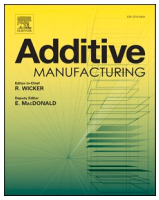
Additive Manufacturing 61 (2023) 103357

Contents lists available at ScienceDirect

Additive Manufacturing

journal homepage: www.elsevier.com/locate/addma



Review

Review of transfer learning in modeling additive manufacturing processes

Yifan Tang, M. Rahmani Dehaghani, G. Gary Wang *

Product Design and Optimization Laboratory, Simon Fraser University, Surrey, BC, Canada

ARTICLE INFO

Keywords:
AM modeling
Additive manufacturing
Transfer learning

ABSTRACT

Modeling plays an important role in the additive manufacturing (AM) process and quality control. In practice, however, only limited data are available for each product due to the relatively high AM cost, which brings challenges in building either a high-quality physics-based or data-based model. Transfer learning (TL) is a new and promising group of approaches where the model of one product (source) may be reused for another product (target) with limited new target data. This paper focuses on reviewing applications of TL in AM modeling to help advance research in this area. First, notations, definitions, and categories of TL methods are introduced along with their application scenarios. Then current applications of TL in AM modeling are summarized along with their limitations. Based on reviewed applications, recommendations are given on how to apply TL for a certain AM problem, from the perspectives of source domain determination, TL method selection, target data generation, and data preprocessing. Finally, future research directions about TL in AM modeling are discussed in the hope to explore more potential of TL in improving the AM model quality with limited data.

1. Introduction

Additive manufacturing (AM), or three-dimensional (3D) printing, refers to a product construction process by adding successive layers of materials according to computer-aided design (CAD) models [1]. AM has been applied widely in various industrial fields, such as electronics (i.e., batteries [2]), medical (i.e., hearing aids and biomanufacturing) [3], automotive (i.e., car frame, body, and door) [4], aerospace (i.e., Airbus bracket, NASA turbopump stator, and thrust chamber) [5], military [6], architecture (i.e., concrete beam, bench, and house) [7], and others. However, the unstable AM processes have brought about disparate mechanical properties, resulting in low productivity and poor quality. AM modeling is promising to alleviate these problems, considering its potential to reveal correlations among processes, structures, and properties. These models are then used for AM process optimization and control to achieve better and consistent product qualities. According to different sources for modeling, AM models could be classified into physics-based models and data-based models, whose comparisons are shown in Fig. 1.

1.1. Physics-based models

Physics-based models are those describing the underlying physical relationship between conditions and responses in AM process by mathematical/differential equations, which could be solved by numerical methods or simulation such as finite element analysis (FEA). For example, Flach and Chartoff [8] constructed a polymer shrinkage model for the stereolithography (SLA) process, by analyzing the observed shrinkage and the conversion process from monomer to polymer. Based on the prior knowledge of physical phenomena in selective laser sintering (SLS) and assumptions about powder geometry, laser, absorption coefficient, and powder bed, Wang and Kruth [9] proposed a new analytical ray-tracing model to simulate the energy absorption and penetration when metal powder mixtures are adopted. Similarly, to control the material composition of multi-material parts, a simulator of the direct laser deposition (DED) process was constructed by modeling all manufacturing steps mathematically, including part/path description, numerical control, process operation, bead deposition, and manufactured part [10]. To investigate dominant SLS process parameters affecting the sintering depth and the liquid pool geometry, Chen and Zhang [11] developed a two-dimensional temperature-transforming model, by simplifying the enthalpy-based energy equation as a nonlinear equation dependent on temperature only. Panagiotis [12] proposed a hybrid inactive/quiet element method with a transient conductive heat transfer function to efficiently simulate the thermal history of the selective laser melting (SLM) process.

The advantages of these models are that they can be transferred to various relevant processes with only a limited number of expensive

* Corresponding author.
E-mail addresses: yta88@sfu.ca (Y. Tang), mra91@sfu.ca (M. Rahmani Dehaghani), gary_wang@sfu.ca (G.G. Wang).

https://doi.org/10.1016/j.addma.2022.103357

Available online 12 December 2022

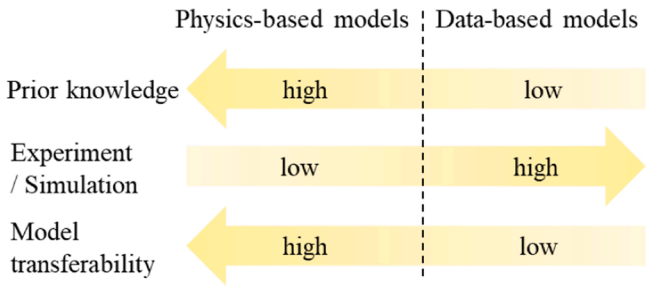


Fig. 1. Comparison of physics-based models and data-based models.

experiments for calibration and validation. However, their accuracies are restricted by prior knowledge and underlying assumptions (i.e., simplification of the process) [13]. Therefore, as shown in Fig. 1, although the transferability is high for physics-based models, the requirement for prior knowledge is also high whilst the demand for experiments/simulation is relatively low. As one can see in the figure, data-based models demonstrate opposite characteristics to physics-based models.

1.2. Data-based models

Data-based models are those constructed based on data collected from different sources, which need less prior knowledge about AM processes. The purpose of these models is to predict observations (i.e., process and product quality) from interested variables (i.e., process parameters or signatures) directly. Based on the selection of model formula, this category consists of experiment-based empirical/regression models and machine learning (ML) models learned from simulations and/or experiments.

The experiment-based empirical/regression model has a manually defined mathematical formula, whose parameters are approximated by data from existing experiments or newly designed experiments. For instance, Raghunath and Pandey [14] adopted the Taguchi method to design 16 SLS experiments considering five process parameters, including laser powder, scan speed, hatch spacing, part bed temperature, and scan length. Then three simplified linear regression model formulas were defined based on ANOVA analysis and trained with collected experiment data to predict the shrinkage at X, Y, and Z directions respectively. Similarly, the central composite design method was used to design 32 experiments for fused deposition modelling (FDM) process concerning five important parameters [15]. Then, a quadratic response surface model (QRSM) was constructed to predict the compressive strength from process parameters (i.e., orientation, layer thickness, raster angle, raster width, and air gap) and the significance of each parameter was analyzed with ANOVA. Also, the relationship between the melt pool depth and parameters (laser powder and scan speed) in laser foil printing (LFP) was extracted by QRSM [16], and the dependency between the melt pool depth and parameters was also quantified by ANOVA. Another similar work was Chikkanna *et al.* [17], where two QRSM models were constructed based on 18 experiments to predict FDM static and dynamic flexural properties according to the temperature and annealing time, respectively. Although the above empirical/regression methods complete both model construction and sensitivity analysis, these models constructed with certain formulas only perform well in specific processes with certain conditions [13]. In other words, it is difficult to transfer them to different processes with acceptable model quality.

Different from pre-defined mathematical formulation in the above empirical/regression models, ML models estimate AM process behavior by various ML techniques from experiments, simulations, or a combination of both [18]. For instance, the random forest method was used to analyze the energy performance in FDM, where the energy consumption was predicted based on features extracted from G-code files [19]. This model provides a way to control the energy cost when designing new FDM products but has limited applicability in other AM processes considering only features from G-code files of the FDM process. Olleak and Xi [20] constructed a Gaussian process regression (GPR) model to predict the melt pool geometry according to SLM process parameters and adopted the *u*-Pooling metric strategy to calibrate the uncertainties in powder absorptivity and spot diameter. This framework is promising for different AM modeling problems with uncertainties. But the final model accuracy is affected greatly by the initial assumptions of interested uncertainties, as only one-step uncertainty calibration is performed in the framework. If the initial assumptions are far from the actual uncertainty parameters, the final calibrated model could be unreliable.

To improve the product geometry accuracy in wire arc additive manufacturing (WAAM), Ding *et al.* [21] adopted support vector regression (SVR) to find the correlation between welding parameters (i.e., wire feed rate, travel speed, and inter-pass temperatures) and product properties (i.e., overlapping distance and bead height). Then an SVR backward model was designed to suggest appropriate welding parameters to produce parts with high accuracy, given the desired geometry. Such a model could control the WAAM process layer-by-layer to reduce the overbuilt volume, which in turn reduces the material waste and the machining cost. However, its applicability in complex geometries and various printing paths should be further studied, as only a hollow frustum with a concentric-circle printing path is tested in their case study.

To reveal the correlation between laser scanning patterns and thermal history distribution in laser-aided additive manufacturing (LAAM), Ren *et al.* [22] developed a recurrent neural network and deep neural network model to predict the thermal history of a single layer with arbitrary geometries. Then, Zhou *et al.* [23] extended their work to simulate the thermal history in the multi-layers gas metal arc welding process, by considering the thermal diffusion among adjacent finite elements. Both models are promising in solving path planning problems in LAAM considering different geometries, but their reliability is limited by the fixed matrix size to represent AM parts. If the new part is much larger than those generating the training data, the data matrix would be too sparse to accurately represent the new part for training and prediction.

For porosity problems in various SLM products constructed by different machines, a set of physics-informed models integrated with linear regression, GPR and SVR, were developed to correlate porosity levels with physical effects defined by machine settings (i.e., laser energy density and laser radiation pressure) [24]. Also, the self-organizing map method was adopted to project the high-dimensional spatial distortion distribution (i.e., 3-dimensional cloud data) to a low-dimensional feature, whose relationship with process parameters was constructed by a hybrid group method of data handling and genetic programming [25]. With accessible molten pool images and temperature data in wire-feed LAAM, Jamnikar *et al.* [26] developed a convolutional neural network (CNN) model to predict geometries of the bead and fusion zone simultaneously. These models could be used for *in-situ* process monitoring to detect geometry properties, but their online prediction accuracy is difficult to maintain as only limited offline data with certain AM process scenarios are used for training.

Apart from several works mentioned above about quality and process control, more details about current ML applications in AM design, process/performance optimization, and *in-situ* process monitoring/control could be found in several review papers, where the potential of ML in solving AM modeling problems has been sufficiently demonstrated [27–29]. More importantly, references [27,28] pointed out that exploring the design space for better properties of new products would benefit from data sharing among different infrastructures or *in/ex-situ* processes.

The idea of data sharing is similar to transfer learning (TL), which improves the learning performance of the target model by reusing the source knowledge in the target domain [30]. In recent decades,

successes of TL applications have been observed in numerous research areas, such as medical tests [31], bio-information analysis [32], transportation [33], recommendation systems [34], and defect detection [35]. Compared with conventional ML methods, TL methods are more promising in solving AM modeling problems considering the following two factors:

- *Requires much less training data and thus reduces time and cost.* Currently, the modeling of different AM products is isolated from each other. More specifically, given a new AM product, one needs to collect data from experiments or simulations, and then perform ML-based modeling. This procedure is time-consuming as AM experiments or simulations would take hours or days. Besides, to construct an acceptable model, a reasonable number of experiments or simulations are required, which means massive time expenses are used for different products. For this problem, TL could break the isolation among different-but-similar products by transferring knowledge from historical products to new products. Therefore, the modeling procedures would be simplified by reducing the required number of experiments or simulations whose compatibility could be verified by the similarity metrics in TL, which reduces the requirement of expert verification and improves the process efficiency. More details about applications of TL in AM and criteria for applying TL in AM will be discussed in the next sections.
- *Bridge among various data sources.* Although various modeling methods have been proposed, it is still difficult to guarantee their performance in all AM problems involving different machines, materials, products, or processes, due to expensive data collection and complicated physical relationships. In current applications, various data sources have been collected with different fidelities. For example, experimental results are usually limited and expensive but have the highest fidelity. Computer simulations, such as finite element analysis and computational fluid dynamics, are relatively inexpensive but their predictions could deviate significantly from real-world experiments. These data sources can reveal some information about parameters of interest but cannot be applied directly to new products when a different machine, material, and AM process is used. TL provides a possibility to connect various data sources with limited new data and construct an informed model with high transferability.

Based on these considerations, this paper aims to reveal how to construct acceptable AM models when only limited data on new products are available but sufficient data on historical products are accessible. For clarification, this paper wants to explore directions about applying TL in AM modeling via analyzing existing relevant works, where various products (i.e., different geometries) are fabricated with different printers (i.e., EOS M290 and MSU Renishaw AM 400) [36], materials (i.e., Ti-6Al-4 V and 316 L stainless steel) [37], and AM processes (i.e., bead-on-plate process and bead-on-powder process) [38].

The general connection between TL techniques and AM application scenarios are shown in Fig. 2. Generally, the instance-based and feature-based TL techniques are only applied in quality prediction and process optimization, while the model-based TL and multi-task learning are applied widely in quality prediction and inspection, defect detect, and process monitor. More details are discussed in the following sections. The background of TL is briefly reviewed in Section 2 with its definition, categories, and theoretical comparisons. Section 3 discusses their connections in detail with a review of applications of TL techniques in AM modeling. In Section 4, a discussion on how to apply TL to a certain AM problem is presented in detail from four different aspects. Based on the above analysis, several potential future research directions are proposed

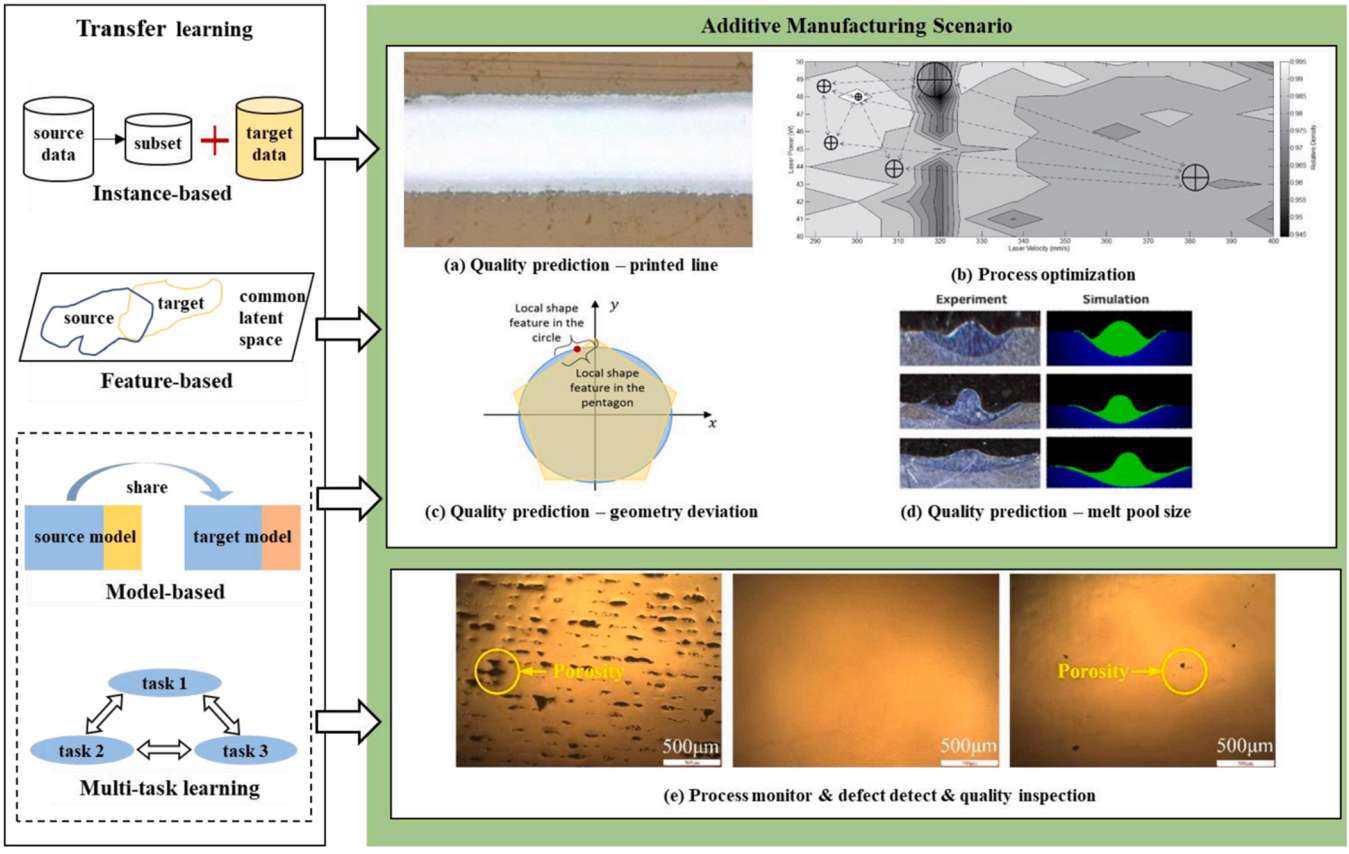


Fig. 2. Connections between transfer learning and additive manufacturing scenarios: (a) quality prediction of printed line [39], (b) process optimization [40], (c) quality prediction of geometry deviation [41], (d) quality prediction of melt pool size [38], and (e) process monitor & defect detect & quality inspection [42].

in Section 5, as well as suggestions on how to select TL frameworks for different AM modeling problems. Section 6 provides a summary of this paper.

2. Introduction of transfer learning

In this section, the background of TL is briefly reviewed about its notations, definitions, and categories.

2.1. Notations and definitions

Some notations and definitions about TL are listed below.

- *Domain* $\mathscr{D}$: A domain consists of a feature space $\mathscr{X}$ and a marginal distribution $P(\boldsymbol{X})$, represented as $\mathscr{D} = \{\mathscr{X}, P(\boldsymbol{X})\}$. The symbol $\boldsymbol{X} = \{\boldsymbol{x}_i \in \mathscr{X} | i \in [1, n]\}$ denotes the tuple of input variable $\boldsymbol{x}$, and n is the size of input variables.
- *Task* $\mathscr{T}$: A task $\mathscr{T}$ is composed of a label space $\mathscr{Y}$ and a decision function f, denoted as $\mathscr{T} = \{\mathscr{Y}, f\}$. The function f can be learned from the training data $(\boldsymbol{X}, \boldsymbol{Y})$ where $\boldsymbol{Y} = \{y_i \in \mathscr{Y} | i \in [1, n]\}$ is the tuple of output variables. For regression problems, $y_i = f(\boldsymbol{x}_i|\boldsymbol{\theta})$ where $\boldsymbol{\theta}$ includes all model parameters to be learned. In classification, the outputs are conditional distributions, and the model can be formulated as $f(\boldsymbol{x}_i) = P(y_i|\boldsymbol{x}_i; \boldsymbol{\theta})$. For example, in a binary classification task, the output y_i is "True" or "False".
- *Source/Target*: In reality, different domains and tasks are observed with data of different sizes and output conditions. Generally, the source domain $\mathscr{D}^s$ and task $\mathscr{T}^s$ are defined as the one with sufficient data $\boldsymbol{D}^s = \{(\boldsymbol{x}_i^s, y_i^s) | i \in [1, n^s]\}$. By contrast, the target domain $\mathscr{D}^t$ and task $\mathscr{T}^t$ usually contain limited data $\boldsymbol{D}^t = \{(\boldsymbol{x}_i^t, y_i^t) | i \in [1, n^t], n^t \ll n^s\}$ or some unstructured data without labels.
- *Transfer learning (TL)*: Given source and target domains, TL aims to improve the learning performance of the target task $\mathscr{T}^t$ by extracting knowledge from $\mathscr{D}^s$ and $\mathscr{T}^s$.

2.2. Categories of TL

This section aims to provide a brief high-level classification of various TL frameworks, including instance-based, feature-based, model-based TL, and multi-task learning (MTL). More details about their mathematical backgrounds and various application scenarios could be found in review papers [30,43].

2.2.1. Instance-based TL

The instance-based TL assumes that source and target domains have different marginal distributions $P^s(\boldsymbol{X}) \neq P^t(\boldsymbol{X})$ but with the same conditional distribution $P^s(Y|\boldsymbol{X}) = P^t(Y|\boldsymbol{X})$. Its fundamental idea is the instance weighting strategy, where different weights $\boldsymbol{w}$ are assigned to each source and target data directly [44] or iteratively [45], so that the target model is constructed with the combination of target data and weighted source data. For instance, a kernel mean matching is proposed to estimate the weights by minimizing the difference between means of source and target domain data in a reproduced kernel Hilbert space [44]; a Kullback-Leibler importance estimation procedure is designed to minimize the Kullback-Leibler divergence of source and target domains [46]. Both direct estimation methods could be incorporated into many other methods, but the estimation accuracy of weights depends on the designed distance matrix.

Different from direct estimation methods, the iterative methods want to find a procedure to reduce the weights of source data with negative effects on the TL performance. One representative work is TrAdaBoost in classification [45], where the source dataset $\boldsymbol{D}^s$ and target dataset $\boldsymbol{D}^t$ were combined to train weak classifiers iteratively. During each iteration, the weights of source data were modified based on a designed constant value and their classification results, while the weights of target data were updated according to their classification results and error. As a result, the relative weights were increased gradually for source data with correct classification results, as well as target data with incorrect classification results. These data would have a higher possibility to be selected to construct new weak classifiers in the next iteration. Such framework has been explored and extended in many different tasks, i.e., multi-source classification [47]. To extend TrAdaBoost in regression, Pardoe and Stone [48] proposed a Two-stage TrAdaBoost.R2 (T2ABR2 for short) method, where the first stage aimed to increase only the relative weights of target data with a larger prediction error according to AdaBoost.R2 [49], and the second stage updated all weights of source and target data simultaneously to increase the relative weights of source data with more similarities with the target data.

2.2.2. Feature-based TL

The feature-based TL can tackle source and target domains with different marginal or conditional distributions ($P^s(\boldsymbol{X}) \neq P^t(\boldsymbol{X})$ or $P^s(Y|\boldsymbol{X}) \neq P^t(Y|\boldsymbol{X})$). The underlying idea is a kind of data transformation method, where the source and target features are transformed into a common feature space to minimize their difference. Generally, the feature space is a new representation of the dataset $\boldsymbol{D}$ (i.e., $\boldsymbol{F} = \phi(\boldsymbol{D})$), where $\phi(\cdot)$ is a designed representation operation, such as eigenvalue decomposition [50] and feature augmentation [51].

Different in transformation structure, the asymmetric [52] and symmetric [53] feature-based TL methods are shown in Fig. 3.

The asymmetric feature-based TL transfers the source feature by finding the optimal transformation matrix $H^{s\to t}$. One representative method is the subspace alignment (SA) method [50], whose optimization problem is to minimize the Bregman matrix divergence between the target feature and the transformed source feature:

$$\min_{H^{s\to t}} \|\boldsymbol{M}^s H^{s\to t} - \boldsymbol{M}^t\|_F^2 \tag{1}$$

where $\boldsymbol{M}^s$ and $\boldsymbol{M}^t$ are the bases (i.e., eigenvectors) of source and target subspaces respectively; $\|\cdot\|_F^2$ is the Frobenius norm. After obtaining the optimal $H^{s\to t}$, a learning machine could be trained based on the combination of the target data and the transformed source data directly. Apart from Bregman matrix divergence, various criteria have been applied widely in feature-based TL methods to measure the feature difference, including the maximum mean discrepancy [54], Hilbert-Schmidt independence criterion [55], and Wasserstein distance [56].

The symmetric feature-based TL tries to construct a new common feature space, where the source and target features have a small dif-

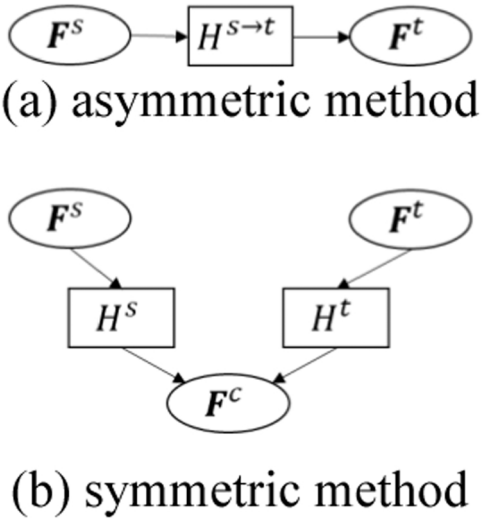


Fig. 3. Frameworks of asymmetric and symmetric feature-based methods.

ference after their respective transformation. The general optimization problem can be formulated as:

$$\min Dist(\boldsymbol{F}^s, \boldsymbol{F}^t, H^s, H^t) = \min_{H^s, H^t} Dist(\boldsymbol{D}^s, \boldsymbol{D}^t, H^s, H^t, \phi(\cdot)) \quad (2)$$

where $Dist(\cdot)$ represents the selected feature difference measurement with/without regularizers; H^s and H^t are the source and the target transformation matrices respectively. In most cases, the common feature space is defined manually according to criteria. For instance, the transfer component analysis method [57] was proposed to extract the common latent features, by minimizing the maximum mean discrepancy of source and target domains in a kernel Hilbert space. The proposed spectral domain-specific feature alignment method [58] identified domain-specific and domain-independent features from the source and target dataset directly, based on which a bipartite graph was constructed to find the co-occurrence relationship between two features. Then a spectral clustering algorithm was applied to the graph to split domain-specific and domain-independent features into clusters that represented the new common feature space. Similarly, the heterogeneous spectral mapping method [59] found the common feature space by minimizing the difference between projected domains, while maintaining the original data structure.

2.2.3. Model-based TL

Different from knowledge represented as data in the instance-based TL and feature space in the feature-based TL, the model-based TL aims to facilitate the target model construction and training process, by using source model structures or parameters.

From the perspective of model construction, the source model could be a sub-structure in the target model, i.e., $f^t = extend(f^s)$. The representative work is fine-tuning framework [60], which shares certain structures of the pre-trained source model and their corresponding parameters in the target model directly. Then new substructures are designed according to the target tasks and tuned with limited target data. Considering its easy implementation, it has been applied successfully in image classification [61], disease detection [31], communication [62], and so on. Besides, with several source domains available at the same time, the target model could be represented as an ensemble of source models, i.e., $f^t = ensemble(\{f_i^s | i = 1, \ldots, k\})$, where k is the number of source domains. Yao and Doretto [47] extended TrAdaBoost to multiple source domains, where a group of candidate weak classifiers were trained for each source domain at each iteration. And the target classifier was a weighted average of optimal source weak classifiers with the minimal classification error at each iteration. Different from the above global weighting, a locally weighted ensemble framework was designed to assign adaptive weights to each source model according to the similarity between target domain and the corresponding source domain [63].

During the training process, a new loss function was defined considering source knowledge:

$$L^t = l^t + l^{t,s} \quad (3)$$

where L^t is the updated loss function for the target model; l^t is the conventional loss function based on target model prediction and target data, such as the mean square error in regression tasks, or cross-entropy and hinge loss in classification tasks; $l^{t,s}$ is a regularizer constructed based on both domains. For instance, the domain-dependent regularizer [64] was designed to control the difference between target and source predictions on the target data, i.e., $l^{t,s} = \lambda \| f^t(\boldsymbol{X}^t) - f^s(\boldsymbol{X}^t) \|_2^2$, where λ is a regularizer hyperparameter. Compared with source model parameters, the target model parameters can be restricted directly in the regularizer, i.e., $l^{t,s} = \lambda \| \boldsymbol{\theta}^t - \beta \boldsymbol{\theta}^s \|_2^2$, where β is the weighting parameter to control the degree of transfer [65]. By assuming source and target domains are similar to some extent, these methods can restrict the target performance or parameters in the vicinity of source counterparts to improve the accuracy of the target model.

2.2.4. Multi-task learning (MTL)

As an inductive TL method [43], MTL learns from multiple tasks $\{\mathscr{T}_i | i = 1, \ldots, m\}$ simultaneously to improve the learning performance of each task by using the knowledge contained in all/some tasks [66]. Different from explicit definitions of source and target in the above three TL categories, each task in MTL can be regarded as a source or target to others. In most cases, MTL tasks can be completed by a hard or soft parameter sharing mechanism [67].

In the hard parameter sharing, different tasks share the input layer and some hidden layers to project to a common latent space, and have their task-specific layers [68], shown in Fig. 4(a). Such a mechanism could reduce the risk of overfitting for each task when a common feature presentation is captured with more tasks involved [67]. Considering the common layers applied, the hard parameter sharing could be applied only in homogeneous TL tasks where the target and source input spaces are the same. Conversely, each task has its own model in the soft parameter sharing mechanism, while the model parameters of all tasks are controlled simultaneously by regularizers or constraints. As no strict restriction is required for input spaces and model structures of various tasks, soft parameter sharing could solve both homogeneous and heterogeneous tasks where the target and source input spaces are different.

In both mechanisms, the MTL loss function is formulated as:

$$L^{MTL} = \sum_{i=1}^{m} l^i + l^r \quad (4)$$

where L^{MTL} is the MTL loss function; l^i is the conventional loss function of the task $\mathscr{T}_i$; m is the number of tasks; l^r is an optional regularizer, such as Frobenius norm [69], $\{l_p | p = 1, 2, \infty\}$ norm [70–72], capped-l_p norm [73] of model parameters, and the trace of a square matrix defined with model parameters and feature covariance in all tasks [74]. During training, information in all tasks is shared mutually to update all parameters together, which represents the knowledge transfer process. Besides, if a certain task is focused, the loss functions of other tasks serve as noise or bias, which would improve the generalization performance of a certain task [75].

2.3. Summary remarks

According to the underlying assumptions or theories, each TL category has its preferable application scenarios.

- *Instance-based TL*: Considering the sampled target and source data are applied directly, both domains are required to have the same feature space, i.e., $\mathscr{X}^s = \mathscr{X}^t$, which is also known as homogeneous TL. In other words, the source and target inputs should consider the same variables. Besides, this category performs best when the source and target domains have an identical conditional distribution [76].
- *Feature-based TL*: Without the restriction of feature space, this category could be applied in both homogeneous and heterogeneous ($\mathscr{X}^s \neq \mathscr{X}^t$) TL tasks [59]. More specifically, the feature-based TL has potential to solve tasks with any combination of $P^s(\boldsymbol{X}) \neq P^t(\boldsymbol{X})$, $P^s(\boldsymbol{Y}|\boldsymbol{X}) = P^t(\boldsymbol{Y}|\boldsymbol{X})$, $\mathscr{X}^s \neq \mathscr{X}^t$, or $\mathscr{Y}^S \neq \mathscr{Y}^t$. This merit is attributed to the transformation matrix, which could map the source and target features with different dimensions to a common feature space.
- *Model-based TL*: Similarly, the model-based TL also has wider applications in both homogeneous and heterogeneous TL tasks, as it allows construction of the target model by combining target-specific input and output layers with sub-structures in the pre-trained source model.
- *Multi-task learning*: Although MTL is applicable for both homogeneous and heterogeneous TL tasks, it would be preferred when only a few relevant historical data are accessible. Without the prerequisite of sufficient source data, MTL can combine insufficient data in

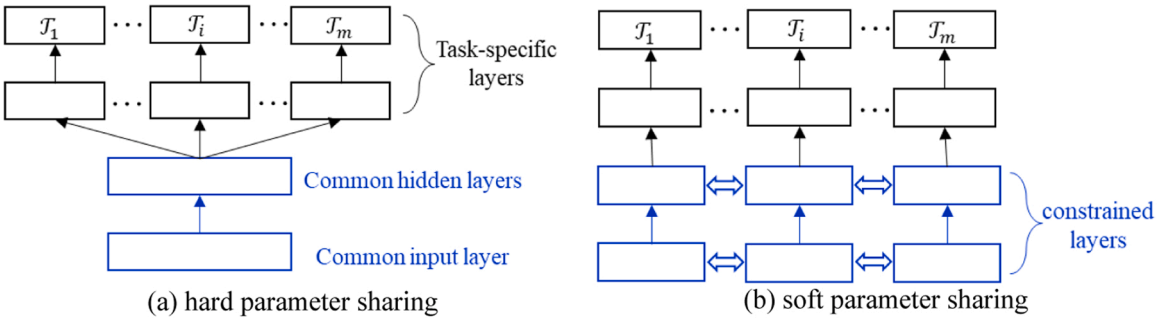


Fig. 4. Commonly applied MTL frameworks [67].

several source domains to improve the performance of the target (main) task. If aiming to obtain an acceptable performance for each task, MTL could lower the total computation cost for training all the tasks separately [75].

Although the above TL frameworks have been applied widely and successfully with various characteristics, they are still afflicted with two common challenges.

- *Similarity:* The most important factor affecting the TL performance is the similarity between the source and target domains. When the similarity is low, transferring source knowledge would hurt target learning, which is known as negative transfer [77]. To mitigate this problem, feature-statistic-based, test-performance-based, and fine-tuning-based similarity estimations have been proposed and studied widely [78]. However, further work is still required to provide a theoretical answer on how to quantify the domain similarity and its effects on the selection and design of TL frameworks [30,43].
- *Source and target data size:* When ML models (i.e., network structures) and source/target domains are determined, the TL performance would be affected by source and target data sizes [79]. For instance, an overfitting problem could be observed when the target data size is small, resulting in a set of sub-optimal parameters. Also initializing a target model from a source model trained with a large source dataset would improve the target learning performance and reduce the risk of overfitting. However, it is difficult to determine how many target and source data are enough for TL [80]. Especially in real-world engineering problems with expensive experiments or simulations, where the source data size has been determined according to completed source tasks, how much target data should be generated is still uncertain for different problems. From the authors' experiments, the target training data size should be 20~70% of the source data, to reach a balance between TL performance and the expense of data collection. More details are discussed in Section 4.

3. Transfer learning in AM modeling

In this section, the definition of the reviewed topic is presented for clarification, after which the literature review about TL in AM modeling is organized and discussed.

3.1. Problem definition

Based on the general definitions in TL, the specific counterparts in AM modeling are provided.

- *Domain* $\mathscr{D}$: For a certain AM product, the domain refers to the manufacturing setting, including the choice of printer machine, material, AM process, process parameters, etc. In each domain, the "*input variable* $\boldsymbol{X}$" is a set of parameters whose relationship with product properties is to be studied.
- *Task* $\mathscr{T}$: The AM task is to construct a product with certain requirements, such as mechanical properties and *in-situ* manufacturing qualities. The property/quality of interest is defined as the "*output variable* $\boldsymbol{Y}$", represented as a function of input variables $\boldsymbol{X}$, i.e., $\boldsymbol{Y} = f(\boldsymbol{X}|\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is model parameters to be learned from $(\boldsymbol{X}, \boldsymbol{Y})$.
- *Source/Target*: The AM source refers to available completed products, while the AM target refers to new similar-but-not-identical products to be fabricated with the assistance of knowledge extracted from the AM source.

With available information $(\boldsymbol{X}^s, \boldsymbol{Y}^s, f^s)$ from source products, where the superscript 's' indicates the source, the reviewed topic in this paper is *how to use them and TL methods to improve the AM modeling performance for target products in other domains, where only limited target data* $(\boldsymbol{X}^t, \boldsymbol{Y}^t)$ *are accessible*. To find all relevant papers possible, the authors searched with keywords "*additive manufacturing*", "*modeling*", and "*transfer learning* (or *knowledge transfer*)" in Google Scholar to cast a larger net. Then a manual selection is done carefully to find highly-cited relevant papers, and ignore irrelevant papers. For instance, a TL-based multi-fidelity point-cloud neural network was proposed to predict the thermal profile of melt pools, where the source model was constructed based on the low-fidelity data from analytical models [81]. Then the target model was obtained by fine-tuning the source model with high-fidelity simulation data. However, both low-fidelity and high-fidelity data are generated from the same domain. Therefore, this work is closer to multi-fidelity modeling than the reviewed topic. Similar works involving TL within the same AM domain could be found (i.e., Xia *et al.* [82], Liao *et al.* [83], etc.), but these works are not reviewed here. In other words, only works involving different domains will be reviewed in this paper. Besides, some papers applied TL to improve the accuracy of material microstructure reconstruction models [84,85], but the connections between the material and AM are not mentioned. Therefore, these works are also ignored, although the material design is an important area in AM. Based on these considerations, the final selected papers are summarized from Tables 1–5, and the details are presented as follows.

3.2. Instance-based TL

Considering the strict requirement of the same input space in both source and target domains, the instance-based TL only has few applications in quality modeling of AM, shown in Table 1.

Zhang *et al.* [39] integrated the GPR model with the T2ABR2 framework to predict electrical performances (i.e., the line width, line thickness, and edge roughness) of aerosol jet printing (AJP) under various operating conditions. With abundant data from the source operation condition (e.g., print velocity, tip diameter, atomizer current, and standard cubic centimeters per minute) and limited data from the target operation condition, the target model can predict three performances with a mean relative error smaller than 20% in most cases. However, such a large error is not acceptable, as a 10% error is the threshold to accept a model for optimization and path planning [39].

Table 1
Summary of instance-based TL in AM modeling.

Source domain	Target domain	Input x	Output y	AM	Ref.
• printed line with operation condition 1	• printed line with operation condition 2	• sheath gas flow rate • carrier gas flow rate	• line width • thickness • edge roughness	AJP	[39]
• GP1 powder + EOS M270 machine • AISI-630 SS powder + M2 Laser CUSING machine • GP1 powder + in-house developed SLM machine	• PS4542A powder + ProX100™ machine	• laser power • laser velocity • hatch spacing • layer thickness	• relative density	SLM	[40]

Table 2
Summary of feature-based TL in AM modeling.

Source domain	Target domain	Input x	Output y	AM	Ref.
• printed line with operation condition 1	• printed line with operation condition 2	• sheath gas flow rate • carrier gas flow rate	• line width • thickness • edge roughness	AJP	[39]
• product with geometry 1	• product with geometry 2	• point location	• shape-dependent deviation	FDM	[41]

Compared with feature-based and model-based TL frameworks, this instance-based TL framework performs worst in their case studies.

Similarly, an extended sequential minimum energy design method was proposed to utilize data from prior studies of the laser-based AM (LBAM) process [40]. In this method, the Bayesian updating framework was adopted to construct the target model, based on the source data collected from published papers (different steel powders and different machines) with increasing target data. Specifically, at the i-th iteration, the target model was constructed on source data and collected $(i-1)$ pieces of target data. An optimization was then performed based on the target model to get the process parameters for printing at the current iteration. Their case study showed that this modeling framework enables obtaining target products with over 99% relative density after five trials of target printing, which illustrates the efficiency of the designed instance transfer mechanism to solve process optimization problems.

3.3. Feature-based TL

Similar to the instance-based TL, there exist few applications of feature-based TL that are relevant to quality modeling, shown in Table 2.

Considering the discrepancy between source and target data sizes in real-world AM cases, Zhang *et al.* [39] applied a simplified transformation matrix [86] to solve the optimization problem in subspace alignment (shown as Eq. (1)), whose new solution is formulated as:

$$H^{s\to t} = \left(A^T A\right)^{-1} A^T B \tag{5}$$

where $A = [\boldsymbol{X}^t, f^s(\boldsymbol{X}^t), \boldsymbol{I}]_{n^t\times(n_d+2)}$, $B = [\boldsymbol{Y}^t]_{n^t\times 1}$, f^s is the pre-trained source model, and n_d is the dimension of the input variable. Then the initial source feature representation $\boldsymbol{D}^{s,0} = [\boldsymbol{X}^s, \boldsymbol{Y}^s, \boldsymbol{I}]_{n^s\times(n_d+2)}$ is used to construct the transformed source data $\boldsymbol{D}^{s,1} = \left[\boldsymbol{X}^s, \boldsymbol{D}^{s,0}H^{s\to t}\right]_{n^s\times(n_d+1)}$, where $\boldsymbol{D}^{s,0}H^{s\to t}$ can be regarded as the transformed source outputs. Finally, $[\boldsymbol{X}^t, \boldsymbol{Y}^t]$ and $\boldsymbol{D}^{s,1}$ are fed to the GPR model integrated with T2ABR2 to construct the target model. This combination of feature-based and instance-based TL frameworks brings better modeling accuracy than only using the instance-based TL in their case study.

Different from the above asymmetric feature-based method, Cheng *et al.* [41] decomposed the geometry deviation of FDM products into shape-dependent and shape-independent parts. In this method, the shape-dependent part of various geometries was assumed to be located in a common latent space about the local shape features, whose formulas were learned from data collected from different geometries. After training, this model could predict the shape-dependent deviation of new geometries based on boundary point positions, which could be used in property control to design compensations for geometry deviations.

3.4. Model-based TL

Compared with the instance-based and feature-based TL, the model-based TL has wider applications in AM modeling, summarized in Tables 3 and 4.

3.4.1. Source model ensemble & transformation

The idea of source model ensemble applied in AM modeling is to directly reuse the source model whose corresponding domain has the largest similarity with the target domain. One application is about FDM products' geometry deviation, which is divided into shape-independent and shape-dependent ones [41,87]. Both works approximated shape-independent components by deviations of points inside various products, based on the assumption that the shape-independent deviation model would be identical for different products. But this assumption is risky and hard to extend to other different AM modeling problems. Even for the geometry deviation prediction problem, the validity of this assumption should be checked by experiments.

The idea of constructing the target model as a transformation of the source model is mainly applied in modeling in-plane or out-of-plane deviations of AM products. For instance, to improve the modeling accuracy and efficiency, Sabbaghi and Huang [88] proposed a model transfer framework for SLA products involving different lurking variables $\boldsymbol{x}_{lur}$ which refers to those that are uncontrollable but affect the interested responses. In this framework, the target model f^t was defined as a transformed source model with the function $T_{t\to s}$:

$$f^t = f^s\left(T_{t\to s}\left(\boldsymbol{X}^t, \boldsymbol{x}^t_{lur}\right), \boldsymbol{x}^t_{lur}\right) \tag{6}$$

where the formula of $T_{t\to s}$ is defined manually according to prior knowledge, and the parameters can be learned from limited data of target products. Based on the same framework, Francis *et al.* [37] studied LPBF products fabricated by the EOS M290 machine with two different materials, Ti-6Al-4 V alloy and 316 L stainless steel. In their work, $T_{t\to s}$ was defined according to material differences (i.e., thermal conductivity), product size difference, and in-plate point locations, which captured the effect of domain differences on the point locations. As stated in [37], it is promising to consider other material information or process difference in $T_{t\to s}$, which could improve the transformation accuracy. At the same year, Francis [36] extended their work to transfer models among different machines (i.e., EOS M290 and MSU Renishaw AM 400) with the same material. Such a transformation framework is promising, but the accuracy of the final target model depends on the reliability of the designed $T_{t\to s}$, which requires expert knowledge about differences and relationships between 2 AM domains.

Similarly, assuming that AJP processes with different operating conditions would follow a similar material deposition mechanism and a

Table 3
Summary of model-based TL in AM with AM-related dataset.

Source domain	Target domain	Input x	Output y	AM	Ref.
• printed line with operation condition 1	• printed line with operation condition 2	• sheath gas flow rate • carrier gas flow rate	• line width • thickness • edge roughness	AJP	[39]
• product with geometry 1	• product with geometry 2	• point location	• shape-independent deviation	FDM	[41, 87]
• product with process 1 • products with Ti-6Al-4 V • products fabricated with EOS M290 machine	• product with process 2 • products with 316 L stainless steel • products fabricated with MSU Renishaw AM 400 machine	• point location with lurking variables	• in-plane deviation • in-plane deviation • in-plane deviation	SLA LPBF LPBF	[88] [37] [36]
• LBPF products	• BJ products	• gray images	defect label and location • Powder • Part • Defect	BJ	[90]
• Prima Power Laserdyne 430 3-axis CNC machine with a four-nozzle Optomec DMD head	• 5-axis DMD machine prototype ("Symbionica")	• laser power • scan speed • curvatures in the printing path	• deposition height	DMD	[91]
• bead-on-plate process	• bead-on-powder process	• laser power • scan speed	• melt pool size	LPBF	[38]
• products by printers with a kinematic model	• products by printers with different kinematics	• feed speed • acceleration	• line width	FDM	[92]
• products with initial process and shape	• products with different processes and shapes	• point location with lurking variables	• in/out-of-plane deviation	SLA	[93, 94]
• historical surfaces with HDM measurements	• new surfaces with LDM measurements	• multi-definition metrology data	• surface height • surface roughness	FDM	[95]
• material 316 L stainless steel	• material CuSn8	• spectrogram images of acoustic signals	• balling • lack of fusion pores • conduction mode • keyhole pores	LPBF	[96]
• low-carbon steel • stainless steel 316 L	• stainless steel 316 L • Inconel 625	• current/voltage data • video frame data • thermal properties of the material	anomaly detection: • normal bead shape • abnormal bead shape	WAAM	[97]
• continuous and short carbon fibers	• continuous carbon fiber • continuous glass fiber and short carbon fiber	• structure parameters • time	• strain-stress curve	FFF	[98]

similar overall trend of the response surface, Zhang *et al.* [39] proposed a general extension framework for the source model, shown as:

$$f^t = M_1 f^s(M_2 \boldsymbol{X}^t + B_2) + B_1 \tag{7}$$

where (M_1, B_1) are the global scale vector and shift vector respectively in the source response surface, (M_2, B_2) represents the local scale and shift vector respectively to map the target feature space $\mathscr{X}^t$ to source feature space $\mathscr{X}^s$. With the optimal transformation vector (M_1, B_1, M_2, B_2) obtained by minimizing the target prediction error with a genetic algorithm (GA), the extended model outperformed the instance-based method and falls behind the feature-based method in terms of the prediction accuracy of line width, thickness, and edge roughness. Based on Eq. (7), the target model in their work could be regarded as a linear transformation of the source model. However, this modeling framework would fail when the relationship between the target and the source domain is nonlinear.

3.4.2. Fine-tuning framework

Apart from the above two methods, another commonly applied method is the fine-tuning framework. From the perspective of involved source domains, its applications in AM can be classified as AM-related and AM-unrelated (i.e., ImageNet [89]) datasets.

- *AM-related dataset*

For fine-tuning frameworks with AM-related datasets, most applications aim to explore the transferability from process to process, from machine to machine, from product to product, and from material to material.

To investigate the performance of TL among different processes, Pandita *et al.* [38] first constructed a multi-fidelity GPR model based on high-fidelity and low-fidelity data collected from the bead-on-plate process with Inconel 625 material. Then a probabilistic deep neural network was trained based on abundant data generated by the regression model. For the bead-on-powder process with the same material, the former three layers of the trained network were frozen and the last several layers were retrained with limited new data. Similarly, Mehta and Shao [90] trained a source defect detection model for LPBF under the federated learning framework and transferred it to the binder jetting (BJ) process by fast fine-tuning only with four images. The final target model is acceptable for defect detection, although the accuracy degrades a little compared with that of the source model in LBPF. Both works demonstrate that the knowledge of different AM processes could be transferred to other similar-but-different processes. But more work should be conducted to explore their capabilities further.

To facilitate designing the geometric deviation compensation strategy among different machines of direct metal deposition (DMD) process, Knuttel *et al.* [91] designed a conventional CNN model to predict the deposition height based on laser power, scan speed, and seven curvatures in the single-layer printing path. After pre-training the model with 43,921 data collected from a Prima Power Laserdyne 430 3-axis CNC machine with a four-nozzle Optomec DMD head, the model was fine-tuned with 14,000 pieces of data collected from a 5-axis DMD machine prototype ("Symbionica") and reaches a root mean square error (RMSE) around 0.017 mm. Although this method works well in tests, the differences between the two machines are not considered. As a result,

Table 4
Summary of model-based TL in AM with AM-unrelated dataset.

Source domain	Target domain	Input x	Output y	AM	Ref.
• ImageNet dataset	• optical tomography/melt pool monitoring sensor data	• AM-related images obtained in the target domains	• melt pool performance	DMLS	[104]
	• images from laser scribes		quality characteristics • debris • scribe width • scribe straightness	LS	[106]
	• digital camera figures		• porosity defect	SLM	[42]
	• quality spectra		• quality fluctuation	–	[107]
	• layer-to-layer sensor images		• Discontinuity • Nonuniformity • irregularity	3D-bioprinting	[108]
	• SEM images of surface fracture		• vertical fracture • horizontal fracture	PBAM	[105]
	• Line camera images about powder bed		powder bed quality: • balling • incomplete spreading • groove • ridge • spatters • protruding part • scattered powder	LPBF	[109]
	• camera images about FDM products		• spaghetti-shape error	FDM	[114]
	• camera images of melt pool		anomalies of melt pool • robot suspend • normal • humping • spattering	WAAM	[113]
• COCO dataset	• camera images of FFF products		warp deformation: • Detection • localization	FFF	[110]

Table 5
Summary of MTL in AM modeling.

Source domain	Target domain	Input x	Output y	AM	Ref.
• products with different geometries and processes		• sensor signal feature • manufacturing feature	• in-plane deviation	SLM	[115, 117]
• products with different shapes and sizes		• point location,	• in-plane deviation	FDM	[118]
• products with different process parameters		• optical signal features	• porosity	LDED	[119]
• interconnected printers on a cloud platform		• speed • acceleration	• printing line variations • infill defects	EAM	[120]
• LPBF products fabricated by different clients		• gray image	defect label and location • Powder • Part • Defect	LPBF	[90]

the design framework could fail when testing on different machines. Different from the common fine-tuning framework, Ren *et al.* [92] proposed a transfer mechanism according to the kinematic properties of different FDM printers. They separated the printing process into five phases according to the kinematic features and defined a five-phase model to predict the line width. The parameters in the model were assumed to follow a Gaussian Process with a zero mean and a non-zero covariance. After training the source model and obtaining the source covariance Σ^s, the target covariance was defined as a scaled source one with a parameter q, i.e., $\Sigma^t = q\Sigma^s$. When training the target model, the scaled covariance provided an auxiliary minimization objective about parameters, i.e., $\min\left\|f^t\left(X^t|\boldsymbol{\theta}^t\right) - \boldsymbol{Y}^t\right\|_2^2 + \left(\boldsymbol{\theta}^t\right)^{\mathrm{T}}\left(\Sigma^{\mathrm{t}}\right)^{-1}\boldsymbol{\theta}^t$, which contributed to the learning of the target model.

In real-world AM applications, different products are often constructed under the same AM process. To explore the knowledge among different SLA products with different geometries, Sabbaghi *et al.* [93] adopted a manually defined formula of the geometry deviation as the source model and defined it as the global deviation feature for any target products. Then, the target model was structured as a combination of the source model and a new model f', i.e., $f^t = f^s + f'$. The purpose of f' was to capture the specific deviation feature of a target product from its limited data, which captures the differences between the target geometry and the common latent geometry. Similarly, Ferreira *et al.* [94] constructed a source Bayesian extreme learning machine to predict the in/out-of-plane deviation at each boundary point. Then a new Bayesian extreme learning machine was selected as f', whose outputs were integrated with the hidden and output layers of the source model to form the final target prediction. In addition, considering the expensive high-definition metrology measurements for surface quality inspection in FDM, Ren and Wang [95] proposed a TL-based surface variation modeling framework, where a source regression model and a source neural network model were trained for surface height and roughness respectively. When training the target regression model, the target prediction loss was combined with differences among model parameters $\left\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^s\right\|_2^2$, which posed a constraint on the target model parameter. For the target neural network, the former three layers in the source model were shared and frozen, while the remaining target layers were tuned with the limited target FDM data. These three works only consider products with simple geometries during their tests, as an arbitrary geometry would bring new challenges and require delicate designs of the transfer mechanism.

For knowledge transfer among materials, Vigneashwara *et al.* [96] designed a TL-based classification framework to detect defects in LPBF, i.e., *balling*, *lack-of-fusion pores*, *conduction mode*, and *keyhole pores*. Two CNN models (VGG16 and ResNet18) of the source were first trained based on the spectrogram images of acoustic signals, collected from line tracks with stainless steel 316 L. Then, the target models were obtained by fine-tuning the last several layers of both source models with the target data of spectrogram images of bronze (CuSn8). In other words, the

target images were used to update the fully connected layers in the source VGG16, as well as the last three convolution layers and the fully connected layer in the source ResNet18. During their experiments, both target models trained with less time and data can reach a comparable accuracy with those trained with full-size data and longer time. However, the prediction accuracy of such a homogeneous model structure would degrade when transferring between various materials, as different materials could demand different model complexity. To mitigate this problem, Shin *et al.* [97] developed an anomaly detection model including three modules, i.e., (i) feature extraction module (i.e., DenseNet169) to learn features from time-series numerical current/-voltage data and video data during WAAM process, (ii) material property concatenation module to consider primary thermal properties (i.e., thermal conductivity, melting point, and specific heat capacity) as partial inputs, and (iii) classification module to detect whether the bead shape is *normal* or *abnormal*. After constructing the source model, the target model could be obtained by reusing source parameters in the former two modules and fine-tuning the classification module based on limited data about the new material. Their classification model achieved an accuracy about 82.95%, 88.55%, and 84.22% when transferring knowledge from low-carbon steel to stainless steel 316 L, from low-carbon steel to Inconel 625, and from stainless steel 316 L to Inconel 625, respectively. Both works [96,97] only test on single-line track or single-layer bead deposition, however. The interaction between layers is not considered in their model, which could limit their capability in printing products with complex geometries. To further explore the applicability of TL in fused filament fabrication (FFF) products with different materials, Zhang *et al.* [98] proposed an optimal transport integrated long short-term memory model to predict the stress-strain curves based on product structures under the fine-tuning framework. In the designed experiments, i.e. transferring from continuous and short carbon fibers to (i) continuous carbon fiber or (2) continuous glass fiber and short carbon fiber, this model can reach an RMSE below 5.7% in both tests.

- *AM-unrelated dataset*

Apart from the above works whose source domain is related to AM, the general-purpose and high-performance CNN backbones (i.e., AlexNet [99], ResNet [100], VGG Net [101], GoogLeNet [102], EfficientNet [103], etc.) trained on the ImageNet database [89] have attracted attentions from AM researchers recently. All these models are transferred to construct the image-property model of AM, as sensor/digital images are available during the *in-situ* monitoring and control, with the assumption that the feature extraction layers in CNN backbones can be transferred to other tasks including AM.

Generally, most CNN backbones are used in AM-related classification problems, such as defect detection and quality control. For instance, with images from the optical tomography sensor or the melt pool monitoring sensor respectively, the pre-trained AlexNet can predict the melt pool performance (i.e., "*good*", "*under-melting*", or "*over-melting*") in direct metal laser solidification (DMLS) after modifying the network structure and retraining [104]. In the test cases, both fine-tuned AlexNet models have an overall accuracy of over 70% and their accuracies increase to 90% after performing a cross-validation scheme.

To detect the porosity of products during the SLM process, Li *et al.* [42] used a high-resolution digital camera to collect the layer-wise visual images, which were fed to a restructured VGG16 model. The parameters and structures of the pre-trained VGG16 model were frozen and the output layer was modified according to the target classes. Compared to the VGG16 model trained from scratch, the fine-tuned model reached a comparable average accuracy of about 99% with only half the training time. By removing the softmax layer in the output, Andrew and Elizabeth [105] utilized the pre-trained VGG16 model to extract features from microstructure images, which were then fed into the unsupervised learning methods (t-distributed Stochastic Neighbour Embedding and k-means) for classification. Based on the In-718 Charpy Fracture Surface dataset about powder bed additive manufacturing (PBAM) products, this framework could classify surface fractures into "*vertical*" and "*horizontal*" categories with 88.4% accuracy. By reusing the feature extraction layers in the pre-trained VGG16 model, Bisheh *et al.* [106] designed some new convolution layers to construct a quality monitor model, whose parameters were learned from 14 images of a laser scribing (LS) process. This model could reach an overall accuracy (96%) to identify debris, part background, and scribe line. And the output pixels could be used to determine the width and straightness of the scribe line.

Hu *et al.* [107] proposed a TL-based quality spectra fluctuation model, where the output layer of ResNet34 was modified according to the quality spectra categories. During the fine-tuning process, the former layers were frozen and the weights of the remaining layers were learned from the constructed quality spectra sets. Compared with AlexNet and ResNet34 trained from scratch, the TL-based ResNet34 had a better model accuracy of over 97% on the test dataset. As the same fine-tuning method in [107], the pre-trained ResNet50 was updated with layer-by-layer sensor images, to detect the disparity between the desired designs and the printed shapes during the 3D bioprinting process [108].

Considering the effects of powder bed homogeneity on final part qualities, Felix *et al.* [109] proposed a powder bed quality classification model based on the pre-trained Xception network. The output layers of the pre-trained CNN model were redesigned and fine-tuned based on over 45,000 images obtained by line cameras in eight LPBF experiments. After training, the classification accuracy 99% was observed in the proposed model, which is promising to provide reasonable guidance during the LPBF process, i.e, adapting the powder amount or recoater speed to correct some inhomogeneities (i.e, *scattered powder*, *ripples*, *grooves*, and *incomplete spreading*), replacing the recoating tool if *ridges* exist, or stopping the process early to save resources when *balling*, *spatters*, and *protruding part* occurs.

To correct or prevent warp deformation in FFF, Brion *et al.* [110] trained an object detection model based on the YOLOv3 model [111] which is initialized by learning from a large-scale COCO dataset [112]. This pre-trained model was then fine-tuned by thousands of FFF images to detect and localize the warp deformation, with a detection accuracy of 88.72%. Integrated with expert-informed heuristics, this model enables estimating the warpage (i.e., *minor*, *moderate*, *major*), which is used to analyze and correct printing and slicing parameters to reduce the warp in future prints.

Different from the above works that only applied one certain CNN structure, Xia *et al.* [113] selected four CNN backbones (i.e., ResNet, EfficientNet, VGG16, and GoogLeNet) for anomaly detection in WAAM. In their work, all models were updated with the same fine-tuning strategy, where all convolutional layers were fixed and all fully connected layers were tuned based on limited WAAM images. Their test results showed that all selected models can identify the robot suspend, normal, humping, and spattering with an accuracy over 97% after fine-tuning, which demonstrates their effectiveness. But some model hyperparameters need careful selections to reach such a high accuracy given a certain problem; otherwise, the classification accuracy after fine-tuning is not acceptable, i.e., 14.09% accuracy in ResNet with learning rates 0.1 and 0.05.

To further understand the effects of fine-tuning strategies on different models, Kim *et al.* [114] proposed a systematic deep TL method considering four ImageNet pre-trained CNN backbones (i.e., VGGNet, GoogLeNet, ResNet, EfficientNet) to detect the spaghetti-shape error in FDM process. A systematic learning matrix was designed by dividing structures of pre-trained models into five training sections, based on which different fine-tuning strategies could be defined. During their experiments, all models with various fine-tuning strategies receive higher classification accuracies than their corresponding baseline models. More importantly, their comparisons demonstrate that given a certain problem, different models with different fine-tuning strategies

should be selected carefully to reach a balance between accuracy, computing time, and memory size.

Although the model-based TL methods have the most applications, the above works only involve a narrow application scenario. For instance, the source model transformation is mainly applied in geometry deviation compensation problems, and most applications of ImageNet-based fine-tuning framework are AM-related classification problems, i.e., *in-situ* process monitoring and defect detection to determine whether there is a defect, or which type the defect belongs to. The potential of these frameworks needs to be explored further by solving other AM modeling problems.

Although all these TL-based CNN models could be tailored and applied to various AM classification problems, it is seldom discussed in the literature about which CNN model should be trained on and which large-scale dataset should be selected. This would give readers a misleading message that all pre-trained CNN models have potential to solve all AM classification problems, regardless of differences between AM tasks and others (i.e., ImageNet and COCO). To guarantee the TL performance for different AM problems, the differences should be considered and discussed further.

3.5. Multi-task learning

Inspired by MTL, Wang *et al.* [115] proposed a Family Learning framework to efficiently use limited data of each SLM product. In this method, the geometric deviation of each layer in one product was assumed as a linear function of the signal features of the corresponding layer. During the learning process, the regularizer contained two parts, i) the LASSO regularization term [116] that makes parameters of all insignificant variables zero; and ii) a penalty function about the difference between each model parameter and the weighted average parameter of all models, which enables more information to be collected from other models. To share more information among relevant products, a similarity coefficient was defined based on the manufacturing features, including the scanning path and process settings. Given a new product, the pre-trained model of the product with the largest similarity is reused directly to predict the geometry deviation of each layer. Similarly, the Family Learning framework is also applied to SLM products with different geometries and processes [117], where l_1 norm and l_2 norm are both applied in order to encourage learning from similar products.

Zhu *et al.* [118] developed an MTL-based GPR framework for FDM products with different shapes (circular, regular pentagon, and regular hexagon) and sizes (small, medium, and large). The relationship between the in-plane deviations and point locations is captured by a GPR model, but the model parameters are learned without the regularization of model parameters in the objective function. For products manufactured by the laser-directed energy deposition (LDED) method, Sun *et al.* [119] applied the MTL framework to train a set of semi-supervised clustering methods to predict the part quality according to the optical spectrum signal features. Two penalty regularization terms are adopted in the training, including i) the penalty about the weighted average of the difference between the objective quality labels and the predicted labels, and ii) the penalty of the pairwise disparity between cluster parameters. In the test case, the clustering accuracy on a new product reached 100% with a set of training data occupying only 3.6% of the dataset, which demonstrates the effectiveness of the work [119].

To solve the quality control problem efficiently in extrusion-based AM (EAM), Wei *et al.* [120] extended the TL-based kinematic-quality model [92] to a more generalized one to measure the relatedness between processes. Based on the calculated relatedness, a bidirectional and unidirectional co-learning framework was proposed for interconnected printers on a cloud platform, where each printer only performed a limited number of experiments. This modeling framework used limited data from various printers to train the quality prediction model simultaneously, under a proposed hybrid metaheuristic training process. In their tests, the co-learning model outperforms the TL-based model in terms of prediction accuracy and performance improvement compared with the baseline model.

Mehta and Shao [90] designed a semantic segmentation model based on the U-Net structure for defect detection in LPBF under the federated learning framework. Their work aimed to combine limited data from different clients (i.e., machines and manufacturers) to alleviate the data scarcity problem. During the training procedure, a global model was sent to each client for pre-training, after which the model parameters from all clients were averaged to define the new global model until convergence. Such a model shows better defect detection performance than conventional centralized learning and individual learning for gray images of products fabricated by ConceptLaser M2 LPBF machine with Stainless Steel 316 L powder. However, this work considers all defects as one classification label, which restricts its applicability when specific defect types are of interest.

Different from sufficient source data in the applications reviewed in Sections 3.2–3.4, the reviewed applications of MTL focus on problems where each AM domain only has limited data available. But their applicability is restricted by the number of found AM domains. If the number is smaller, the data scarcity problem also exists and deteriorates the modeling performance.

3.6. Summary remarks

With available datasets of AM products from publications or online repositories, different TL frameworks have been successfully explored to improve the modeling performance and reduce the requirements of expensive data about new products. However, current studies have two common limitations.

- Different from that in computer science and fault diagnosis [35], the applications of TL in AM modeling are relatively few. Current studies focus on process-structure-property modeling for performance prediction (or optimization), process control, and defect detection, where each aspect has few relevant works. Therefore, how to apply TL widely in the whole AM lifecycle still requires more exploration.
- Most applications have the simple assumption that the target and source domains are relevant to some extent but the similarity between domains is seldom discussed. In real-world applications, it is important to measure these similarities quantitatively or qualitatively so that the appropriate TL frameworks could be selected according to their characteristics and capacities.

4. Discussions and recommendations

Unfortunately, the reviewed works seldom answer these questions: which source domain to use, which TL framework to adopt, how much target data we need, and whether to apply data preprocessing techniques for a given AM modeling problem. Consequently, when encountering new problems, researchers would spend a significant amount of time in trial-and-error to find the best choices. Therefore, this section aims to present some relevant discussions to shed light on these issues.

4.1. Which source domain to use?

As mentioned in Section 2.3, the similarity between the source and target domains has a great impact on TL performances. When studying a target AM modeling problem, the first work is to determine which source domain could be selected for data collection. Generally, the higher the similarity, the better the source domain. However, the similarity between two domains is difficult to quantify, especially when only few data are accessible for the target AM problem. Therefore, qualitative similarity could be an alternative criterion to tell us where to find possible relevant source data. In other words, given a certain target AM problem, it is better to find source domain data from products fabricated

with geometries, materials, AM processes, or machines that belong to the same series as the target product. But in real-word applications, qualitative similarity should be carefully defined considering both source and target domains, as it represents a different metric compared with quantitative similarity. Besides, to identify which part of the source data should be transferred, a quantitative similarity would be more helpful, which however needs delicate design and further development.

4.2. Which TL framework to adopt?

After determining the source domain, the matching TL frameworks for real-world target AM problems could be found by comparing source and target domains in detail, according to the underlying characteristics discussed in Section 2.3:

- $\boldsymbol{X}^s = \boldsymbol{X}^t$ and $\boldsymbol{Y}^s = \boldsymbol{Y}^t$: When the source and target problems involve the same input (i.e., process parameters) and output (i.e., product properties) variables, conventional modeling methods (i.e., GPR) integrated with instance-based TL or source model transformation frameworks are preferred. The reason is that compared with various TL-based ANN models, they have fewer parameters to be learned from limited target data, which reduces the risk of negative effects caused by insufficient target data.
- $\boldsymbol{X}^s \subseteq \boldsymbol{X}^t$ or $\boldsymbol{X}^s \supseteq \boldsymbol{X}^t$: Sometimes, the accessible source AM task considers more/fewer input variables than the target AM task, such as thermal predictions with or without considering material properties. For these AM problems, the feature-based and model-based TL frameworks would perform better, as both frameworks can tackle the disparity among $\mathscr{X}^s$ and $\mathscr{X}^t$.
- $\boldsymbol{X}^s = \boldsymbol{X}^t$ and $\boldsymbol{Y}^s \neq \boldsymbol{Y}^t$: In some cases, the accessible source data and target data involve the same input variables but different outputs, such as constructing the relationships between process parameters and melt pool geometries (i.e., height, weight, and depth). In these problems, the feature-based TL, model-based TL, and MTL are all applicable as they share common knowledge about the input feature space by different mechanisms.
- Few $(\boldsymbol{X}^s, \boldsymbol{Y}^s)$: In reality, sufficient relevant source data are inaccessible for some target tasks. For example, when AM techniques are applied in a new industrial field, or a new product completely different from existing products is studied, obtaining sufficient source data would be difficult and expensive. In these cases, MTL would outperform another three TL frameworks. By utilizing insufficient data from several source domains, MTL can solve them simultaneously to reach an acceptable performance for each task with a lower average budget [75].

4.3. How much target data is needed?

Based on collected source data and selected TL frameworks, the next work is to determine how much target data (including training, validation, and testing data) should be generated for the AM modeling. According to the authors' knowledge and experience, the negative transfer could occur in two different data size conditions.

(i) When the amount of target training data is much less than that of the source training data (i.e., $n^t \ll n^s$), it would be difficult to extract relevant knowledge from the source data according to the scarce information in the limited target data. Therefore, a lack of sufficient target data could cause inappropriate source knowledge transfer, which in turn misleads the construction process and results in negative transfer.

(ii) When the target training data size is close to the source training data size (i.e., $n^t \approx n^s$), a target model could be constructed based on target data alone with acceptable accuracy. Then, the difference between source and target domains would be amplified when integrating source data, leading to negative effects on the training of the target model. Besides, generating target data with a similar size to the source data implies high costs, which is not preferable in TL-based applications.

Therefore, in real-world AM problems, the data size ratio n^t/n^s should be in an appropriate range to reach a tradeoff among computational expense, modeling accuracy, and TL performances. According to the authors' experiments, the ratio of target to source training data should be in the range of $[0.2, 0.7]$.

4.4. Whether to apply data preprocessing techniques?

Before feeding source and target data to the modeling process, data preprocessing techniques should be selected carefully according to the ML methods employed; otherwise, it would deteriorate learning performances [121]. For instance, given one source data $(\boldsymbol{x}^s, \boldsymbol{y}^s)$, one target data $(\boldsymbol{x}^t, \boldsymbol{y}^t)$, and $\boldsymbol{x}^s \neq \boldsymbol{x}^t$, the preprocessed source and target input variables may be too close or even the same after normalization (i.e., both being normalized to the same range [0,1]). As a result, some inherent information between this data pair would degenerate. When transferring source data as knowledge directly (i.e., the combination of source/target data in instance-based TL [39]) or comparing source and target data directly (i.e., space alignment in feature-based TL, and model scale/shift in model-based TL [39]), such a pair of normalized data would deteriorate modeling performances. Although both fine-tuning and multi-task learning frameworks transfer the implicit source knowledge represented as model parameters or structures, the negative transfer could also be observed if the data are normalized. Besides, when considering different data size ratio n^t/n^s, the effect of data preprocessing such as normalization would vary significantly.

Therefore, in real-world AM applications, whether to use data preprocessing techniques (i.e., normalization) should be determined according to which TL framework is used and how much target data are available.

5. Future research

According to the above discussions, future research on TL-assisted AM modeling can be conducted from the perspective of the modeling framework and the type of knowledge to be transferred.

5.1. Modeling framework

As discussed in Section 3, current studies only involve a few stages in the AM lifecycle, and the applied frameworks are correlated strongly to the research objectives. The potential of these frameworks is seldom explored in other problems. This would cause expensive and inefficient information transfer among different AM research objects when the AM lifecycle is studied. Hence, it is important to develop a general TL-assisted AM modeling framework, which covers most aspects of the AM lifecycle and can be tailored to specific problems.

Besides, current studies pay little attention to the effect of domain relevance on the choice and design of TL frameworks. As a result, the TL framework with great performance on test cases would perform worse on a new similar task. Therefore, one important future research is to construct a similarity quantification framework for different AM domains, which could provide some guidance for designing a TL-assisted modeling framework.

Another future research direction is online TL-assisted modeling. Different from offline TL with available target domains in advance, online TL aims to tackle real-world problems where the target domain is received sequentially during the learning procedure. Recently, it has been explored widely in classification problems containing multiple source domains [122,123], where the increasing online target data is applied to fine-tune the model gradually. However, current studies in

AM mainly train the target models offline and validate them with limited expensive data, after which the trained models are used directly in the target domain without any further modifications. This mechanism poses a risk to the rationality of target models, as the limited target data is unable to fully represent the product performance. To solve the problem, the online TL-assisted modeling framework will be helpful, due to its potential in improving the model performance online with *in-situ* data.

5.2. Knowledge to be transferred

According to the definition of knowledge by Wu and Wang [124], the information transferred in most relevant studies, such as data, model structures, and model parameters, belongs to algorithmic knowledge. Apart from this, there are several other kinds of knowledge in AM. For instance, expert experience and physical laws are used as a constraint to make the prediction model of FDM product qualities more reasonable [125] via training a physics-informed neural network [126]. The causal relationship between AM process parameters and product properties has been studied to help select the main parameters for monitoring certain performances [127,128]. However, these applications do not focus on knowledge transfer among different domains. Therefore, how to transfer different knowledge simultaneously is a promising research direction for AM modeling.

6. Summary

Unlike general-purpose reviews about machine learning in AM, this paper reviews a specific topic, i.e., *how to improve AM modeling performance by transfer learning techniques*. To clarify the topic, some notations, definitions, and categories in transfer learning are first discussed. Based on their underlying assumptions or theories, the instance-based methods are suggested for problems with the same input feature space, while the feature-based and model-based methods are recommended for problems involving different input/output feature spaces and conditional or marginal distributions. Without requirements of sufficient source data, multi-task learning is a better choice when few data are accessible in several source domains. Then some AM-specific notations and definitions are presented, and current works about transfer learning-assisted AM modeling are summarized according to four categories of transfer learning technologies along with their limitations. Based on the reviewed applications, for a target AM modeling problem, discussions are presented to answer questions including *which source domain to use, which TL framework to adopt, how much target data to use,* and *which data preprocessing technique should be selected.* Generally, the source domain could be determined according to qualitative similarities when a quantified similarity is hard to obtain. The transfer learning framework could be selected by mapping their preferred application scenario to AM: (a) the instance-based framework integrated with the conventional modeling method is recommended for AM problems involving the same input and output variables; (b) the feature-based and model-based frameworks are better for AM problems involving different input/output variables; (c) the above two frameworks and multi-task learning are applicable when same inputs but different outputs are studied in AM problems; (d) the multi-task learning may be the best choice when only insufficient data are accessible in several source AM problems. Besides, the target data size should be selected carefully according to the source data size, to balance the computational expense, modeling accuracy, and TL performances. And the data preprocessing techniques should be selected carefully considering the TL framework and target data size. Finally, future research directions are presented and discussed, including the general modeling framework for AM lifecycle, similarity quantification framework, online transfer learning-assisted modeling framework, and the application of various types of knowledge available in AM.

Based on the above discussions, the main contributions of this paper are summarized as follows:

- According to the knowledge of the authors, this review paper is the first one focusing on the utilization of transfer learning in AM modeling. In this paper, most papers relevant to this topic are summarized and critically reviewed according to the categories of transfer learning techniques.
- Based on the literature review, discussions and recommendations are given on the selection of the transfer learning framework, choosing the right source domain, determining the target data size, and selection of data preprocessing technique, to shed light on how to apply transfer learning in modeling AM processes.
- Future research directions are discussed concerning applying transfer learning in the modeling of additive manufacturing processes.

CRediT authorship contribution statement

Yifan Tang: Writing – original draft, Validation, Methodology, Investigation, Formal analysis, Conceptualization. **Mostafa Rahmani Dehaghani:** Resources, Methodology, Formal analysis, Conceptualization. **G. Gary Wang:** Writing – review & editing, Supervision, Resources, Project administration, Methodology, Funding acquisition, Conceptualization.

Declaration of Competing Interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Coauthors Mostafa Rhamani Dehaghani and Yifan Tang report financial support that was provided by Natural Sciences and Engineering Research Council of Canada.

Data Availability

No data was used for the research described in the article.

Acknowledgements

Funding from the Natural Science and Engineering Research Council (NSERC) of Canada under the project RGPIN-2019-06601, as well Graduate Dean's Entrance Scholarship (GDES) from Simon Fraser University, is appreciated.

References

[1] B.M. Colosimo, Q. Huang, T. Dasgupta, F. Tsung, Opportunities and challenges of quality engineering for additive manufacturing, J. Qual. Technol. 50 (2018) 233–252, https://doi.org/10.1080/00224065.2018.1487726.
[2] Y. Pang, Y. Cao, Y. Chu, M. Liu, K. Snyder, D. MacKenzie, C. Cao, Additive manufacturing of batteries, Adv. Funct. Mater. 30 (2020) 1–22, https://doi.org/10.1002/adfm.201906244.
[3] M. Salmi, Additive manufacturing processes in medical applications, Materials 14 (2021) 1–16, https://doi.org/10.3390/ma14010191.
[4] J.C. Vasco, Additive manufacturing for the automotive industry, in: Addit. Manuf., Elsevier Inc, 2021, pp. 505–530, https://doi.org/10.1016/B978-0-12-818411-0.00010-0.
[5] B. Blakey-Milner, P. Gradl, G. Snedden, M. Brooks, J. Pitot, E. Lopez, M. Leary, F. Berto, A. du Plessis, Metal additive manufacturing in aerospace: A review, Mater. Des. 209 (2021), 110008, https://doi.org/10.1016/j.matdes.2021.110008.
[6] J. den Boer, W. Lambrechts, H. Krikke, Additive manufacturing in military and humanitarian missions: advantages and challenges in the spare parts supply chain, J. Clean. Prod. 257 (2020), 120301, https://doi.org/10.1016/j.jclepro.2020.120301.
[7] A. Al Rashid, S.A. Khan, S.G. Al-Ghamdi, M. Koç, Additive manufacturing: technology, applications, markets, and opportunities for the built environment, Autom. Constr. 118 (2020), 103268, https://doi.org/10.1016/j.autcon.2020.103268.
[8] L. Flach, R.P. Chartoff, A simple polymer shrinkage model applied to stereolithography, Solid Free. Fabr. Symp. (1994) 225–233. http://hdl.handle.net/2152/68649.
[9] X.C. Wang, J.P. Kruth, A simulation model for direct selective laser sintering of metal powders, in: 5th Int. Conf. Comput. Struct. Technol. (2000) 57–71. https://dl.acm.org/doi/10.5555/372484.372491.

[10] P. Muller, P. Mognol, J.Y. Hascoet, Modeling and control of a direct laser powder deposition process for Functionally Graded Materials (FGM) parts manufacturing, J. Mater. Process. Technol. 213 (2013) 685–692, https://doi.org/10.1016/j.jmatprotec.2012.11.020.

[11] T. Chen, Y. Zhang, Numerical simulation of two-dimensional melting and resolidification of a two-component metal powder layer in selective laser sintering process, Numer. Heat. Transf. Part A Appl. 46 (2004) 633–649, https://doi.org/10.1080/104077890504177.

[12] P. Michaleris, Modeling metal deposition in heat transfer analyses of additive manufacturing processes, Finite Elem. Anal. Des. 86 (2014) 51–60, https://doi.org/10.1016/j.finel.2014.04.003.

[13] H. Bikas, P. Stavropoulos, G. Chryssolouris, Additive manufacturing methods and modelling approaches: a critical review, Int. J. Adv. Manuf. Technol. 83 (2016) 389–405, https://doi.org/10.1007/s00170-015-7576-2.

[14] N. Raghunath, P.M. Pandey, Improving accuracy through shrinkage modelling by using Taguchi method in selective laser sintering, Int. J. Mach. Tools Manuf. 47 (2007) 985–995, https://doi.org/10.1016/j.ijmachtools.2006.07.001.

[15] A.K. Sood, R.K. Ohdar, S.S. Mahapatra, Experimental investigation and empirical modelling of FDM process for compressive strength improvement, J. Adv. Res. 3 (2012) 81–90, https://doi.org/10.1016/j.jare.2011.05.001.

[16] M.H. Sehhat, B. Behdani, C.H. Hung, A. Mahdianikhotbesara, Development of an empirical model on melt pool variation in laser foil printing additive manufacturing process using statistical analysis, Metallogr. Microstruct. Anal. 10 (2021) 684–691, https://doi.org/10.1007/s13632-021-00795-x.

[17] N. Chikkanna, S. Krishnapillai, V. Ramachandran, Static and dynamic flexural behaviour of printed polylactic acid with thermal annealing: parametric optimisation and empirical modelling, Int. J. Adv. Manuf. Technol. 119 (2022) 1179–1197, https://doi.org/10.1007/s00170-021-08127-7.

[18] Z. Wang, W. Yang, Q. Liu, Y. Zhao, P. Liu, D. Wu, M. Banu, L. Chen, Data-driven modeling of process, structure and property in additive manufacturing: a review and future directions, J. Manuf. Process. 77 (2022) 13–31, https://doi.org/10.1016/j.jmapro.2022.02.053.

[19] L. Yi, C. Gläßner, N. Krenkel, J.C. Aurich, Energy simulation of the fused deposition modeling process using machine learning approach, Procedia CIRP 86 (2020) 216–221, https://doi.org/10.1016/j.procir.2020.01.002.

[20] A. Olleak, Z. Xi, Calibration and validation framework for selective laser melting process based on multi-fidelity models and limited experiment data, J. Mech. Des. Trans. ASME 142 (2020) 1–13, https://doi.org/10.1115/1.4045744.

[21] D. Ding, F. He, L. Yuan, Z. Pan, L. Wang, M. Ros, The first step towards intelligent wire arc additive manufacturing: an automatic bead modelling system using machine learning through industrial information integration, J. Ind. Inf. Integr. 23 (2021), 100218, https://doi.org/10.1016/j.jii.2021.100218.

[22] K. Ren, Y. Chew, Y.F. Zhang, J.Y.H. Fuh, G.J. Bi, Thermal field prediction for laser scanning paths in laser aided additive manufacturing by physics-based machine learning, Comput. Methods Appl. Mech. Eng. 362 (2020), 112734, https://doi.org/10.1016/j.cma.2019.112734.

[23] Z. Zhou, H. Shen, B. Liu, W. Du, J. Jin, Thermal field prediction for welding paths in multi-layer gas metal arc welding-based additive manufacturing: A machine learning approach, J. Manuf. Process. 64 (2021) 960–971, https://doi.org/10.1016/j.jmapro.2021.02.033.

[24] R. Liu, S. Liu, X. Zhang, A physics-informed machine learning model for porosity analysis in laser powder bed fusion additive manufacturing, Int. J. Adv. Manuf. Technol. 113 (2021) 1943–1958, https://doi.org/10.1007/s00170-021-06640-3.

[25] O. Aljarrah, J. Li, A. Heryudono, W. Huang, J. Bi, Predicting part distortion field in additive manufacturing: a data-driven framework, . J. Intell. Manuf. (2022) https://doi.org/10.1007/s10845-021-01902-z.

[26] N. Jamnikar, S. Liu, C. Brice, X. Zhang, Machine learning based in situ quality estimation by molten pool condition-quality relations modeling using experimental data, ArXiv Prepr. ArXiv2103.12066. (2021). http://arxiv.org/abs/2103.12066.

[27] S.S. Razvi, S. Feng, A. Narayanan, Y.T.T. Lee, P. Witherell, A review of machine learning applications in additive manufacturing, in: ASME Des. Eng. Tech. Conf., Anaheim, California, USA, August 18–21, 2019. V001T02A040., 2019. https://doi.org/10.1115/DETC2019-98415.

[28] N.S. Johnson, P.S. Vulimiri, A.C. To, X. Zhang, C.A. Brice, B.B. Kappes, A. P. Stebner, Invited review: Machine learning for materials developments in metals additive manufacturing, Addit. Manuf. 36 (2020), 101641, https://doi.org/10.1016/j.addma.2020.101641.

[29] L. Meng, B. McWilliams, W. Jarosinski, H.Y. Park, Y.G. Jung, J. Lee, J. Zhang, Machine learning in additive manufacturing: a review, JOM 72 (2020) 2363–2377, https://doi.org/10.1007/s11837-020-04155-y.

[30] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, Q. He, A comprehensive survey on transfer learning,, Proc. IEEE (2021) 43–76, https://doi.org/10.1109/JPROC.2020.3004555.

[31] M. Maqsood, F. Nazir, U. Khan, F. Aadil, H. Jamal, I. Mehmood, O.Y. Song, Transfer learning assisted classification and detection of alzheimer's disease stages using 3D MRI scans, Sens. (Switz.) 19 (2019) 1–19, https://doi.org/10.3390/s19112645.

[32] G. Schweikert, C. Widmer, B. Schölkopf, G. Rätsch, An empirical analysis of domain adaptation algorithms for genomic sequence analysis, in: Adv. Neural Inf. Process. Syst. 21 - Proc. 2008 Conf., Vancouver, BC, Canada, December 8–10, 2009, pp. 1433–1440.

[33] S. Di, H. Zhang, C.G. Li, X. Mei, D. Prokhorov, H. Ling, Cross-domain traffic scene understanding: a dense correspondence-based transfer learning approach, IEEE Trans. Intell. Transp. Syst. 19 (2018) 745–757, https://doi.org/10.1109/TITS.2017.2702012.

[34] L. Zhao, S.J. Pan, E.W. Xiang, E. Zhong, Z. Lu, Q. Yang, Active transfer learning for cross-system recommendation, in: 27th AAAI Conf. Artif. Intell. AAAI 2013, Bellevue, Washington, USA, July 14–17, 2013, pp. 1205–1211. https://doi.org/10.1609/aaai.v27i1.8458.

[35] H. Zheng, R. Wang, Y. Yang, J. Yin, Y. Li, Y. Li, M. Xu, Cross-domain fault diagnosis using knowledge transfer strategy: a review, IEEE Access 7 (2019) 129260–129290, https://doi.org/10.1109/ACCESS.2019.2939876.

[36] J. Francis, Transfer learning in laser-based additive manufacturing: Fusion, calibration, and compensation, Theses and Dissertations, Mississippi State University, US, 2020.

[37] J. Francis, A. Sabbaghi, M. Ravi Shankar, M. Ghasri-Khouzani, L. Bian, Efficient distortion prediction of additively manufactured parts using Bayesian model transfer between material systems, J. Manuf. Sci. Eng. Trans. ASME 142 (2020) 1–16, https://doi.org/10.1115/1.4046408.

[38] P. Pandita, S. Ghosh, V.K. Gupta, A. Meshkov, L. Wang, Application of deep transfer learning and uncertainty quantification for process identification in powder bed fusion, ASCE-ASME J. Risk Uncertain. Eng. Syst. Part B Mech. Eng. 8 (2022) 1–12, https://doi.org/10.1115/1.4051748.

[39] H. Zhang, J.P. Choi, S.K. Moon, T.H. Ngo, A knowledge transfer framework to support rapid process modeling in aerosol jet printing, Adv. Eng. Inform. 48 (2021), 101264, https://doi.org/10.1016/j.aei.2021.101264.

[40] A.M. Aboutaleb, L. Bian, A. Elwany, N. Shamsaei, S.M. Thompson, G. Tapia, Accelerated process optimization for laser-based additive manufacturing by leveraging similar prior studies, IISE Trans. 49 (2017) 31–44, https://doi.org/10.1080/0740817X.2016.1189629.

[41] L. Cheng, K. Wang, F. Tsung, A hybrid transfer learning framework for in-plane freeform shape accuracy control in additive manufacturing, IISE Trans. 53 (2020) 298–312, https://doi.org/10.1080/24725854.2020.1741741.

[42] J. Li, Q. Zhou, X. Huang, M. Li, L. Cao, In situ quality inspection with layer-wise visual images based on deep transfer learning during selective laser melting, . J. Intell. Manuf. (2021) https://doi.org/10.1007/s10845-021-01829-5.

[43] S.J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22 (10) (2009) 1345–1359, https://doi.org/10.1109/TKDE.2009.191.

[44] J. Huang, A. Gretton, K.M. Borgwardt, B. Schölkopf, A.J. Smola, Correcting sample selection bias by unlabeled data, Adv. Neural Inf. Process. Syst. 19 (2006) 601–608. https://dl.acm.org/doi/10.5555/2976456.2976532.

[45] W. Dai, Q. Yang, X. Gui-Rong, Y. Yong, Boosting for transfer learning, in: Proc. 24th Int. Confer- Ence Mach. Learn., Corvalis Oregon, USA, June 20–24, 2007, pp. 193–200. https://doi.org/10.1145/1273496.1273521.

[46] M. Sugiyama, T. Suzuki, S. Nakajima, H. Kashima, P. Von Bünau, M. Kawanabe, Direct importance estimation for covariate shift adaptation, Ann. Inst. Stat. Math. 60 (2008) 699–746, https://doi.org/10.1007/s10463-008-0197-x.

[47] Y. Yao, G. Doretto, Boosting for transfer learning with multiple sources, in: in: IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., IEEE, 2010, pp. 1855–1862, https://doi.org/10.1109/CVPR.2010.5539857.

[48] D. Pardoe, P. Stone, Boosting for regression transfer, in: Proc. 27th Int. Conf. Mach. Learn., Haifa, Israel, June 21–24, 2010, pp. 863–870. https://dl.acm.org/doi/10.5555/3104322.3104432.

[49] H. Drucker, Improving regressors using boosting techniques, in: Proc. 14th Int. Conf. Mach. Learn., 1997, pp. 107–115. https://dl.acm.org/doi/10.5555/645526.657132.

[50] B. Fernando, A. Habrard, M. Sebban, T. Tuytelaars, Unsupervised visual domain adaptation using subspace alignment, in: IEEE Int. Conf. Comput. Vis., Sydney, NSW, Australia. December 1–8, 2013, pp. 2960–2967. https://doi.org/10.1109/ICCV.2013.368.

[51] L. Duan, D. Xu, I.W. Tsang, Learning with augmented features for heterogeneous domain adaptation, in: Proc. 29th Int. Conf. Mach. Learn. ICML 2012, pp. 711–718. https://doi.org/10.48550/arXiv.1206.4660.

[52] S.J. Pan, J.T. Kwok, Q. Yang, Transfer learning via dimensionality reduction, Proc. Natl. Conf. Artif. Intell. (2008) 677–682. https://dl.acm.org/doi/10.5555/1620163.1620177.

[53] J. Blitzer, R. McDonald, F. Pereira, Domain adaptation with structural correspondence learning, in: Proc. 2006 Conf. Empir. Methods Nat. Lang. Process. (EMNLP 2006), Sydney, Australia, July, 2006, pp. 120–128. https://doi.org/10.3115/1610075.1610094.

[54] K.M. Borgwardt, A. Gretton, M.J. Rasch, H.P. Kriegel, B. Schölkopf, A.J. Smola, Integrating structured biological data by Kernel Maximum Mean Discrepancy, Bioinformatics 22 (2006) 49–57, https://doi.org/10.1093/bioinformatics/btl242.

[55] A. Gretton, O. Bousquet, A. Smola, B. Schölkopf, Measuring statistical dependence with Hilbert-Schmidt norms, in: Int. Conf. Algorithmic Learn. Theory, Springer, 2005, pp. 63–77, https://doi.org/10.1007/11564089_7.

[56] C.Y. Lee, T. Batra, M.H. Baig, D. Ulbricht, Sliced wasserstein discrepancy for unsupervised domain adaptation, IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (2019) 10277–10287, https://doi.org/10.1109/CVPR.2019.01053.

[57] S.J. Pan, I.W. Tsang, J.T. Kwok, Q. Yang, Domain adaptation via transfer component analysis, IEEE Trans. Neural Netw. 22 (2011) 199–210, https://doi.org/10.1109/TNN.2010.2091281.

[58] S.J. Pan, X. Ni, J.T. Sun, Q. Yang, Z. Chen, Cross-domain sentiment classification via spectral feature alignment, in: Proc. 19th Int. Conf. World Wide Web, WWW 10, 2010, pp. 751–760. https://doi.org/10.1145/1772690.1772767.

[59] X. Shi, Q. Liu, W. Fan, P.S. Yu, R. Zhu, Transfer learning on heterogenous feature spaces via spectral transformation, in: Proc. - IEEE Int. Conf. Data Mining, ICDM, IEEE, 2010, pp. 1049–1054. https://doi.org/10.1109/ICDM.2010.65.

[60] J. Yosinski, J. Clune, Y. Bengio, H. Lipson, How transferable are features in deep neural networks? Adv. Neural Inf. Process. Syst. (2014) 3320–3328. https://doi.org/10.48550/arXiv.1411.1792.

[61] N. Houlsby, A. Giurgiu, S. Jastrzębski, B. Morrone, Q. de Laroussilhe, A. Gesmundo, M. Attariyan, S. Gelly, Parameter-efficient transfer learning for NLP, in: 36th Int. Conf. Mach. Learn., PMLP, California, USA, June 9–15, 2019, pp. 2790–2799. https://doi.org/10.48550/arXiv.1902.00751.

[62] P. Li, H. Cui, A. Khan, U. Raza, R. Piechocki, A. Doufexi, T. Farnham, Deep transfer learning for WiFi localization, in: IEEE Natl. Radar Conf., IEEE, 2021. doi: 10.1109/RadarConf2147009.2021.9455237.

[63] J. Gao, W. Fan, J. Jiang, J. Han, Knowledge transfer via multiple model local structure mapping, Int. Conf. Knowl. Discov. Data Min. (2008) 283–291. https://doi.org/10.1145/1401890.1401928.

[64] L. Duan, I.W. Tsang, D. Xu, T.S. Chua, Domain adaptation from multiple sources via auxiliary classifiers, in: Proc. 26th Int. Conf. Mach. Learn. ICML 2009, 2009, pp. 289–296. https://doi.org/10.1145/1553374.1553411.

[65] T. Tommasi, B. Caputo, The more you know, the less you learn: From knowledge transfer to one-shot learning of object categories, Br. Mach. Vis. Conf. BMVC 2009 - Proc. (2009) 1–11. https://doi.org/10.5244/C.23.80.

[66] Y. Zhang, Q. Yang, A survey on multi-task learning, IEEE Trans. Knowl. Data Eng. 4347 (2021) 1–20, https://doi.org/10.1109/TKDE.2021.3070203.

[67] S. Ruder, An overview of multi-task learning in deep neural networks, ArXiv Prepr. ArXiv1706.05098. (2017). ⟨http://arxiv.org/abs/1706.05098⟩http://arxiv.org/abs/1706.05098.

[68] V. Dankers, M. Rei, M. Lewis, E. Shutova, Modelling the interplay of metaphor and emotion through multitask learning (in:), Adv. Intell. Data Anal. Xv. IDA 2016 (2019) 2218–2229, https://doi.org/10.18653/v1/d19-1227.

[69] R.K. Ando, T. Zhang, P. Bartlett, A framework for learning predictive structures from multiple tasks and unlabeled data, J. Mach. Learn. Res 6 (2005) 1817–1853. https://dl.acm.org/doi/10.5555/1046920.1194905.

[70] Y. Zhou, R. Jin, S.C.H. Hoi, Exclusive lasso for multi-task feature selection, J. Mach. Learn. Res 9 (2010) 988–995, in: http://proceedings.mlr.press/v9/zhou10a/zhou10a.pdf.

[71] A. Argyriou, T. Evgeniou, M. Pontil, Convex multi-task feature learning, Mach. Learn. 73 (2008) 243–272, https://doi.org/10.1007/s10994-007-5040-8.

[72] H. Liu, M. Palatucci, J. Zhang, Blockwise coordinate descent procedures for the multi-task Lasso, with applications to neural semantic basis discovery, Conf. Mach. Learn. ICML (2009) 649–656. https://doi.org/10.1145/1553374.1553458.

[73] P. Gong, J. Ye, C. Zhang, Multi-stage multi-task feature learning, J. Mach. Learn. Res. 14 (2013) 2979–3010. https://dl.acm.org/doi/10.5555/2567709.2567756.

[74] A. Argyriou, C.A. Micchelli, M. Pontil, Y. Ying, A spectral regularization framework for multi-task structure learning, in: Adv. Neural Inf. Process. Syst. 20 - Proc. 2007 Conf., 2007, pp. 1–8. https://dl.acm.org/doi/10.5555/2981562.2981566.

[75] R. Caruana, Multitask learning, Mach. Learn. 28 (1997) 41–75, https://doi.org/10.1023/A:1007379606734.

[76] K. Weiss, T.M. Khoshgoftaar, D.D. Wang, A survey of transfer learning, J. Big Data 3 (2016) 1–40, https://doi.org/10.1186/s40537-016-0043-6.

[77] Z. Wang, Z. Dai, B. Póczos, J. Carbonell, Characterizing and avoiding negative transfer, in: IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 11285-11294. doi: 10.1109/CVPR.2019.01155.

[78] W. Zhang, L. Deng, L. Zhang, D. Wu, A survey on negative transfer, ArXiv Prepr. ArXiv2009.00909. (2020). http://arxiv.org/abs/2009.00909.

[79] D. Soekhoe, P. van der Putten, A. Plaat, On the impact of data set size in transfer learning using deep neural networks, Adv. Intell. Data Anal. XV. IDA (2016), https://doi.org/10.1007/978-3-319-46349-0_5.

[80] J.G.A. Barbedo, Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification, Comput. Electron. Agric. 153 (2018) 46–53, https://doi.org/10.1016/j.compag.2018.08.013.

[81] X. Huang, T. Xie, Z. Wang, L. Chen, Q. Zhou, Z. Hu, A transfer learning-based multi-fidelity point-cloud neural network approach for melt pool modeling in additive manufacturing, ASCE ASME J. Risk Uncertain. Eng. Syst. Part B Mech. Eng. 8 (2022) 1–11, https://doi.org/10.1115/1.4051749.

[82] M. Xia, H. Shao, Z. Huang, Z. Zhao, F. Jiang, Y. Hu, Intelligent process monitoring of laser-induced graphene production with deep transfer learning, IEEE Trans. Instrum. Meas. 71 (2022) 1–9, https://doi.org/10.1109/TIM.2022.3186688.

[83] S. Liao, T. Xue, J. Jeong, S. Webster, K. Ehmann, J. Cao, Hybrid full-field thermal characterization of additive manufacturing processes using physics-informed neural networks with data, ArXiv Prepr. (2022). http://arxiv.org/abs/2206.07756.

[84] X. Li, Y. Zhang, H. Zhao, C. Burkhart, L.C. Brinson, W. Chen, A transfer learning approach for microstructure reconstruction and structure-property predictions, Sci. Rep. 8 (2018) 1–13, https://doi.org/10.1038/s41598-018-31571-7.

[85] Z. Yang, X. Li, L.C. Brinson, A.N. Choudhary, W. Chen, A. Agrawal, Microstructural materials design via deep adversarial learning methodology, J. Mech. Des. Trans. ASME 140 (2018) 1–10, https://doi.org/10.1115/1.4041371.

[86] G. Chen, Y. Li, X. Liu, Pose-dependent tool tip dynamics prediction using transfer learning, Int. J. Mach. Tools Manuf. 137 (2019) 30–41, https://doi.org/10.1016/j.ijmachtools.2018.10.003.

[87] L. Cheng, F. Tsung, A. Wang, A statistical transfer learning perspective for modeling shape deviations in additive manufacturing, IEEE Robot. Autom. Lett. 2 (2017) 1988–1993, https://doi.org/10.1109/LRA.2017.2713238.

[88] A. Sabbaghi, Q. Huang, Model transfer across additive manufacturing processes via mean effect equivalence of lurking variables, Ann. Appl. Stat. 12 (2018) 2409–2429, https://doi.org/10.1214/18-AOAS1158.

[89] J. Deng, W. Dong, R. Socher, L.J. Li, K. Li, F.F. Li. ImageNet: a large-scale hierarchical image database, IEEE, 2009, pp. 248–255, https://doi.org/10.1109/CVPR.2009.5206848.

[90] M. Mehta, C. Shao, Federated learning-based semantic segmentation for pixel-wise defect detection in additive manufacturing, J. Manuf. Syst. 64 (2022) 197–210, https://doi.org/10.1016/j.jmsy.2022.06.010.

[91] D. Knüttel, S. Baraldo, A. Valente, K. Wegener, E. Carpanzano, Transfer learning of neural network based process models in Direct Metal Deposition, Procedia CIRP 107 (2022) 863–868, https://doi.org/10.1016/j.procir.2022.05.076.

[92] J. Ren, A.T. Wei, Z. Jiang, H. Wang, X. Wang, Improved modeling of kinematics-induced geometric variations in extrusion-based additive manufacturing through between-printer transfer learning, IEEE Trans. Autom. Sci. Eng. (2021) 1–12, https://doi.org/10.1109/TASE.2021.3063389.

[93] A. Sabbaghi, Q. Huang, T. Dasgupta, Bayesian model building from small samples of disparate data for capturing in-plane deviation in additive manufacturing, Technometrics 60 (2018) 532–544, https://doi.org/10.1080/00401706.2017.1391715.

[94] R.D.S.B. Ferreira, A. Sabbaghi, Q. Huang, Automated geometric shape deviation modeling for additive manufacturing systems via Bayesian neural networks, IEEE Trans. Autom. Sci. Eng. 17 (2020) 584–598, https://doi.org/10.1109/TASE.2019.2936821.

[95] J. Ren, H. Wang, Surface variation modeling by fusing multiresolution spatially nonstationary data under a transfer learning framework, J. Manuf. Sci. Eng. Trans. ASME 141 (2019), 011002, https://doi.org/10.1115/1.4041425.

[96] V. Pandiyan, R. Drissi-Daoudi, S. Shevchik, G. Masinelli, T. Le-Quang, R. Logé, K. Wasmer, Deep transfer learning of additive manufacturing mechanisms across materials in metal-based laser powder bed fusion process, J. Mater. Process. Technol. 303 (2022), https://doi.org/10.1016/j.jmatprotec.2022.117531.

[97] S.-J. Shin, J.-H. Lee, J. Sainand, D.B. Kim, Material-adaptive anomaly detection using property-concatenated transfer learning in wire arc additive manufacturing. https://doi.org/10.2139/ssrn.4242808.

[98] Z. Zhang, Q. Liu, D. Wu, Predicting stress–strain curves using transfer learning: knowledge transfer across polymer composites, Mater. Des. 218 (2022), 110700, https://doi.org/10.1016/j.matdes.2022.110700.

[99] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Adv. Neural Inf. Process. Syst. (2012) 1097–1105. https://doi.org/10.1145/3065386.

[100] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: IEEE Conf. Comput. Vis. Pattern Recognit., Las Vegas, NV, USA, June 27–30, 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[101] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: 3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc., San Diego, California, USA, May 7–9, 2015, pp. 1–14. https://doi.org/10.48550/arXiv.1409.1556.

[102] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision, in: IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (2016) pp: 2818–2826, https://doi.org/10.1109/CVPR.2016.308.

[103] M. Tan, Q. Le, EfficientNet: rethinking model scaling for convolutional neural networks, in: Proc. 36th Int. Conf. Mach. Learn. PMLR, 2019, pp. 6105–6114. https://doi.org/10.48550/arXiv.1905.11946.

[104] Y.M. Ren, Y. Zhang, Y. Ding, Y. Wang, P.D. Christofides, Computational fluid dynamics-based in-situ sensor analytics of direct metal laser solidification process using machine learning, Comput. Chem. Eng. 143 (2020), 107069, https://doi.org/10.1016/j.compchemeng.2020.107069.

[105] A.R. Kitahara, E.A. Holm, Microstructure cluster analysis with transfer learning and unsupervised learning, Integr. Mater. Manuf. Innov. 7 (2018) 148–156, https://doi.org/10.1007/s40192-018-0116-9.

[106] M.N. Bisheh, X. Wang, S.I. Chang, S. Lei, J. Ma, Image-based characterization of laser scribing quality using transfer learning, J. Intell. Manuf. (2022), https://doi.org/10.1007/s10845-022-01926-z.

[107] S. Hu, Z. Li, S. Zhang, Quality spectra fluctuation modeling for manufacturing process based on deep transfer learning, J. Phys. Conf. Ser. 1983 (2021) 01201, https://doi.org/10.1088/1742-6596/1983/1/012101.

[108] Z. Jin, Z. Zhang, X. Shao, G.X. Gu, Monitoring anomalies in 3D bioprinting with deep neural networks, ACS Biomater. Sci. Eng. (2021), https://doi.org/10.1021/acsbiomaterials.0c01761.

[109] F.G. Fischer, M.G. Zimmermann, N. Praetzsch, C. Knaak, Monitoring of the powder bed quality in metal additive manufacturing using deep transfer learning, Mater. Des. 222 (2022), 111029, https://doi.org/10.1016/j.matdes.2022.111029.

[110] D.A.J. Brion, M. Shen, S.W. Pattinson, Automated recognition and correction of warp deformation in extrusion additive manufacturing, Addit. Manuf. 56 (2022), 102838, https://doi.org/10.1016/j.addma.2022.102838.

[111] K.J. Kim, P.K. Kim, Y.S. Chung, D.H. Choi, Performance enhancement of YOLOv3 by adding prediction layers with spatial pyramid pooling for vehicle detection, in: Proc. AVSS 2018 - 2018 15th IEEE Int. Conf. Adv. Video Signal-Based Surveill., IEEE, 2018, pp. 14–19. https://doi.org/10.1109/AVSS.2018.8639438.

[112] T.Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, Microsoft COCO: Common objects in context, in: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (Eds.), Lecture Notes in Computer Science, 8693, Springer, Cham, 2014, https://doi.org/10.1007/978-3-319-10602-1_48.

[113] C. Xia, Z. Pan, Y. Li, J. Chen, H. Li, Vision-based melt pool monitoring for wire-arc additive manufacturing using deep learning method, Int. J. Adv. Manuf. Technol. 120 (2022) 551–562, https://doi.org/10.1007/s00170-022-08811-2.
[114] H. Kim, H. Lee, S.H. Ahn, Systematic deep transfer learning method based on a small image dataset for spaghetti-shape defect monitoring of fused deposition modeling, J. Manuf. Syst. 65 (2022) 439–451, https://doi.org/10.1016/j.jmsy.2022.10.009.
[115] L. Wang, X. Chen, D. Henkel, R. Jin, Family learning: a process modeling method for cyber-additive manufacturing network, IISE Trans. 54 (2021) 1–16, https://doi.org/10.1080/24725854.2020.1851824.
[116] R. Tibshirani, Regression shrinkage and selection via the Lasso, J. R. Stat. Soc. Ser. B 58 (1996) 267–288, https://doi.org/10.1111/j.2517-6161.1996.tb02080.x.
[117] Y. Zhang, L. Wang, X. Chen, R. Jin, Fog computing for distributed family learning in cyber-manufacturing modeling, in: 2019 IEEE Int. Conf. Ind. Cyber Phys. Syst. ICPS 2019, IEEE, Taipei, Taiwan, China, 6–9 May, 2019, pp. 88–93, doi: 10.1109/ICPHYS.2019.8780264.
[118] Z. Zhu, N. Anwer, Q. Huang, L. Mathieu, Machine learning in tolerancing for additive manufacturing, CIRP Ann. 67 (2018) 157–160, https://doi.org/10.1016/j.cirp.2018.04.119.
[119] W. Sun, Z. Zhang, W. Ren, J. Mazumder, J.J. Jin, In situ monitoring of optical emission spectra for microscopic pores in metal additive manufacturing, J. Manuf. Sci. Eng. Trans. ASME 144 (2022) 1–13, https://doi.org/10.1115/1.4051532.
[120] A.T. Wei, H. Wang, T. Dickens, H. Chi, Co-learning of extrusion deposition quality for supporting interconnected additive manufacturing systems, IISE Trans. (2022), https://doi.org/10.1080/24725854.2022.2080306.
[121] J. Huang, Y.F. Li, M. Xie, An empirical analysis of data preprocessing for machine learning-based software cost estimation, Inf. Softw. Technol. 67 (2015) 108–127, https://doi.org/10.1016/j.infsof.2015.07.004.
[122] Q. Wu, H. Wu, X. Zhou, M. Tan, Y. Xu, Y. Yan, T. Hao, Online transfer learning with multiple homogeneous or heterogeneous sources, IEEE Trans. Knowl. Data Eng. 29 (2017) 1494–1507, https://doi.org/10.1109/TKDE.2017.2685597.
[123] Z. Kang, B. Yang, S. Yang, X. Fang, C. Zhao, Online transfer learning with multiple source domains for multi-class classification, Knowl. Based Syst. 190 (2020), 105149, https://doi.org/10.1016/j.knosys.2019.105149.
[124] D. Wu, G.G. Wang, Knowledge-assisted optimization for large-scale design problems: a review and proposition, J. Mech. Des. 142 (2020), 010801, https://doi.org/10.1115/1.4044525.
[125] B. Kapusuzoglu, S. Mahadevan, Physics-informed and hybrid machine learning in additive manufacturing: application to fused filament fabrication, JOM 72 (2020) 4695–4705, https://doi.org/10.1007/s11837-020-04438-4.
[126] G.E. Karniadakis, I.G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, L. Yang, Physics-informed machine learning, Nat. Rev. Phys. 3 (2021) 422–440, https://doi.org/10.1038/s42254-021-00314-5.
[127] R. Chen, Y. Lu, P. Witherell, T.W. Simpson, S. Kumara, H. Yang, Ontology-driven learning of Bayesian network for causal inference and quality assurance in additive manufacturing, IEEE Robot. Autom. Lett. 6 (2021) 6032–6038, https://doi.org/10.1109/LRA.2021.3090020.
[128] B.J. Simonds, J. Tanner, A. Artusio-Glimpse, P.A. Williams, N. Parab, C. Zhao, T. Sun, The causal relationship between melt pool geometry and energy absorption measured in real time during laser-based manufacturing, Appl. Mater. Today 23 (2021), 101049, https://doi.org/10.1016/j.apmt.2021.101049.