high-dimensional image can be expressed as a linear combination of orthogonal modes, i.e., the extracted features. Feature maps in CNNs, albeit in a 2D matrix form, play a similar role as the coefficients of the linear combination in POD. Like those coefficients corresponding to their respective mode, a feature map in CNN would strongly correlate with a certain feature in the pixel space if one projects them back to pixel-based image reversely along the encoding process [147]. Recent researches also prove that one may build separate deconvolution networks to visualize detailed representation of encoded latent variables in high-dimensional image space [148] and

even analyze their energy contribution like the POD mode decomposition [149]. Although those researches focus on the latent variables (i.e., further compressed latent representation from feature maps), similar methodologies can be used to deconvolve the feature maps to fully reveal their high-dimensional looks. All of those facts imply the similar function of feature maps in CNN to that of coefficients of the linear combination in POD. Thus, direct manipulation of those "coefficients" in CNN will effectively generate new combinations of features towards the target image, as does a POD. The deep decoder would then further contribute to the accurate obtainment of target or output image through the non-linear and complex reconstruction

6.3 Experimentally calibrated data-driven AM model

For those data-driven AM models built purely on simulation data, experimental calibration is of huge importance. Therefore, although experimental calibration is not the major focus of this dissertation, we still offer a careful analysis and demonstrate its significance for building fully trustworthy data-driven model. Specifically, we will use data-driven melt pool modeling as an example, since melt pool is currently one of most easily measurable AM quantities and there is a great availability of experimental melt pool data. Details about experimentally calibrated data-driven melt pool model will be presented as follows.

6.3.1 Experimental calibration of data-driven melt pool model with uncertainty

6.3.1.1 Background: uncertainty in AM

First of all, we highlight that instead of focusing on the conventional experimental calibration, we calibrate data-driven model with uncertainty. The calibrated data-driven model is thus able to reasonably predict not only the expected melt pool dimensions, but also the potential variability/uncertainty of melt pool dimensions. This is anticipated to close long-existing gap

between theoretical model and practical experiment; that is, theoretical model is usually deterministic, while AM experiments in practice involve much variability. It is extremely challenging to repeat the manufacturing process and result in mass production, as evidenced by the interlaboratory AM tests [150, 151]. The variability in AM practice is mostly caused by uncertainty propagation from various uncertainty sources existing in the complex AM process [152]. Typical uncertainty sources include natural variation in powder absorptivity, fluctuation in temperature boundary, uncertainty in powder particle properties, and many others [153, 154]. For example, it has been pointed out that the quality and properties of deposits can vary greatly even when all producers used the same materials, processing parameters, and, in some cases, even the same type of AM machine [155]. By introducing uncertainty into data-driven model, the data-driven prediction will further approach the AM practice featuring variability.

6.3.1.2 Method

Fig. 6.5 shows the overall procedure for experimental calibration of data-driven melt pool model built on simulation data. Therefore, different from the data-driven melt pool model in Section 3.1, in this section we purposely build a melt pool model based on high-throughput physical simulations (instead of experimental data). We then employ the conventional and novel sequential Bayesian calibration method, (SeCAV) to perform uncertainty source parameter calibration and model bias correction. Appendix D.1 describes the two calibration methods in details. The experimental calibration will significantly improve the validity of the 3D melt pool surrogate model. Comparison of efficacy in model improvement will be compared between the two adopted calibration methods.

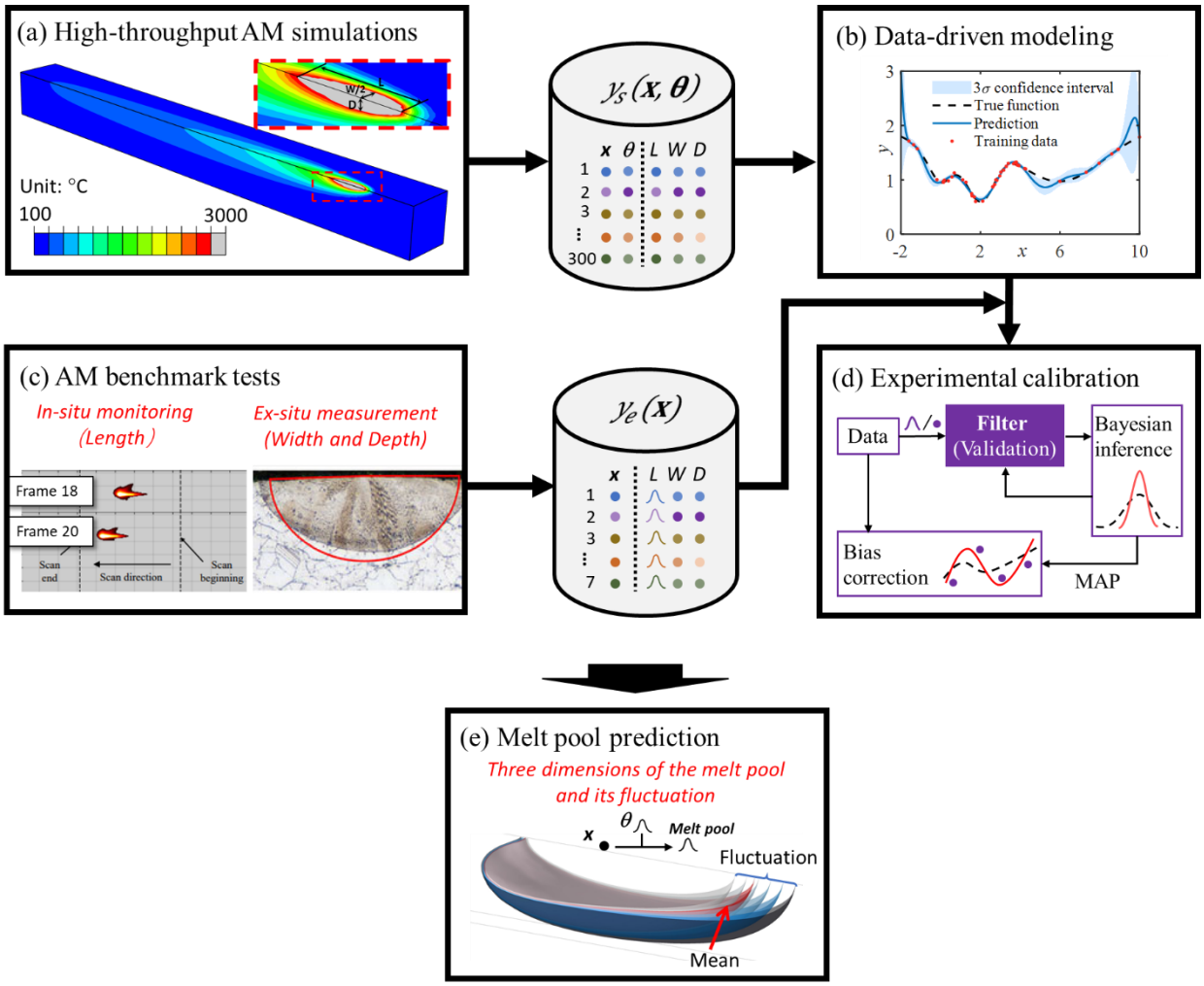

Figure 6.5 Experimental calibration of data-driven melt pool model with consideration of uncertainty.

6.3.1.3 Initial data-driven melt pool model based on simulation data

As shown in Fig. 6.5a, we first performed 300 AM simulations at random training points by the Latin hypercube sampling method[80]. We then construct three initial melt pool surrogate models for the melt pool length, width and depth, respectively, by considering both control factors **x** (applied power and scanning speed), and uncertainty factors, **θ**; see Fig. 6.5b. We considered five uncertainty sources, **θ** , including the fluctuating preheating temperature, laser absorptivity, power distribution radius, heat convectionat surface, and ambient temeprature. We conducted a 10-fold cross validation to test the effectiveness of the surrogate models in capturing the

relationship between the melt pool dimensions and the multiple inputs (i.e., **x** and **θ**). We split all the training data into ten groups with one group employed as test data and the other nine groups as training data for each cross-validation test. We adopted the Gaussian process (GP) model for data-driven modelling. The cross-validation results in Fig. 6.6 show a very minor deviation of all points from the reference line of $y = x$, implying a high prediction accuracy of the data-driven model. The average relative errors for the melt pool length/width/depth prediction are respectively 5.74%, 3.26% and 10.08%, giving us confidence that the computationally intensive AM simulations can be replaced by the constructed melt pool surrogate models.

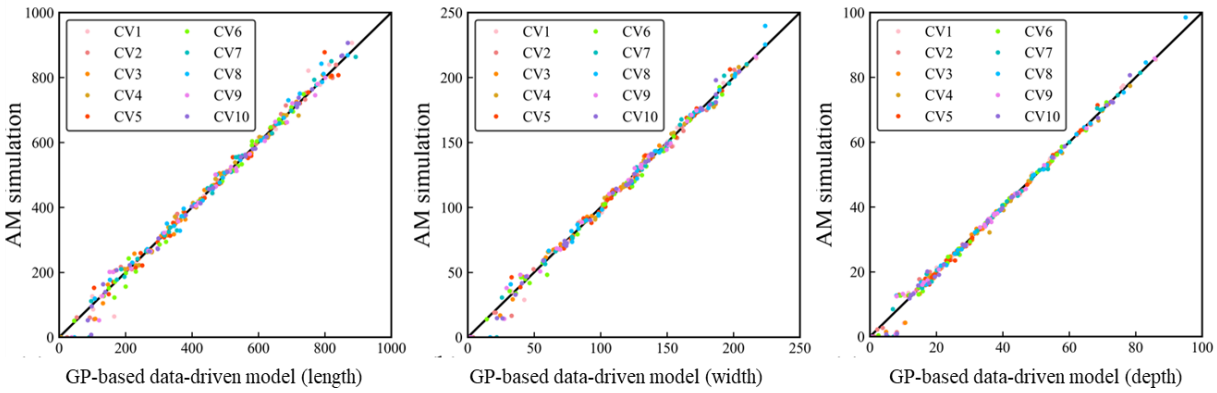

Figure 6.6 Comparison of melt pool between the AM simulation and prediction by the trained GP-based surrogate model..

6.3.1.4 Calibration of uncertainty source parameters

The variability of melt pool is caused by uncertainty sources, here denoted as **θ**, in AM. An intuitive approach to determining their accurate distributions is based on copious point data obtained from sufficient number of repetitive measurements. While directly measuring the uncertainty sources during AM process (e.g., fluctuating laser power absorptivity) is often technically challenging and sometimes even impossible, we would inversely infer their distributions based on relatively easily measurable QoI, herein the melt pool and its fluctuations.

This is basically accomplished through experimental calibration of their assumed prior distributions.

Table 6.1 Experimental data of the melt pool taken from AM-Bench project by NIST

Number of experiments	Laser power, P (W)	Scanning speed, V (m/s)	Melt pool dimensions(um)		
			Length	Width	Depth
1	49	0.2	171±16	111	24
2	122	0.2	519±29	186	48
3	122	0.5	361±27	134	33
4	122	0.8	315±27	117	25
5	195	0.2	824±109	259	109
6	195	0.5	903±102	162	49
7	195	0.8	813±79	133	38

As shown in Table 6.1 we use seven groups of experimental melt pool data (i.e. $y_e(\mathbf{x})$) taken from the AM-Bench project by NIST [156]. The pool fluctuation is represented by the interval data of measured pool length. With pool fluctuation data, we inversely infer the distribution of uncertainty source parameters. Fig. 6.7 shows updated distributions (i.e., posterior distributions) of various uncertainty source parameters after experimental calibration, by using the direct and SeCAV calibration methods respectively. Through experimental calibration, uncertainty sources exhibit apparently more concentrated distributions compared to their prior assumption for both methods. It is expected that, with availability of more melt pool fluctuation data, we can further approach the true distributions of those uncertainty sources.

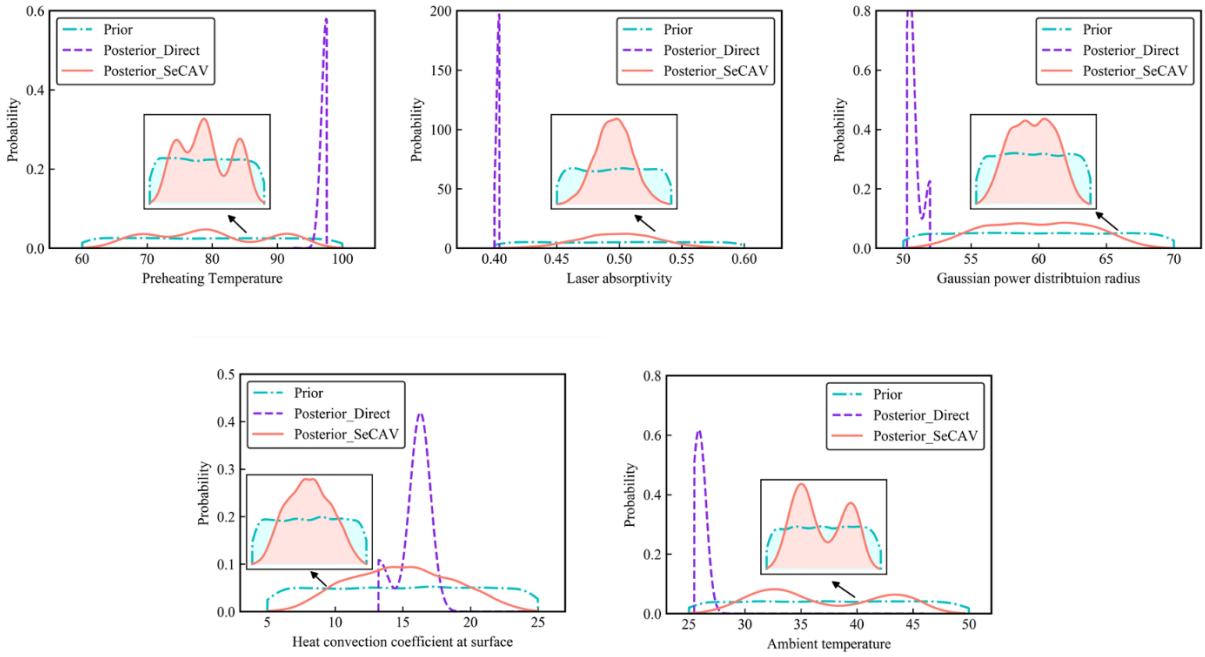

Figure 6.7 Experimental calibration of uncertainty source parameters by using the direct Bayesian and SeCAV methods.

6.3.1.5 Model bias correction

Another source of predictive error comes from model bias, i.e., the discrepancy between the simulation and experiment. We correct the model bias by training another data-driven model to capture the model bias, which is expressed as:

$$y_e(\mathbf{x}) = y_s(\mathbf{x}, \boldsymbol{\theta}) + \delta(\mathbf{x}) \tag{6.3}$$

where $\delta(\mathbf{x})$ is the model bias between the simulation model/surrogate model prediction, y_s and the experimental measurement, y_e. Fig. 6.8 summarizes all the 7 leave-one-out cross-validation results by using the direct and SeCAV calibration methods. Overall, the predictions of melt pool and its fluctuations from the surrogate model calibrated by using the SeCAV method match well with experimental measurements. In Fig. 6.8b and c, nearly all point data (measured pool width and depth) are located within the predicted intervals. However, in Fig 6.8a, comparing the predicted and measured pool length reveals a general overestimation of the variability in melt pool. This

overestimation might be alleviated or even eliminated with the availability of additional experimental data by further experimental calibration and enhanced inference of uncertainty sources. Nevertheless, for all 7 cases in Fig. 6.8a, the currently predicted intervals well encompass practical pool length variations, indicative of our experimental calibration proceeding in the correct direction. For the calibrated surrogate model by using the direct method, one can easily notice large errors in the estimations of the uncertainty source induced fluctuations; see Fig. 6.8a. Due to inaccurate inference about uncertainty source parameters, the calibrated surrogate model by using the direct method would subsequently suggest near-zero variability in melt pool length, which is clearly contradictory to the experimental facts.

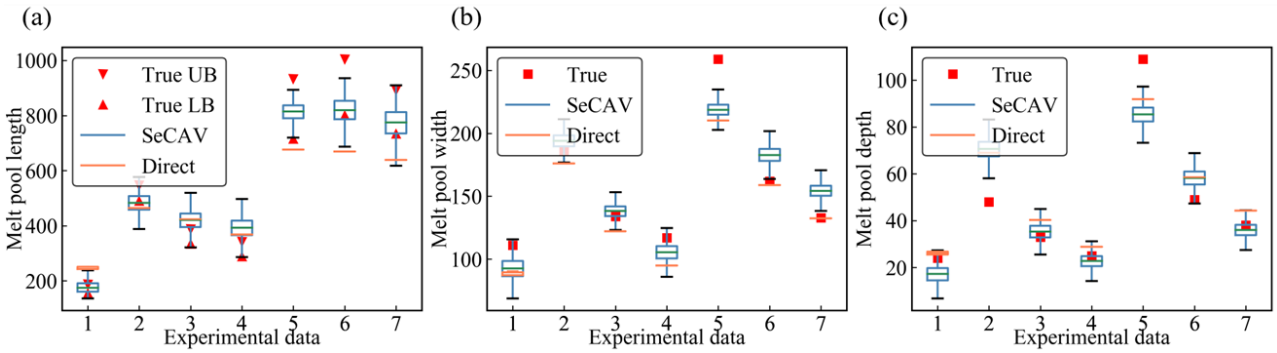

Figure 6.8 Cross-validation of the calibrated model in predicting mean and variability of melt pool.

Chapter 7 Conclusion

This dissertation has presented high-level data-driven modeling of AM process, structure, and property, which are enabled by advanced, maximum and flexible use of various machine learning techniques. Concluding remarks are made as below:

- Chapter 3 – Section 3.1: We leverage convolutional neural network to automatically denoise large amounts of melt pool images for high-quality data, and utilize multi-layer perceptron to build the relationship between scanning history and melt pool development at the current location. The proposed data-driven melt pool model can reduce predictive error by one order of magnitude compared to prior art.
- Chapter 3 – Section 3.2: We use singular value decomposition for dimension reduction of temperature field data, and then construct the temperature field development as a function of AM conditions. The proposed data-driven thermal model can accurately predict the 3D temperature field under various manufacturing conditions. It can replace the known Rosenthal analytical model for high-fidelity yet fast prediction of temperature field development.
- Chapter 4 – Section 4.1: We propose a novel multi-input convolutional neural network, yNet, to construct the relationship between laser conditions and porosity evolution in selective laser sintering. The proposed data-driven model is 5-6 orders of magnitude faster than original physics-based sintering model. It thus allows for extreme-scale component-level sintering simulation even with limited computational resource.
- Chapter 5 – Section 5.1: We further use the multi-input convolutional neural network for data-driven modeling of stress-field development for AM-fabricated structure.
- Chapter 5 – Section 5.2: Inspired by image captioning, we present a hybrid model (convolutional neural network + recurrent neural network) to accurately construct the relationship between structure and its stress-strain curve response.

- ❖ Chapter 6: We discuss and demonstrate the potential impact of this research on other data-driven modeling in AM and beyond. One of the proposed data-driven models, multi-input CNN, shows significant computational advantage (i.e., one-tenth of the model size) over conventional CNN. Also, we analyze the experimental calibration for data-driven model built on simulation data.

While this research shows the utilization of CNN to improve AM data through image denoising, there is much potential of ML for AM data improvement. For example, natural language processing (MLP) techniques have exhibit great capability in information extraction, for example, from literature (including tables, figures, and textual paragraphs) [157, 158] and medical records [159]. Similarly, NLP techniques may be used to collect useful AM data from their unstructured form in the voluminous AM literature and other digitalized resources. Also, generative adversarial network (GAN) [160] has been used for data augmentation by synthesizing experimental images of grain structures. All of those facts show the tremendous potential and versatility of ML in improving the quality and quantity of AM data in future.

Appendices

Appendix A Data-driven models

A.1 Multi-layer perceptron

Multilayer perceptron (MLP) is a primitive neural network. As shown in Fig. A.1, a MLP consists of multi-layer of fully connected neurons, each of which is a simple activation function. The signal flow within a neuron can be expressed as:

$$y = f\left(\sum w_i X_i + b\right) \tag{A.1}$$

where y is output of a neuron by receiving all input, X_i, from previous layer, and f represents activation function that can introduce great nonliearity into the MLP. Wi are weight and b is bias; they are updated during training to make MLP approximate the function.

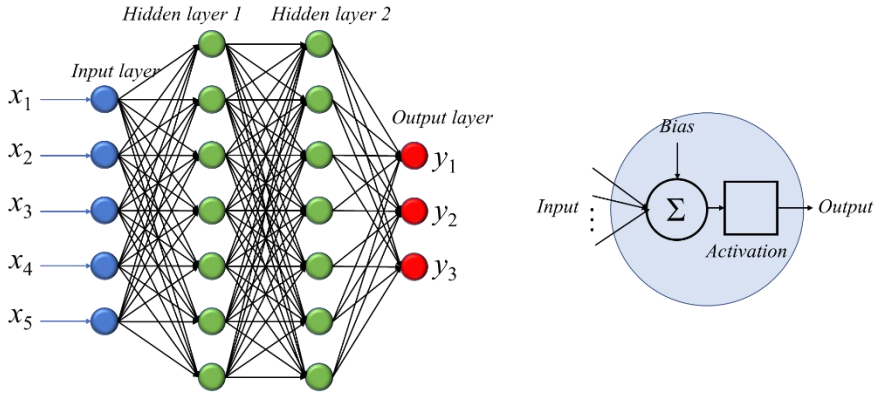

Figure A.1 Graphical illustration of multilayer perceptron and its neuron unit.

While the first and final layer are input and output respectively, one can tune the hidden layer by adjusting its number of hidden layers and the number of neurons contained in a hidden

layer. With enough layers and large number of neurons, a MLP can approximate any nonlinear and complex function, which is also known as universal approximation theorem [161] that supports the unparalleled regression capability of MLP. MLP might be the most used neural network for common regression analysis in various AM data-driven modeling.

Although MLP has exceptional regression capability and excels at simple regression tasks, it usually falls short when it comes to high-dimensional data like image. Because even a small image often contains tens of thousands of pixels. Imagine each pixel as an input to the MLP. It will make a MLP extremely parameter-intensive and very difficult to train.

A.2 Convolutional neural network

Convolutional neural network (CNN) is designed to make predictions by directly using image data as an input. More generally, CNNs perform well on array data that has a spatial relationship, because they can develop an internal representation of both one-dimensional sequence and two-dimensional matrix (e.g., images) by using stacks of convolutional layers to extract salient patterns or features.

Fig. A.2a shows the basic principle of convolutional layer. If one slides convolutional filter or kernel (3×3 herein) over the entire image (which is nothing but a pixel matrix), we can obtain

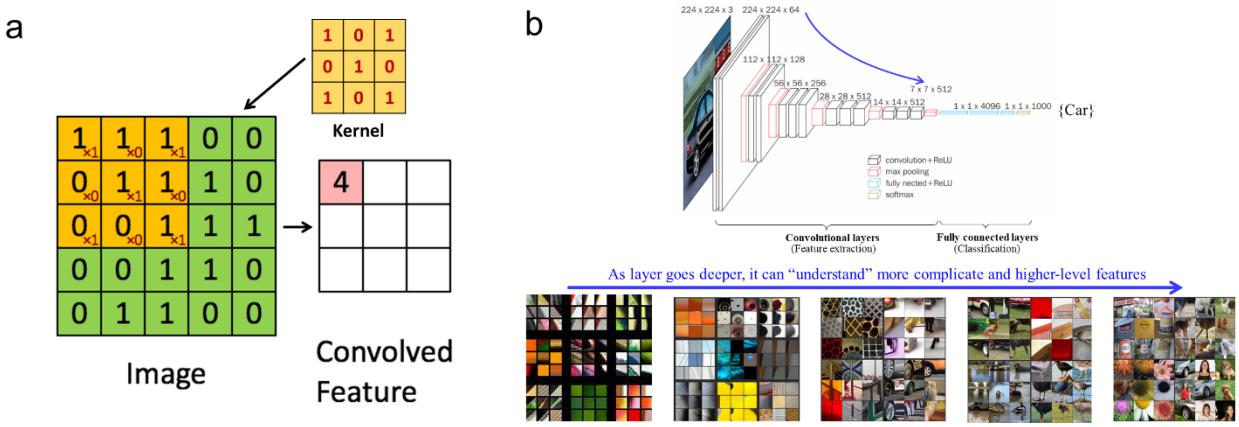

Figure A.2 Illustration of convolutional neural network. (a) Convolutional filter; (b) Deep with stacks of convolutional layers. (Images downloaded from: towardsdatascience.com)

convolved feature of input image. (For video illustration, readers are referred to: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53). Different kernel can be automatically tuned during training to recognize different meaningful features of objects. Fig. A.2b shows a typical classification-type CNN consisting of stacks of convolutional layers. As CNN goes deeper, it can extract increasingly abstracted and informative features of the input image that help identify the objects contained in the image [147]. The abstraction of original image into numbers of informative features by CNN can be considered as a non-linear dimension-reduction process. It would thus facilitate further operations on the image by dealing with its high-level form, instead of raw pixels.

During training of CNN, two types of loss functions are used in this study. For output field with strong texture (i.e, stress field), we use mean squared error (MSE) to evaluate loss:

$$Loss = \frac{1}{N}\sum_{i=1}^{N}\left|f\left(\boldsymbol{a}^i, X^i\right) - \bar{Y}^i\right|^2 \tag{A.2}$$

where N is number of samples processed, $\boldsymbol{a}$ is the condition-parameter inputs (e.g., laser power and speed in porosity simulation), X is the field input (e.g., powder structure before sintering), $f\left(\boldsymbol{a}, X\right)$ is the evolved field by prediction (e.g., powder structure after sintering) and $\bar{Y}$ is evolved field of ground truth. For output field that clearly displays two distinct phases (e.g., porosity structure), we use binary cross-entropy loss to better penalize predictive errors:

$$Loss = -\frac{1}{N}\sum_{i=1}^{N}\left|f\left(\boldsymbol{a}^i, X^i\right)\cdot\log\bar{Y}^i + \left(1 - f\left(\boldsymbol{a}^i, X^i\right)\right)\cdot\left(1 - \log\bar{Y}^i\right)\right|^2 \tag{A.3}$$

A.3 Gaussian process model

Gaussian process (GP) model is a "non-parametric" (i.e., without a rigid regression model) and more universal approximation method. The input-response relationship is formulated within covariance kernel. GP has the outstanding capability of measuring uncertainty on the prediction,

as GP essentially considers the whole data as a sample from a multivariate Gaussian distribution and therefore the prediction as a conditional distribution. GP would become painfully slow in face of large dataset, as whole samples or features information are used to perform the prediction. The GP model assumes the deterministic response surface is a realization of spatial-dependent stochastic process $y(\mathbf{x})$ with prior mean $\mathbf{h}^T(\mathbf{x})\boldsymbol{\beta}$, variance σ^2 and correlation function $R(\bullet,\bullet)$. The stochastic process can be formulated by separating the mean and covariance as

$$y(\mathbf{x}) = \mathbf{h}^T(\mathbf{x})\boldsymbol{\beta} + Z(\mathbf{x}) \tag{A.4}$$

where $\mathbf{h}^T(\mathbf{x})$ is a regression vector with some specific regression functions (e.g. constant, linear, etc), $\boldsymbol{\beta}$ is a vector of regression coefficients to be estimated while modelling, $Z(\mathbf{x})$ is a stochastic process with mean zero and covariance $\sigma^2 R(\bullet,\bullet)$. The most commonly used correlation function between the responses at point $\mathbf{x}$ and $\mathbf{x}'$ is given by

$$R(\mathbf{x},\mathbf{x}') = \exp\left\{-\sum_{i=1}^{d}\omega_i(x_i - x_i')^2\right\} \tag{A.5}$$

where d is the number of dimensions in the input $\mathbf{x}$, $\boldsymbol{\omega} = \left(\omega_1, \omega_2, ..., \omega_d\right)^T$ is a vector of the correlation related parameters that determine the correlation of the responses $y(\mathbf{x})$ and $y(\mathbf{x}')$. If the point $\mathbf{x}$ is close to point $\mathbf{x}'$, then the correlation between them will be large and the response $y(\mathbf{x})$ and $y(\mathbf{x}')$ should be similar. Based on the aforementioned definition, a GP model can be completely characterized by the hyper-parameters $\boldsymbol{\gamma} \triangleq \left(\boldsymbol{\beta}, \sigma^2, \boldsymbol{\omega}\right)$ and the predefined regression functions.

Given a group of training data $\left\{(\mathbf{x}_j, \mathbf{y}_j), j = 1,2,\ldots,N\right\}$, it is obvious that the response vector $\mathbf{y} = (\mathbf{y}_1, \mathbf{y}_2, .., \mathbf{y}_N)^T$ is normally distributed given $\boldsymbol{\gamma} \triangleq \left(\boldsymbol{\beta}, \sigma^2, \boldsymbol{\omega}\right)$ with a likelihood function. The mean vector and covariance matrix in the likelihood function are

$\left[\mathbf{h}^T(\mathbf{x}_1)\boldsymbol{\beta}, \mathbf{h}^T(\mathbf{x}_2)\boldsymbol{\beta}, \ldots \mathbf{h}^T(\mathbf{x}_N)\boldsymbol{\beta}\right]^T$ and $\sigma^2\mathbf{R}$ respectively, where $\mathbf{R}$ is a $N \times N$ correlation matrix with (*i, j*) element given by $R(\mathbf{x}_i, \mathbf{x}_j)$. By conducting Bayesian approach [162] or maximum likelihood estimation [163] for the multivariate normal likelihood function, the prior information will be updated (i.e. the hyper-parameters $\boldsymbol{\gamma} \triangleq \left(\boldsymbol{\beta}, \sigma^2, \boldsymbol{\omega}\right)$ will be estimated as $\hat{\boldsymbol{\gamma}} \triangleq \left(\hat{\boldsymbol{\beta}}, \hat{\sigma}^2, \hat{\boldsymbol{\omega}}\right)$) and the posterior mean and covariance of the GP model will be observed as

$$E\left[y(\mathbf{x}) \mid \mathbf{y}\right] = \mathbf{h}^T(\mathbf{x})\hat{\boldsymbol{\beta}} + \mathbf{r}^T(\mathbf{x})\mathbf{R}^{-1}(\mathbf{y} - \mathbf{H}\hat{\boldsymbol{\beta}}) \tag{A.6}$$

and

$$\begin{aligned} &Cov\left[y(\mathbf{x}), y(\mathbf{x}') \mid \mathbf{y}\right] = \\ &\hat{\sigma}^2 \left\{ R(\mathbf{x}, \mathbf{x}') - \mathbf{r}^T(\mathbf{x})\mathbf{R}^{-1}\mathbf{r}(\mathbf{x}') + \right. \\ &\left. \left[\mathbf{h}^T(\mathbf{x}) - \mathbf{H}^T\mathbf{R}^{-1}\mathbf{r}(\mathbf{x})\right]\left[\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\right]^{-1}\left[\mathbf{h}^T(\mathbf{x}') - \mathbf{H}^T\mathbf{R}^{-1}\mathbf{r}(\mathbf{x}')\right] \right\}, \end{aligned} \tag{A.7}$$

where $\mathbf{H} = \left[\mathbf{h}^T(\mathbf{x}_1), \mathbf{h}^T(\mathbf{x}_2), \ldots \mathbf{h}^T(\mathbf{x}_N)\right]^T$, $\mathbf{r}(\mathbf{x})$ is a $N \times 1$ vector with *i*-th element given by $R(\mathbf{x}, \mathbf{x}_i), i = 1, 2, \ldots, N$ and the estimated $\hat{\boldsymbol{\beta}} = \left[\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\right]^{-1}\mathbf{H}^T\mathbf{R}^{-1}\mathbf{y}$.

A.4 Recurrent neural network

Recurrent neural networks (RNNs) were designed to handle sequence-involved modeling problems, by permitting output from previous step to be fed to the current step during processing long sequences, as shown in . RNNs perform well on processing and predicting sequential data, e.g., words and spoken language in natural language processing (NLP) applications, as well as stress sequences data in the current research. Long short-term memory (LSTM) [164, 165] is one of the most widely used RNNs, and is selected in this research. The memory cell in LSTM can improve the modeling of long-term dependency in sequences. More specifically, LSTM contains a forget layer, f_t, to account for previous hidden state (as indicated by red arrow) and current input, and then output a value (0: completely forget, 1: retain information), which is expressed as:

$$f_t = \sigma\left(W_f * h_{t-1} + V_f * x_t + b_f\right) \tag{A.8}$$

next, LSTM has to decide the new information stored in the cell. The input gate layer, i_t, is first used to decide values updated and new information, g_t, will be then added to cell state:

$$i_t = \sigma\left(W_i * h_{t-1} + V_i * x_t + b_i\right) \tag{A.9}$$

$$g_t = \tanh\left(W_g * h_{t-1} + V_g * x_g + b_g\right) \tag{A.10}$$

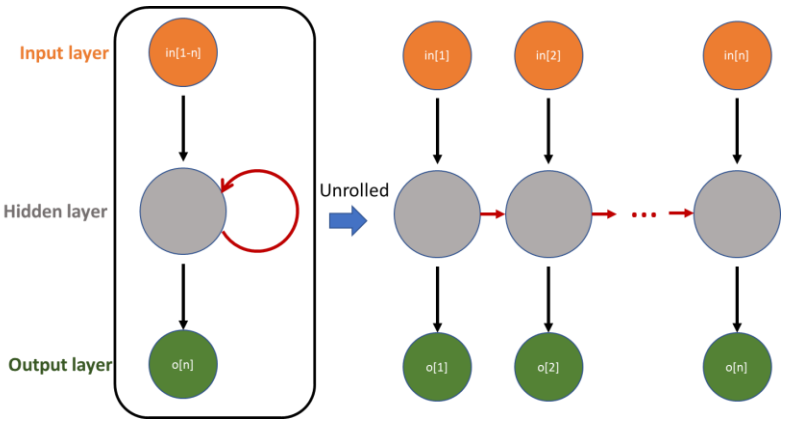

Figure A.3 Illustration of recurrent neural network (RNN) on processing a sequence with length of *n*.

where *f* is forget gate state, *i* is hidden state, *W* is weight matrix for previous to current hidden state, *b* is bias, *V* is weight matrix for input to hidden state. The memory cell can thus help model the stress-strain curve as a whole, which is contrary to previous step-wise prediction, where prediction at each strain step is in fact completely separate. The looping structure of RNNs for dealing with sequences can increase training complexity due to the more complicated signal movements.

Appendix B Physical models

B.1 Thermal model

Physics-based thermal model is used to generate temperature field data in data-driven modeling of temperature field. In this study, a FE-based heat transfer model incorporating a moving heat source is utilized to predict the in-process temperature distribution [21], which are then extracted as HF data. Specifically, the tempo-spatial evolution of temperature field with external heat input is governed by

$$\nabla \cdot \left(k \nabla T \right) + \rho Q_e = \frac{\partial \left(\rho c_p T \right)}{\partial t} \tag{B.1}$$

where $T(x, y, z, t)$ is the tempo-spatial temperature field, k is thermal conductivity, ρ is density and $c_p$is specific heat. Of special importance is the external heat input from the moving electron beam, Q_e, which is described using the Gaussian distribution equation as the following [39]

$$\begin{cases} T_e(x,y,z,t) = \eta P \dfrac{4\ln(0.1)}{\pi d^2 z_e} \exp\left(\dfrac{4\ln(0.1)\left((x - v \cdot t)^2 + y^2 \right)}{d^2} \right) \left\{ -3\left(\dfrac{z}{z_e} \right)^2 - 2\left(\dfrac{z}{z_e} \right) + 1 \right\} \\ z_e = 2.1 \times 10^{-5} \times \dfrac{V_e^2}{\rho} \end{cases} \tag{B.2}$$

where η is the power absorption efficiency of powder bed, P is the nominal power of electron beam, d is the diameter of electron beam, v is the beam velocity and z_e is the absolute penetration depth of electron beam associated with the acceleration voltage, V_e.

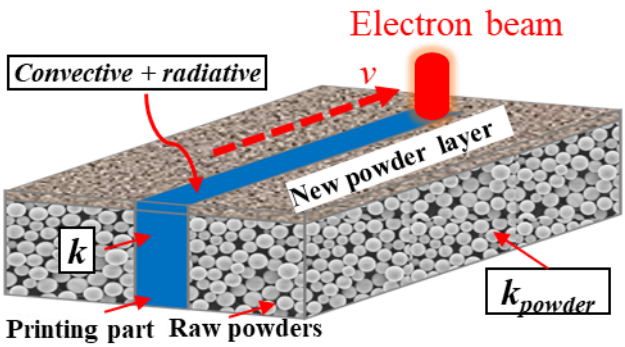

Figure B.1 Illustration of thermal model of electron beam melting (EBM)

The boundary conditions of the FE-based thermal model, as graphically illustrated in Fig. B.1, are summarized as follows: 1) the initial boundary condition of substrate and deposits is $T = T_{pre}$, where T_{pre} is the preheating temperature; 2) the types of heat transfers at the surface are convective and radiative; 3) in view of the limited thermal conductivity of loose powders, adiabatic conditions are imposed on the sides of the printing part, thus with raw powders physically ignored in the simulation; 4) evaporation is not incorporated in the FE-based thermal model, as normally did in AM thermal models. The above FE-based thermal model is realized in ABAQUS 6.10 using custom DFLUX user subroutine [166]. The extraction of temperature field (upon reaching steady state) as training data is automated by Python script.

B.2 Multi-physical sintering model

Physics-based SLS simulation provides training and testing data in data-driven modeling of porosity development in SLS. In this study, we basically extend the primitive phase-field-based sintering model [167] for applicability in SLS, by incorporating heat transfer model and a Gaussian heat source model describing the effective heat input from moving laser beam [168]. The sintering model itself has properly taken into account multiple physical processes (see Fig. B.2), by reformulating the effective diffusion coefficient as:

$$D^{eff} = D_{vol}\phi(\rho) + D_{vap}\left[1-\phi(\rho)\right] + D_{sf}\rho(1-\rho) + D_{gb}\sum_{\alpha}\sum_{\alpha'\neq\alpha}(\eta_\alpha\eta_{\alpha'}) \quad \text{(B.3)}$$

where D_{vol}, D_{vap}, D_{sf} and D_{gb} are temperature-dependent diffusion coefficients in solid volume, vapor, along surface, and grain boundary, respectively.

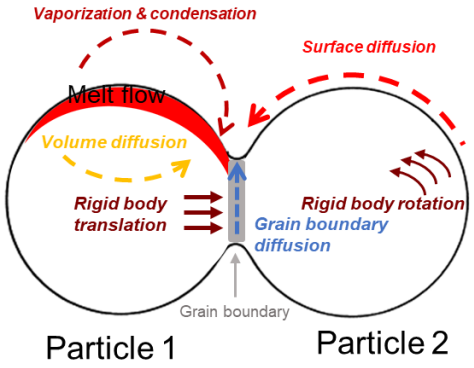

Figure B.2 Various physical processes considered in the multi-physical sintering model.

The alloy we used is stainless steel 316L. The spatial and temporal simulation resolution are $\Delta x = \Delta y = 2\mu m$ and $\Delta t = 1\mu s$, respectively. More details of the adopted multi-physical SLS model can be found in [169].

B.3 Powder bed generation model

During sintering simulation, generation of powder bed (i.e. initial porous structure) is simulated using a "rain" model **[23]**. As shown in Fig. B.3, the basic principle of rain model is to

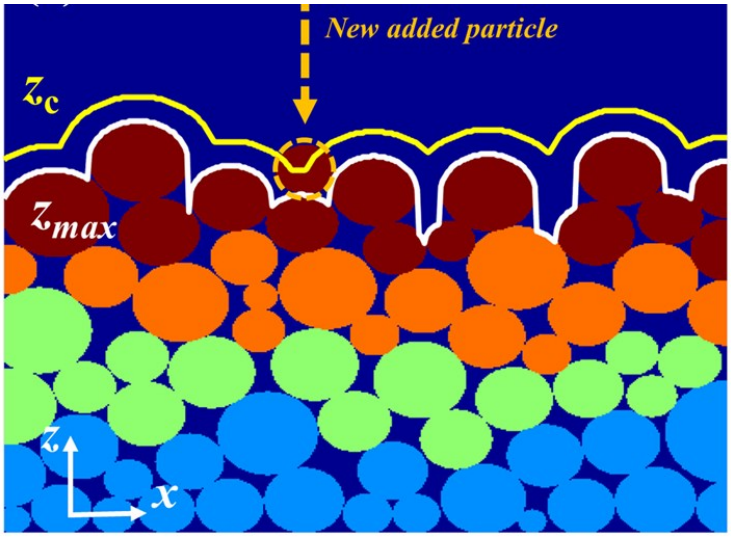

Figure B.3 Illustration of rain model for powder bed generation modeling.

add a powder particle to the lowest position on the surface of the current powder layer, update the powder layer and repeat the above two steps. To find the lowest position, the trajectory of the newly added powder along the surface (yellow curve), z_c, can be calculated by:

$$z_c(x) = \max\left[z_{\max}(x) + \sqrt{r_c^2 - (x - x_c)^2} \right]\Big|_{\sqrt{(x-x_c)^2} \le r_c} \tag{B.4}$$

where r_c is the raius of newly added powder, and x_c is the center of newly added powder in horizontal direction.

This iterative process is terminated when the newly added particle reaches the user-defined layer height. The mean and standard deviation of the diameter of deposited powders are 25μm and 0.5μm, respectively. It should be pointed out that CNN in this study will be trained to simulate porosity evolution and thus replace physics-based sintering model only. Powder bed generation model is a separate model that provides initial structure for physical sintering simulation and CNN-based sintering simulation.

Appendix C Dara-driven modeling of porosity evolution

C.1 Partition of sampling points in data-driven porosity modeling

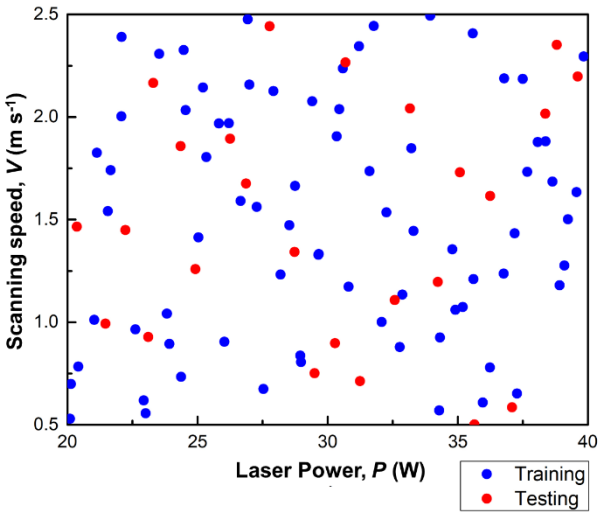

Figure C.1 Partition of [*P*, *V*] conditions for training and testing of yNet. 75 out of 100 sampling points are randomly selected for training and the remaining 25 for testing.

C.2 Workflow of multi-layer sintering simulation

As an illustrative example, we use laser power of P = 20 W, scanning speed of V = 0.5 m s^{-1}, track length of l = 640 pixels (or 1280 μm), and layer thickness of t = 70 pixels (or 140 μm). Considering the high computational cost of physics-based simulation, we perform sintering simulation for 3 layers in total, as shown in Fig. C.2. Following the same exact workflow, we additionally perform the multi-layer simulation at P =25, and 30 W, and then get the sintering effect-power map in Fig. 4.5 in Subsection 4.1.3.

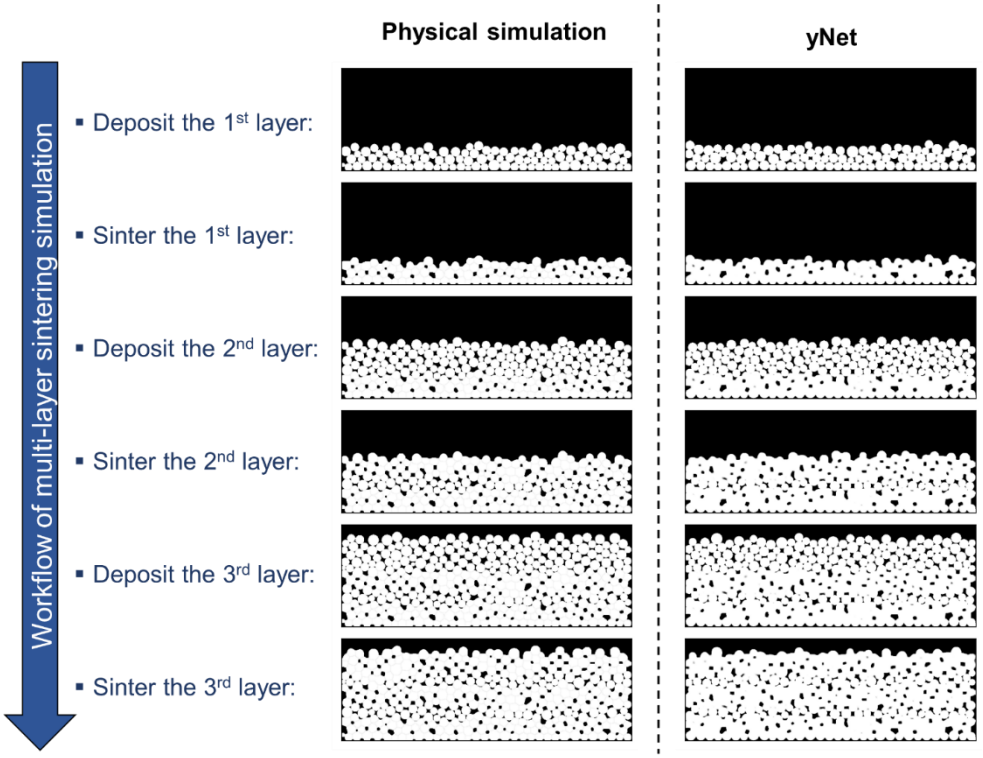

Figure C.2 Workflow of multi-layer sintering simulation based on physical model and the trained multi-input CNN, yNet.

C.3 Workflow of yNet-based layer-by-layer component-level SLS simulation

As shown in Fig. C.3, the full-component simulation is performed in a layer-by-layer manner by following the AM practice, i.e., alternating powder deposition and selective sintering until completion. The selective sintering of each newly deposited layer is guided by sliced CAD model, as well as AM command file containing information of applied laser conditions. For illustration, constant laser power and scanning speed of 40 W and 0.5 m s^{-1} are used in the current case. Note that, for some layers, there are more than one regions to be sintered, e.g., the two regions of interest in the first layer as shown, in which case yNet need to predict twice.

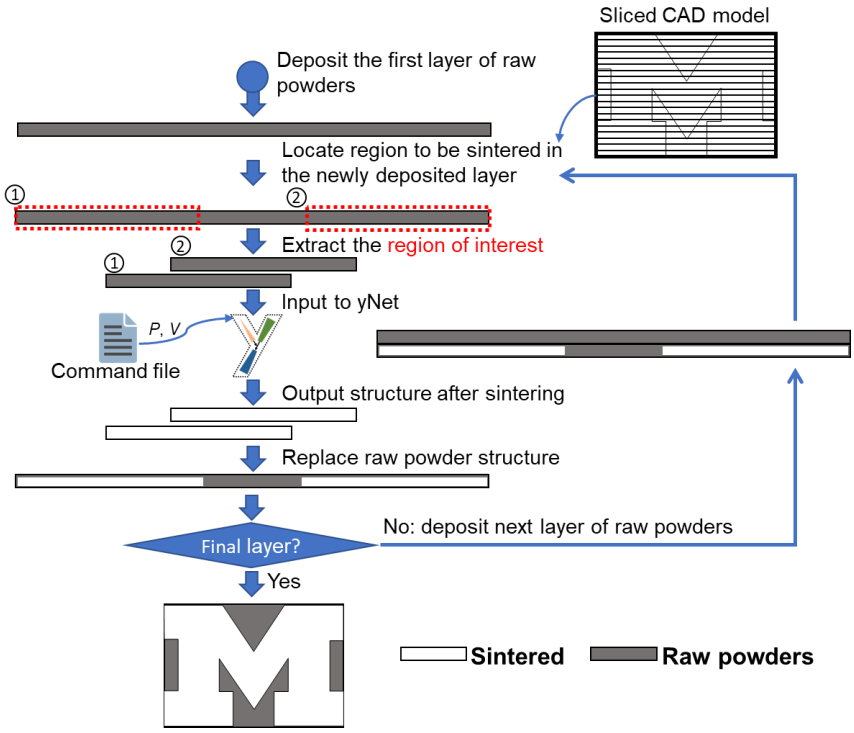

Figure C.3 Workflow of layer-by-layer component-level SLS simulation based on the trained yNet.

Appendix D Experimental calibration

The calibration method that follows is published in [170, 171].

D.1 Conventional Bayesian and SeCAV calibration

The initial melt pool surrogate models based on simulation data require experimental calibration, i.e., the calibration of the uncertainty source parameters and bias correction. We employ $y_s(\mathbf{x},\boldsymbol{\theta})$ and $y_e(\mathbf{x})$ to represent the AM simulation model or the simulation-derived surrogate model prediction and the corresponding experimental observation for the QoI, y. They are simply related by

$$y_e(\mathbf{x}) = y_s(\mathbf{x},\boldsymbol{\theta}) + \delta(\mathbf{x}) \tag{D.1}$$

where $\delta(\mathbf{x})$ is the model bias between the simulation model/surrogate model prediction and the experimental measurement, $\mathbf{x}$ represent controllable variables that can be precisely controlled during experiments (e.g. laser power and scanning speed), and $\boldsymbol{\theta}$ are the uncertainty source parameters to be calibrated (see Table 1). The $\boldsymbol{\theta}$ parameters are distribution-type in nature, and so is $y_e(\mathbf{x})$ by uncertainty propagation. The accurate distributions of $\boldsymbol{\theta}$ are unknown and thus require inverse inference from available experimental data $y_e(\mathbf{x})$. To confidently predict QoI and its uncertainty for any given $\mathbf{x}$, we will perform the calibration of uncertain parameters $\boldsymbol{\theta}$ and determine the model bias $\delta(\mathbf{x})$ to systematically reduce the model uncertainty.

We conducted the calibration of uncertain parameters employing Bayesian calibration [172], which is one of the most popular methods for statistical inference. The conventional Bayesian calibration yields a quantitative update of uncertainty based on probability theory.

Solving the Bayesian problem involves computing the posterior probability density function (PDF) of uncertain parameters $p\left(\boldsymbol{\theta}|\mathbf{y}_e\right)$ with the knowledge of the experimental data (i.e. $\mathbf{y}_e$) of QoI:

$$\overbrace{p\left(\boldsymbol{\theta}|\mathbf{y}_e\right)}^{posterior}=\frac{\overbrace{p\left(\mathbf{y}_e|\boldsymbol{\theta}\right)}^{likelihood}\overbrace{p(\boldsymbol{\theta})}^{prior}}{\underbrace{p\left(\mathbf{y}_e\right)}_{evidence}}\propto p\left(\mathbf{y}_e|\boldsymbol{\theta}\right)p(\boldsymbol{\theta}) \tag{D.2}$$

where $p\left(\mathbf{y}_e|\boldsymbol{\theta}\right)$ represents the likelihood or probability of observing $\mathbf{y}_e$ for the AM simulation model $y_s(\mathbf{x},\boldsymbol{\theta})$ with respect to different specific values of $\boldsymbol{\theta}$. $p\left(\boldsymbol{\theta}\right)$ is the prior PDF of uncertain parameters (i.e. before observing data $\mathbf{y}_e$), and the $p\left(\mathbf{y}_e\right)$ is the probability for the given evidence $\mathbf{y}_e$ and is usually fixed. There are different methods for computing $p\left(\boldsymbol{\theta}|\mathbf{y}_e\right)$. Here we employ the particle filter [173] involving the prior sampling, likelihood weighting, and posterior resampling. It should be pointed out that the conventional Bayesian calibration ignores the discrepancy (i.e. $\delta(\mathbf{x})$ in Eq. (D.1) between the simulation model and the reality. To address this discrepancy, we further adopt a novel SeCAV method that implements the physics-based model validation and Bayesian calibration by using each of experimental data (i.e. $\mathbf{y}_e$) of QoI in a sequential manner. In each iteration, the physics-based model validation is first conducted at each experimental data point to answer the question: to what degree can we trust the likelihood $p\left(\mathbf{y}_e|\boldsymbol{\theta}\right)$ and the Bayesian calibration result obtained based on $p\left(\mathbf{y}_e|\boldsymbol{\theta}\right)$. In other words, the physics-based model validation serves as a filter to select the experimental data with the maximum confidence probability for the subsequent Bayesian calibration. The confidence probability confidence probability to accept the hypothesis that the GP model is credible at data $(\mathbf{x}_e^i, y_e^j(\mathbf{x}_e^i))$ can be defined by

$$P_\lambda\left(\mathbf{x}_e^i, y_e^j(\mathbf{x}_e^i)\right)|\boldsymbol{\theta}=\Pr\left\{\varepsilon_r\left(\mathbf{x}_e^i, y_e^j(\mathbf{x}_e^i)\right)|\boldsymbol{\theta}\le\varepsilon_r^c\right\} \tag{D.3}$$

where $\varepsilon_r\left(\mathbf{x}_e^i, y_e^j(\mathbf{x}_e^i)\right)\big|\boldsymbol{\theta}$ is an uncertain variable due to the uncertainty in the surrogate model prediction $\hat{y}_s^j(\mathbf{x}_e^i, \boldsymbol{\theta})$ and ε_r^c is the confident threshold that is used to determine whether the agreement is acceptable or not. By integrating $P_\lambda\left(\mathbf{x}_e^i, y_e^j(\mathbf{x}_e^i)\right)\big|\boldsymbol{\theta}$ with respect to $\boldsymbol{\theta}$, we can obtain the unconditioned confidence probability as

$$P_\lambda\left(\mathbf{x}_e^i, y_e^j(\mathbf{x}_e^i)\right) = \int\left[P_\lambda\left(\mathbf{x}_e^i, y_e^j(\mathbf{x}_e^i)\right)\big|\boldsymbol{\theta}\right] f(\boldsymbol{\theta})\mathbf{d}\boldsymbol{\theta} \tag{D.4}$$

where $f(\boldsymbol{\theta})$ is the joint prior distribution of $\boldsymbol{\theta}$. A larger confidence probability means a more trust-worthy calibration result, and thus a larger weight will be given to the Bayesian calibration in weighted averaging. Otherwise, we retain the calibration result from the previous iteration. Such a weighted averaging in a sequentially updating manner from a larger confidence to a lower confidence enables a great trade-off between the accuracy and uncertainty reduction.

We further correct the bias between the melt pool model and true physics by building another GP model for the bias. The training data, i.e. $\left(\mathbf{x}, \delta(\mathbf{x})\right)$, are observed by fixing uncertain parameters $\boldsymbol{\theta}$ at the values with the maximum posterior PDF. We finally achieved a reliable and accurate posterior prediction of melt pool surrogate model by integrating the simulation-derived surrogate model with respect to the posterior distribution of uncertain parameters and by adding the prediction of the bias surrogate model.

Bibliography

[1] Wang, R.-J., Li, J., Wang, F., Li, X., and Wu, Q., 2009, "ANN model for the prediction of density in selective laser sintering," International Journal of Manufacturing Research, 4(3), pp. 362-373.

[2] Wang, Y., Liu, Y., Ciobanu, C., and Patton, B. R., 2000, "Simulating microstructural evolution and electrical transport in ceramic gas sensors," Journal of the American Ceramic Society, 83(9), pp. 2219-2226.

[3] Ji, Y.-Z., Wang, Z., Wang, B., Chen, Y., Zhang, T., Chen, L.-Q., Song, X., and Chen, L., 2017, "Effect of Meso-Scale Geometry on Piezoelectric Performances of Additively Manufactured Flexible Polymer-Pb(ZrxTi1−x)O3 Composites " Advanced Engineering Materials, 19(6), p. 1600803.

[4] Song, X., He, L., Yang, W., Wang, Z., Chen, Z., Guo, J., Wang, H., and Chen, L., 2019, "Additive Manufacturing of Bi-Continuous Piezocomposites With Triply Periodic Phase Interfaces for Combined Flexibility and Piezoelectricity," Journal of Manufacturing Science and Engineering, 141(11).

[5] Monzón, M., Ortega, Z., Martínez, A., and Ortega, F., 2015, "Standardization in additive manufacturing: activities carried out by international organizations and projects," The International Journal of Advanced Manufacturing Technology, 76(5-8), pp. 1111-1121.

[6] Xiao, Y., Cagle, M., Mujahid, S., Liu, P., Wang, Z., Yang, W., and Chen, L., 2021, "A gleeble-assisted study of phase evolution of Ti-6Al-4V induced by thermal cycles during additive manufacturing," Journal of Alloys and Compounds, 860, p. 158409.

[7] Antonysamy, A. A., Meyer, J., and Prangnell, P., 2013, "Effect of build geometry on the β-grain structure and texture in additive manufacture of Ti 6Al 4V by selective electron beam melting," Materials Characterization, 84, pp. 153-168.

[8] Liu, P., Ji, Y., Wang, Z., Qiu, C., Antonysamy, A., Chen, L.-Q., Cui, X., and Chen, L., 2018, "Investigation on evolution mechanisms of site-specific grain structures during metal additive manufacturing," Journal of Materials Processing Technology, 257, pp. 191-202.

[9] Carroll, B. E., Palmer, T. A., and Beese, A. M., 2015, "Anisotropic tensile behavior of Ti–6Al–4V components fabricated with directed energy deposition additive manufacturing," Acta Materialia, 87, pp. 309-320.

[10] Tang, M., Pistorius, P. C., and Beuth, J. L., 2017, "Prediction of lack-of-fusion porosity for powder bed fusion," Additive Manufacturing, 14, pp. 39-48.

[11] Yang, W., Wang, Z., Yenusah, C., and Liu, Y., 2020, "An Integrated Model for Prediction of Process-Structure-Property Relationship for Additively Manufactured Al-10Si-Mg Alloy," No. 0148-7191, SAE Technical Paper.

[12] Qiu, C., Panwisawas, C., Ward, M., Basoalto, H. C., Brooks, J. W., and Attallah, M. M., 2015, "On the role of melt flow into the surface structure and porosity development during selective laser melting," Acta Materialia, 96, pp. 72-79.

[13] Qiu, C., Wang, Z., Aladawi, A. S., Al Kindi, M., Al Hatmi, I., Chen, H., and Chen, L., 2019, "Influence of laser processing strategy and remelting on surface structure and porosity development during selective laser melting of a metallic material," Metallurgical and Materials Transactions A, 50(9), pp. 4423-4434.

[14] King, W. E., Barth, H. D., Castillo, V. M., Gallegos, G. F., Gibbs, J. W., Hahn, D. E., Kamath, C., and Rubenchik, A. M., 2014, "Observation of keyhole-mode laser melting in laser powder-bed fusion additive manufacturing," Journal of Materials Processing Technology, 214(12), pp. 2915-2925.

[15] Olakanmi, E., Cochrane, R., and Dalgarno, K., 2011, "Densification mechanism and microstructural evolution in selective laser sintering of Al–12Si powders," Journal of Materials Processing Technology, 211(1), pp. 113-121.

[16] Tian, Q., Guo, S., Melder, E., Bian, L., and Guo, W., 2021, "Deep Learning-Based Data Fusion Method for In Situ Porosity Detection in Laser-Based Additive Manufacturing," Journal of Manufacturing Science and Engineering, 143(4).

[17] Zhao, C., Fezzaa, K., Cunningham, R. W., Wen, H., De Carlo, F., Chen, L., Rollett, A. D., and Sun, T., 2017, "Real-time monitoring of laser powder bed fusion process using high-speed X-ray imaging and diffraction," Scientific Reports, 7(1), pp. 1-11.

[18] Al-Bermani, S., Blackmore, M., Zhang, W., and Todd, I., 2010, "The origin of microstructural diversity, texture, and mechanical properties in electron beam melted Ti-6Al-4V," Metallurgical and Materials Transactions A, 41(13), pp. 3422-3434.

[19] Simchi, A., 2006, "Direct laser sintering of metal powders: Mechanism, kinetics and microstructural features," Materials Science and Engineering: A, 428(1-2), pp. 148-158.

[20] Xie, F., He, X., Cao, S., and Qu, X., 2013, "Structural and mechanical characteristics of porous 316L stainless steel fabricated by indirect selective laser sintering," Journal of Materials Processing Technology, 213(6), pp. 838-843.

[21] Francois, M. M., Sun, A., King, W. E., Henson, N. J., Tourret, D., Bronkhorst, C. A., Carlson, N. N., Newman, C. K., Haut, T. S., and Bakosi, J., 2017, "Modeling of additive manufacturing processes for metals: Challenges and opportunities," Current Opinion in Solid State and Materials Science, 21(LA-UR-16-24513; SAND-2017-6832J).

[22] Liu, P., Wang, Z., Xiao, Y., Lebensohn, R. A., Liu, Y., Horstemeyer, M. F., Cui, X., and Chen, L., 2020, "Integration of phase-field model and crystal plasticity for the prediction of process-structure-property relation of additively manufactured metallic materials," International Journal of Plasticity, 128, p. 102670.

[23] Lu, L.-X., Sridhar, N., and Zhang, Y.-W., 2018, "Phase field simulation of powder bed-based additive manufacturing," Acta Materialia, 144, pp. 801-809.

[24] Khairallah, S. A., Anderson, A. T., Rubenchik, A., and King, W. E., 2016, "Laser powder-bed fusion additive manufacturing: physics of complex melt flow and formation mechanisms of pores, spatter, and denudation zones," Acta Materialia, 108, pp. 36-45.

[25] Zinoviev, A., Zinovieva, O., Ploshikhin, V., Romanova, V., and Balokhonov, R., 2016, "Evolution of grain structure during laser additive manufacturing. Simulation by a cellular automata method," Materials & Design, 106, pp. 321-329.

[26] Geng, Y., and Harrison, N., 2020, "Functionally graded bimodal Ti6Al4V fabricated by powder bed fusion additive manufacturing: Crystal plasticity finite element modelling," Materials Science and Engineering: A, 773, p. 138736.

[27] Mani, M., Feng, S., Lane, B., Donmez, A., Moylan, S., and Fesperman, R., 2015, Measurement science needs for real-time control of additive manufacturing powder bed fusion processes, US Department of Commerce, National Institute of Standards and Technology.

[28] Yadroitsev, I., 2009, Selective laser melting: Direct manufacturing of 3D-objects by selective laser melting of metal powders, LAP Lambert Academic Publishing.

[29] Bertoli, U. S., Wolfer, A. J., Matthews, M. J., Delplanque, J.-P. R., and Schoenung, J. M., 2017, "On the limitations of volumetric energy density as a design parameter for selective laser melting," Materials & Design, 113, pp. 331-340.

[30] Hey, T., Tansley, S., and Tolle, K. M., 2009, "Jim Gray on eScience: a transformed scientific method."

[31] Bell, G., Hey, T., and Szalay, A., 2009, "Beyond the data deluge," Science, 323(5919), pp. 1297-1298.

[32] Haenlein, M., and Kaplan, A., 2019, "A brief history of artificial intelligence: On the past, present, and future of artificial intelligence," California Management Review, 61(4), pp. 5-14.

[33] Brunton, S. L., Noack, B. R., and Koumoutsakos, P., 2020, "Machine learning for fluid mechanics," Annual Review of Fluid Mechanics, 52, pp. 477-508.

[34] Brunton, S. L., and Kutz, J. N., 2019, Data-driven science and engineering: Machine learning, dynamical systems, and control, Cambridge University Press.

[35] Schlexer Lamoureux, P., Winther, K. T., Garrido Torres, J. A., Streibel, V., Zhao, M., Bajdich, M., Abild‐Pedersen, F., and Bligaard, T., 2019, "Machine learning for computational heterogeneous catalysis," ChemCatChem, 11(16), pp. 3581-3601.

[36] Han, J., Jentzen, A., and Weinan, E., 2018, "Solving high-dimensional partial differential equations using deep learning," Proceedings of the National Academy of Sciences, 115(34), pp. 8505-8510.

[37] Bergen, K. J., Johnson, P. A., Maarten, V., and Beroza, G. C., 2019, "Machine learning for data-driven discovery in solid Earth geoscience," Science, 363(6433).

[38] Ramprasad, R., Batra, R., Pilania, G., Mannodi-Kanakkithodi, A., and Kim, C., 2017, "Machine learning in materials informatics: recent applications and prospects," npj Computational Materials, 3(1), p. 54.

[39] Li, Y., Liu, K., Foley, A. M., Zülke, A., Berecibar, M., Nanini-Maury, E., Van Mierlo, J., and Hoster, H. E., 2019, "Data-driven health estimation and lifetime prediction of lithium-ion batteries: A review," Renewable and sustainable energy reviews, 113, p. 109254.

[40] Bock, F. E., Aydin, R. C., Cyron, C. J., Huber, N., Kalidindi, S. R., and Klusemann, B., 2019, "A review of the application of machine learning and data mining approaches in continuum materials mechanics," Frontiers in Materials, 6, p. 110.

[41] Wang, J., Ma, Y., Zhang, L., Gao, R. X., and Wu, D., 2018, "Deep learning for smart manufacturing: Methods and applications," Journal of manufacturing systems, 48, pp. 144-156.

[42] Xiong, Y., Duong, P. L. T., Wang, D., Park, S.-I., Ge, Q., Raghavan, N., and Rosen, D. W., 2019, "Data-driven design space exploration and exploitation for design for additive manufacturing," Journal of Mechanical Design, 141(10).

[43] Jiang, J., Xiong, Y., Zhang, Z., and Rosen, D. W., 2020, "Machine learning integrated design for additive manufacturing," Journal of Intelligent Manufacturing, pp. 1-14.

[44] Mukherjee, T., and DebRoy, T., 2019, "A digital twin for rapid qualification of 3D printed metallic components," Applied Materials Today, 14, pp. 59-65.

[45] Zhang, L., Chen, X., Zhou, W., Cheng, T., Chen, L., Guo, Z., Han, B., and Lu, L., 2020, "Digital Twins for Additive Manufacturing: A State-of-the-Art Review," Applied Sciences, 10(23), p. 8350.

[46] Majeed, A., Zhang, Y., Ren, S., Lv, J., Peng, T., Waqar, S., and Yin, E., 2021, "A big data-driven framework for sustainable and smart additive manufacturing," Robotics and Computer-Integrated Manufacturing, 67, p. 102026.

[47] Lehmhus, D., Wuest, T., Wellsandt, S., Bosse, S., Kaihara, T., Thoben, K.-D., and Busse, M., 2015, "Cloud-based automated design and additive manufacturing: a usage data-enabled paradigm shift," Sensors, 15(12), pp. 32079-32122.

[48] Wang, G. G., and Shan, S., 2007, "Review of metamodeling techniques in support of engineering design optimization,".

[49] Zhang, W., Mehta, A., Desai, P. S., and III, C. F. H., 2017, "MACHINE LEARNING ENABLED POWDER SPREADING PROCESS MAP FOR METAL ADDITIVE MANUFACTURING (AM)," 2017 Solid Freeform Fabrication Symposium Proceedings.

[50] Tapia, G., Khairallah, S., Matthews, M., King, W. E., and Elwany, A., 2018, "Gaussian process-based surrogate modeling framework for process planning in laser powder-bed fusion additive manufacturing of 316L stainless steel," The International Journal of Advanced Manufacturing Technology, 94(9-12), pp. 3591-3603.

[51] Paul, A., Mozaffar, M., Yang, Z., Liao, W.-k., Choudhary, A., Cao, J., and Agrawal, A., 2019, "A real-time iterative machine learning approach for temperature profile prediction in additive manufacturing processes," 2019 IEEE International Conference on Data Science and Advanced Analytics (DSAA), IEEE, pp. 541-550.

[52] Gan, Z., Li, H., Wolff, S. J., Bennett, J. L., Hyatt, G., Wagner, G. J., Cao, J., and Liu, W. K., 2019, "Data-Driven Microstructure and Microhardness Design in Additive Manufacturing Using a Self-Organizing Map," Engineering, 5(4), pp. 730-735.

[53] Wang, Z., Liu, P., Xiao, Y., Cui, X., Hu, Z., and Chen, L., 2019, "A Data-driven Approach for Process Optimization of Metallic Additive Manufacturing under Uncertainty," Journal of Manufacturing Science and Engineering, 141(8), p. 081004.

[54] Popova, E., Rodgers, T. M., Gong, X., Cecen, A., Madison, J. D., and Kalidindi, S. R., 2017, "Process-structure linkages using a data science approach: application to simulated additive manufacturing data," Integrating Materials and Manufacturing Innovation, 6(1), pp. 54-68.

[55] Gupta, A., Cecen, A., Goyal, S., Singh, A. K., and Kalidindi, S. R., 2015, "Structure–property linkages using a data science approach: application to a non-metallic inclusion/steel composite system," Acta Materialia, 91, pp. 239-254.

[56] Garg, A., Tai, K., and Savalani, M., 2014, "State-of-the-art in empirical modelling of rapid prototyping processes," Rapid Prototyping Journal.

[57] Herriott, C., and Spear, A. D., 2020, "Predicting microstructure-dependent mechanical properties in additively manufactured metals with machine-and deep-learning methods," Computational Materials Science, 175, p. 109599.

[58] Thuerey, N., Weißenow, K., Prantl, L., and Hu, X., 2020, "Deep learning methods for Reynolds-averaged Navier–Stokes simulations of airfoil flows," AIAA Journal, 58(1), pp. 25-36.

[59] Kashefi, A., Rempe, D., and Guibas, L. J., 2020, "A Point-Cloud Deep Learning Framework for Prediction of Fluid Flow Fields on Irregular Geometries," arXiv preprint arXiv:2010.09469.

[60] Nie, Z., Jiang, H., and Kara, L. B., 2019, "Stress Field Prediction in Cantilevered Structures Using Convolutional Neural Networks," Journal of Computing and Information Science in Engineering, 20(1).

[61] Jiang, H., Nie, Z., Yeo, R., Farimani, A. B., and Kara, L. B., 2021, "StressGAN: A Generative Deep Learning Model for Two-Dimensional Stress Distribution Prediction," Journal of Applied Mechanics, 88(5), p. 051005.

[62] Wei, H., Mazumder, J., and DebRoy, T., 2015, "Evolution of solidification texture during additive manufacturing," Scientific Reports, 5.

[63] Kamath, C., 2016, "Data mining and statistical inference in selective laser melting," The International Journal of Advanced Manufacturing Technology, 86(5), pp. 1659-1677.

[64] Meng, L., and Zhang, J., 2020, "Process Design of Laser Powder Bed Fusion of Stainless Steel Using a Gaussian Process-Based Machine Learning Model," JOM Journal of the Minerals Metals and Materials Society, 72(1), pp. 420-428.

[65] Lee, S., Peng, J., Shin, D., and Choi, Y. S., 2019, "Data analytics approach for melt-pool geometries in metal additive manufacturing," Science and Technology of Advanced Materials, 20(1), pp. 972-978.

[66] Yang, Z., Lu, Y., Yeung, H., and Krishnamurty, S., 2020, "From scan strategy to melt pool prediction: A neighboring-effect modeling method," Journal of Computing and Information Science in Engineering, 20(5).

[67] LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D., 1989, "Backpropagation applied to handwritten zip code recognition," Neural Computation, 1(4), pp. 541-551.

[68] LeCun, Y., Bengio, Y., and Hinton, G., 2015, "Deep learning," Nature, 521(7553), pp. 436-444.

[69] Chen, W., Allen, J. K., Tsui, K.-L., and Mistree, F., 1996, "A procedure for robust design: minimizing variations caused by noise factors and control factors," Journal of Mechanical Design, 118(4), pp. 478-485.

[70] Sahoo, S., and Chou, K., 2016, "Phase-field simulation of microstructure evolution of Ti–6Al–4V in electron beam additive manufacturing process," Additive Manufacturing, 9, pp. 14-24.

[71] Nie, P., Ojo, O., and Li, Z., 2014, "Numerical modeling of microstructure evolution during laser additive manufacturing of a nickel-based superalloy," Acta Materialia, 77, pp. 85-95.

[72] Raghavan, N., Dehoff, R., Pannala, S., Simunovic, S., Kirka, M., Turner, J., Carlson, N., and Babu, S. S., 2016, "Numerical modeling of heat-transfer and the influence of process parameters on tailoring the grain morphology of IN718 in electron beam additive manufacturing," Acta Materialia, 112, pp. 303-314.

[73] Liu, P., Wang, Z., Xiao, Y., Horstemeyer, M. F., Cui, X., and Chen, L., 2019, "Insight into the mechanisms of columnar to equiaxed grain transition during metallic additive manufacturing," Additive Manufacturing, 26, pp. 22-29.

[74] Zhang, Z., Yao, X., and Ge, P., 2020, "Phase-field-model-based analysis of the effects of powder particle on porosities and densities in selective laser sintering additive manufacturing," International Journal of Mechanical Sciences, 166, p. 105230.

[75] Liu, P., Wang, Z., Xiao, Y., Lebensohn, R., Liu, Y., Horstemeyer, M., Cui, X., and Chen, L., 2020, "Integration of phase-field model and crystal plasticity for the prediction of process-structure-property relation of additively manufactured metallic materials," International Journal of Plasticity, p. 102670.

[76] Stathatos, E., and Vosniakos, G.-C., 2019, "Real-time simulation for long paths in laser-based additive manufacturing: a machine learning approach," The International Journal of Advanced Manufacturing Technology, 104(5-8), pp. 1967-1984.

[77] Mozaffar, M., Paul, A., Al-Bahrani, R., Wolff, S., Choudhary, A., Agrawal, A., Ehmann, K., and Cao, J., 2018, "Data-driven prediction of the high-dimensional thermal history in directed energy deposition processes via recurrent neural networks," Manufacturing letters, 18, pp. 35-39.

[78] Roy, M., and Wodo, O., 2020, "Data-driven modeling of thermal history in additive manufacturing," Additive Manufacturing, 32, p. 101017.

[79] Chatterjee, A., 2000, "An introduction to the proper orthogonal decomposition," Current Science, pp. 808-817.

[80] Helton, J. C., and Davis, F. J., 2003, "Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems," Reliability Engineering & System Safety, 81(1), pp. 23-69.

[81] Hu, Z., Ao, D., and Mahadevan, S., 2017, "Calibration experimental design considering field response and model uncertainty," Computer Methods in Applied Mechanics and Engineering, 318, pp. 92-119.

[82] Wang, Z., Liu, P., Xiao, Y., Cui, X., Hu, Z., and Chen, L., 2019, "A data-driven approach for process optimization of metallic additive manufacturing under uncertainty," Journal of Manufacturing Science and Engineering, 141(8).

[83] Wang, Z., Liu, P., Hu, Z., and Chen, L., 2019, "Simulation-Based Process Optimization of Metallic Additive Manufacturing Under Uncertainty," International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, American Society of Mechanical Engineers, p. V02AT03A030.

[84] Bontha, S., Klingbeil, N. W., Kobryn, P. A., and Fraser, H. L., 2006, "Thermal process maps for predicting solidification microstructure in laser fabrication of thin-wall structures," Journal of Materials Processing Technology, 178(1-3), pp. 135-142.

[85] Kundin, J., Mushongera, L., and Emmerich, H., 2015, "Phase-field modeling of microstructure formation during rapid solidification in Inconel 718 superalloy," Acta Materialia, 95, pp. 343-356.

[86] Liu, J., and To, A. C., 2017, "Quantitative texture prediction of epitaxial columnar grains in additive manufacturing using selective laser melting," Additive Manufacturing, 16, pp. 58-64.

[87] Huang, X., Xie, T., Wang, Z., Chen, L., Zhou, Q., and Hu, Z., 2021, "A Transfer Learning-Based Multi-Fidelity Point-Cloud Neural Network Approach for Melt Pool Modeling in Additive Manufacturing," ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part B: Mechanical Engineering, 8(1), p. 011104.

[88] Al-Bermani, S. S., Blackmore, M. L., Zhang, W., and Todd, I., 2010, "The Origin of Microstructural Diversity, Texture, and Mechanical Properties in Electron Beam Melted Ti-6Al-4V," Metallurgical and Materials Transactions A, 41(13), pp. 3422-3434.

[89] Dürr, H., Pilz, R., and Eleser, N. S., 1999, "Rapid tooling of EDM electrodes by means of selective laser sintering," Computers in Industry, 39(1), pp. 35-45.

[90] Tapia, G., Elwany, A., and Sang, H., 2016, "Prediction of porosity in metal-based additive manufacturing using spatial Gaussian process models," Additive Manufacturing, 12, pp. 282-290.

[91] Rankouhi, B., Jahani, S., Pfefferkorn, F. E., and Thoma, D. J., 2021, "Compositional grading of a 316L-Cu multi-material part using machine learning for the determination of selective laser melting process parameters," Additive Manufacturing, 38, p. 101836.

[92] Lee, C. H., Kühn, U., Lee, S. C., Park, S. J., Schwab, H., Scudino, S., and Kosiba, K., 2020, "Optimizing laser powder bed fusion of Ti-5Al-5V-5Mo-3Cr by artificial intelligence," Journal of Alloys and Compounds, p. 158018.

[93] Liu, Q., Wu, H., Paul, M. J., He, P., Peng, Z., Gludovatz, B., Kruzic, J. J., Wang, C. H., and Li, X., 2020, "Machine-learning assisted laser powder bed fusion process optimization for AlSi10Mg: New microstructure description indices and fracture mechanisms," Acta Materialia, 201, pp. 316-328.

[94] Imani, F., Chen, R., Diewald, E., Reutzel, E., and Yang, H., 2019, "Deep learning of variant geometry in layerwise imaging profiles for additive manufacturing quality control," Journal of Manufacturing Science and Engineering, 141(11).

[95] Wang, Z., Wang, X., Yang, W., Xiao, Y., Liu, Y., and Chen, L., 2020, "yNet: a multi-input convolutional network for ultra-fast simulation of field evolvement," arXiv preprint arXiv:2012.10575.

[96] Ronneberger, O., Fischer, P., and Brox, T., 2015, "U-net: Convolutional networks for biomedical image segmentation," International Conference on Medical image computing and computer-assisted intervention, Springer, pp. 234-241.

[97] Long, J., Shelhamer, E., and Darrell, T., 2015, "Fully convolutional networks for semantic segmentation," Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3431-3440.

[98] Ioffe, S., and Szegedy, C., 2015, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," arXiv preprint arXiv:1502.03167.

[99] Agarap, A. F., 2018, "Deep learning using rectified linear units (relu)," arXiv preprint arXiv:1803.08375.

[100] Yuan, Z., Jiang, Y., Li, J., and Huang, H., 2020, "Hybrid-DNNs: Hybrid Deep Neural Networks for Mixed Inputs," arXiv preprint arXiv:2005.08419.

[101] Law, S., Paige, B., and Russell, C., 2019, "Take a look around: using street view and satellite images to estimate house prices," ACM Transactions on Intelligent Systems and Technology (TIST), 10(5), pp. 1-19.

[102] Allaire, F., Mallet, V., and Filippi, J.-B., 2021, "Emulation of wildland fire spread simulation using deep learning," Neural Networks, 141, pp. 184-198.

[103] Noh, H., Hong, S., and Han, B., 2015, "Learning deconvolution network for semantic segmentation," Proceedings of the IEEE international conference on computer vision, pp. 1520-1528.

[104] Badrinarayanan, V., Kendall, A., and Cipolla, R., 2017, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," IEEE transactions on pattern analysis and machine intelligence, 39(12), pp. 2481-2495.

[105] Hu, J., Shen, L., and Sun, G., 2018, "Squeeze-and-excitation networks," Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7132-7141.

[106] Abdeljawad, F., Bolintineanu, D. S., Cook, A., Brown-Shaklee, H., DiAntonio, C., Kammler, D., and Roach, A., 2019, "Sintering processes in direct ink write additive manufacturing: A mesoscopic modeling approach," Acta Materialia, 169, pp. 60-75.

[107] Rosenthal, W. S., Grogan, F. C., Li, Y., Barker, E. I., Christ, J. F., Pope, T. R., Battu, A. K., Varga, T., Barrett, C. A., and Warner, M. G., "" Sintering" Models and In-Situ Experiments: Data Assimilation for Microstructure Prediction in SLS Additive Manufacturing of Nylon Components," MRS Advances, pp. 1-9.

[108] Satpathy, B. B., Nandy, J., and Sahoo, S., 2018, "Investigation of consolidation kinetics and microstructure evolution of Al alloys in direct metal laser sintering using phase field simulation," IOP ConfSer Mater ScEng, 338, pp. 1-6.

[109] Bierwisch, C., Mohseni-Mofidi, S., Dietemann, B., Kraft, T., Rudloff, J., and Lang, M., 2020, "Particle-based simulation, dimensional analysis and experimental validation of laser absorption and thermo-viscous flow during sintering of polymers," Procedia Cirp, 94, pp. 74-79.

[110] Furumoto, T., Koizumi, A., Alkahari, M. R., Anayama, R., Hosokawa, A., Tanaka, R., and Ueda, T., 2015, "Permeability and strength of a porous metal structure fabricated by additive manufacturing," Journal of Materials Processing Technology, 219, pp. 10-16.

[111] Basalah, A., Shanjani, Y., Esmaeili, S., and Toyserkani, E., 2012, "Characterizations of additive manufactured porous titanium implants," Journal of Biomedical Materials Research Part B: Applied Biomaterials, 100(7), pp. 1970-1979.

[112] Kantzos, C., Lao, J., and Rollett, A., 2019, "Design of an interpretable Convolutional Neural Network for stress concentration prediction in rough surfaces," Materials Characterization, 158, p. 109961.

[113] Koeppe, A., Padilla, C. A. H., Voshage, M., Schleifenbaum, J. H., and Markert, B., 2018, "Efficient numerical modeling of 3D-printed lattice-cell structures using neural networks," Manufacturing letters, 15, pp. 147-150.

[114] Mangal, A., and Holm, E. A., 2018, "Applied machine learning to predict stress hotspots I: Face centered cubic materials," International Journal of Plasticity, 111, pp. 122-134.

[115] Mangal, A., and Holm, E. A., 2019, "Applied machine learning to predict stress hotspots II: Hexagonal close packed materials," International Journal of Plasticity, 114, pp. 1-14.

[116] Bayraktar, Ö., Uzun, G., Çakiroğlu, R., and Guldas, A., 2017, "Experimental study on the 3D ‐ printed plastic parts and predicting the mechanical properties using artificial neural networks," Polymers for Advanced Technologies, 28(8), pp. 1044-1051.

[117] Garg, A., Tai, K., Lee, C., and Savalani, M., 2014, "A hybrid $$\text {M} 5^\prime $$-genetic programming approach for ensuring greater trustworthiness of prediction ability in modelling of FDM process," Journal of Intelligent Manufacturing, 25(6), pp. 1349-1365.

[118] Collins, P., Haden, C., Ghamarian, I., Hayes, B., Ales, T., Penso, G., Dixit, V., and Harlow, G., 2014, "Progress toward an integration of process–structure–property–performance models for “three-dimensional (3-D) printing” of titanium alloys," JOM Journal of the Minerals Metals and Materials Society, 66(7), pp. 1299-1309.

[119] Zhang, J., Wang, P., and Gao, R. X., 2019, "Deep learning-based tensile strength prediction in fused deposition modeling," Computers in Industry, 107, pp. 11-21.

[120] Wang, Z., Liu, P., Ji, Y., Mahadevan, S., Horstemeyer, M. F., Hu, Z., Chen, L., and Chen, L.-Q., 2019, "Uncertainty Quantification in Metallic Additive Manufacturing Through Physics-Informed Data-Driven Modeling," JOM Journal of the Minerals Metals and Materials Society, 71(8), pp. 2625-2634.

[121] Wang, Z.-L., and Adachi, Y., 2019, "Property prediction and properties-to-microstructure inverse analysis of steels by a machine-learning approach," Materials Science and Engineering: A, 744, pp. 661-670.

[122] Liu, J., Chang, H., Hsu, T., and Ruan, X., 2000, "Prediction of the flow stress of high-speed steel during hot deformation using a BP artificial neural network," Journal of Materials Processing Technology, 103(2), pp. 200-205.

[123] Toros, S., and Ozturk, F., 2011, "Flow curve prediction of Al–Mg alloys under warm forming conditions at various strain rates by ANN," Applied Soft Computing, 11(2), pp. 1891-1898.

[124] Yang, C., Kim, Y., Ryu, S., and Gu, G. X., 2020, "Prediction of composite microstructure stress-strain curves using convolutional neural networks," Materials & Design, 189, p. 108509.

[125] Tanti, M., Gatt, A., and Camilleri, K. P., 2017, "Where to put the image in an image caption generator," arXiv preprint arXiv:1703.09137.

[126] Ketkar, N., 2017, "Introduction to keras," Deep learning with Python, Springer, pp. 97-111.

[127] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C., 2018, "Mobilenetv2: Inverted residuals and linear bottlenecks," Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4510-4520.

[128] Simonyan, K., and Zisserman, A., 2014, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556.

[129] Multiphysics, C., 1998, "Introduction to comsol multiphysics®," COMSOL Multiphysics, Burlington, MA, accessed Feb, 9, p. 2018.

[130] Tapia, G., King, W., Johnson, L., Arroyave, R., Karaman, I., and Elwany, A., 2018, "Uncertainty Propagation Analysis of Computational Models in Laser Powder Bed Fusion Additive Manufacturing using Polynomial Chaos Expansions," Journal of Manufacturing Science and Engineering, 140.

[131] Lu, Z., Li, D., Lu, B., Zhang, A., Zhu, G., and Pi, G., 2010, "The prediction of the building precision in the Laser Engineered Net Shaping process using advanced networks," Optics and Lasers in Engineering, 48(5), pp. 519-525.

[132] Kok, Y., Tan, X. P., Wang, P., Nai, M. L. S., Loh, N. H., Liu, E., and Tor, S. B., 2018, "Anisotropy and heterogeneity of microstructure and mechanical properties in metal additive manufacturing: A critical review," Materials & Design, 139, pp. 565-586.

[133] Wang, Z., Liu, P., Xiao, Y., Cui, X., Hu, Z., and Chen, L., 2019, "A data-driven approach for process optimization of metallic additive manufacturing under uncertainty," Journal of Manufacturing Science and Engineering, accepted.

[134] Hong, R., Zhang, L., Lifton, J., Daynes, S., Wei, J., Feih, S., and Lu, W. F., 2021, "Artificial neural network-based geometry compensation to improve the printing accuracy of selective laser melting fabricated sub-millimetre overhang trusses," Additive Manufacturing, 37, p. 101594.

[135] Zhu, Z., Anwer, N., Huang, Q., and Mathieu, L., 2018, "Machine learning in tolerancing for additive manufacturing," CIRP annals, 67(1), pp. 157-160.

[136] Paul, R., Anand, S., and Gerner, F., 2014, "Effect of thermal deformation on part errors in metal powder based additive manufacturing processes," Journal of Manufacturing Science and Engineering, 136(3).

[137] He, Y., Fei, F., Wang, W., Song, X., Sun, Z., and Baek, S., 2018, "Predicting manufactured shapes of a projection micro-stereolithography process via convolutional encoder-decoder networks," ASME 2018 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, American Society of Mechanical Engineers, pp. V01BT02A033-V001BT002A033.

[138] Chowdhury, S., and Anand, S., 2016, "Artificial neural network based geometric compensation for thermal deformation in additive manufacturing processes," ASME 2016 11th International Manufacturing Science and Engineering Conference, American Society of Mechanical Engineers Digital Collection.

[139] Cheng, L., Wang, A., and Tsung, F., 2018, "A prediction and compensation scheme for in-plane shape deviation of additive manufacturing with information on process parameters," IISE Transactions, 50(5), pp. 394-406.

[140] Shen, Z., Shang, X., Zhao, M., Dong, X., Xiong, G., and Wang, F.-Y., 2019, "A learning-based framework for error compensation in 3D printing," IEEE transactions on cybernetics, 49(11), pp. 4042-4050.

[141] Hasegawa, K., Fukami, K., Murata, T., and Fukagata, K., 2020, "Machine-learning-based reduced-order modeling for unsteady flows around bluff bodies of various shapes," Theoretical and Computational Fluid Dynamics, 34(4), pp. 367-383.

[142] Hasegawa, K., Fukami, K., Murata, T., and Fukagata, K., 2020, "CNN-LSTM based reduced order modeling of two-dimensional unsteady flows around a circular cylinder at different Reynolds numbers," Fluid Dynamics Research, 52(6), p. 065501.

[143] Lee, S., and You, D., 2017, "Prediction of laminar vortex shedding over a cylinder using deep learning," arXiv preprint arXiv:1712.07854.

[144] Guo, X., Li, W., and Iorio, F., 2016, "Convolutional neural networks for steady flow approximation," Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, pp. 481-490.

[145] Bhatnagar, S., Afshar, Y., Pan, S., Duraisamy, K., and Kaushik, S., 2019, "Prediction of aerodynamic flow fields using convolutional neural networks," Computational Mechanics, 64(2), pp. 525-545.

[146] Baldi, P., and Hornik, K., 1989, "Neural networks and principal component analysis: Learning from examples without local minima," Neural Networks, 2(1), pp. 53-58.

[147] Zeiler, M. D., and Fergus, R., 2014, "Visualizing and understanding convolutional networks," European conference on computer vision, Springer, pp. 818-833.

[148] Murata, T., Fukami, K., and Fukagata, K., 2020, "Nonlinear mode decomposition with convolutional neural networks for fluid dynamics," Journal of Fluid Mechanics, 882.

[149] Fukami, K., Nakamura, T., and Fukagata, K., 2020, "Convolutional neural network based hierarchical autoencoder for nonlinear mode decomposition of fluid field data," Physics of Fluids, 32(9), p. 095110.

[150] Slotwinski, J. A., Luecke, W. E., Lass, E., and Possolo, A. M., 2018, "Interlaboratory mechanical property study for Cobalt-Chromium alloy Made by Laser Powder-Bed-Fusion Additive Manufacturing."

[151] Brown, C. U., Jacob, G., Stoudt, M., Moylan, S., Slotwinski, J., and Donmez, A., 2016, "Interlaboratory study for nickel alloy 625 made by laser powder bed fusion to quantify mechanical property variability," Journal of Materials Engineering and Performance, 25(8), pp. 3390-3397.

[152] Hu, Z., and Mahadevan, S., 2017, "Uncertainty quantification in prediction of material properties during additive manufacturing," Scripta Materialia, 135, pp. 135-140.

[153] Moges, T., Ameta, G., and Witherell, P., 2019, "A Review of Model Inaccuracy and Parameter Uncertainty in Laser Powder Bed Fusion Models and Simulations," Journal of Manufacturing Science and Engineering, 141(4), p. 040801.

[154] Hu, Z., and Mahadevan, S., 2017, "Uncertainty quantification and management in additive manufacturing: current status, needs, and opportunities," The International Journal of Advanced Manufacturing Technology, 93(5-8), pp. 2855-2874.

[155] Ma, L., Fong, J., Lane, B., Moylan, S., Filliben, J., Heckert, A., and Levine, L., 2015, "Using design of experiments in finite element modeling to identify critical variables for laser powder bed fusion," International Solid Freeform Fabrication Symposium, Laboratory for Freeform Fabrication and the University of Texas Austin, TX, USA, pp. 219-228.

[156] Ghosh, S., Ma, L., Levine, L. E., Ricker, R. E., Stoudt, M. R., Heigel, J. C., and Guyer, J. E., 2018, "Single-Track Melt-Pool Measurements and Microstructures in Inconel 625," JOM Journal of the Minerals Metals and Materials Society, 70, pp. 1-6.

[157] Swain, M. C., and Cole, J. M., 2016, "ChemDataExtractor: a toolkit for automated extraction of chemical information from the scientific literature," Journal of chemical information and modeling, 56(10), pp. 1894-1904.

[158] Olivetti, E. A., Cole, J. M., Kim, E., Kononova, O., Ceder, G., Han, T. Y.-J., and Hiszpanski, A. M., 2020, "Data-driven materials research enabled by natural language processing and information extraction," Applied Physics Reviews, 7(4), p. 041317.

[159] Sun, W., Cai, Z., Li, Y., Liu, F., Fang, S., and Wang, G., 2018, "Data processing and text mining technologies on electronic medical records: a review," Journal of healthcare engineering, 2018.

[160] Ma, B., Wei, X., Liu, C., Ban, X., Huang, H., Wang, H., Xue, W., Wu, S., Gao, M., Shen, Q., Mukeshimana, M., Abuassba, A. O., Shen, H., and Su, Y., 2020, "Data augmentation in microscopic images for material data mining," npj Computational Materials, 6(1).

[161] Hornik, K., Stinchcombe, M., and White, H., 1989, "Multilayer feedforward networks are universal approximators," Neural Networks, 2(5), pp. 359-366.

[162] Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P., 1989, "Design and Analysis of Computer Experiments," Statistical Science, 4(4), pp. 409-423.

[163] Jones, D. R., 2001, "A Taxonomy of Global Optimization Methods Based on Response Surfaces," Journal of Global Optimization, 21(4), pp. 345-383.

[164] Hochreiter, S., and Schmidhuber, J., 1997, "Long short-term memory," Neural Computation, 9(8), pp. 1735-1780.

[165] Reddy, D. S., and Prasad, P. R. C., 2018, "Prediction of vegetation dynamics using NDVI time series data and LSTM," Modeling Earth Systems and Environment, 4(1), pp. 409-419.

[166] Abaqus, V., 2010, "6.10," User Subroutines Reference Manual.

[167] Wang, Y. U., 2006, "Computer modeling and simulation of solid-state sintering: A phase field approach," Acta Materialia, 54(4), pp. 953-961.

[168] Yang, Y., Kühn, P., Yi, M., Egger, H., and Xu, B.-X., 2020, "Non-isothermal Phase-Field Modeling of Heat–Melt–Microstructure-Coupled Processes During Powder Bed Fusion," JOM Journal of the Minerals Metals and Materials Society.

[169] Wang, X., Liu, Y., Li, L., Yenusah, C. O., Xiao, Y., and Chen, L., 2021, "Multi-scale phase-field modeling of layer-by-layer powder compact densification during solid-state direct metal laser sintering," Materials & Design, 203, p. 109615.

[170] Wang, Z., Jiang, C., Liu, P., Yang, W., Zhao, Y., Horstemeyer, M. F., Chen, L.-Q., Hu, Z., and Chen, L., 2020, "Uncertainty quantification and reduction in metal additive manufacturing," npj Computational Materials, 6(1), p. 175.

[171] Wang, Z., Jiang, C., Horstemeyer, M. F., Hu, Z., and Chen, L., 2020, "Uncertainty Quantification in Metallic Additive Manufacturing Through Data-Driven Modelling Based on Multi-Scale Multi-Physics Models and Limited Experiment Data," International Manufacturing Science and Engineering Conference, American Society of Mechanical Engineers, p. V001T001A032.

[172] Hu, Z., Mahadevan, S., and Ao, D., 2018, "Uncertainty aggregation and reduction in structure–material performance prediction," Computational Mechanics, 61(1-2), pp. 237-257.

[173] Arulampalam, M. S., Maskell, S., Gordon, N., and Clapp, T., 2002, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," IEEE Transactions on signal processing, 50(2), pp. 174-188.