

Technical paper

Personalized feature extraction for manufacturing process signature characterization and anomaly detection

Naichen Shi ^a, Shenghan Guo ^b, Raed Al Kontar ^{a,*}

^a Industrial & Operational Engineering, University of Michigan, Ann Arbor, 48105, MI, USA

^b The School of Manufacturing Systems & Networks, Arizona State University, Mesa, 85212, AZ, USA

ARTICLE INFO

Keywords:

Process signatures

Anomaly detection

Personalized features

ABSTRACT

Manufacturing process signatures reflect the process stability and anomalies that potentially lead to detrimental effects on the manufactured outcomes. Sensing technologies, especially in-situ image sensors, are widely used to capture process signatures for diagnostics and prognostics. This imaging data is crucial evidence for process signature characterization and monitoring. A critical aspect of process signature analysis is identifying the unique patterns in an image that differ from the generic behavior of the manufacturing process in order to detect anomalies. It is equivalent to separating the “unique features” and process-wise (or phase-wise) “shared features” from the same image and recognizing the transient anomaly, i.e., recognizing the outlier “unique features”. In state-of-the-art literature, image-based process signature analysis relies on conventional feature extraction procedures, which limit the “view” of information to each image and cannot decouple the shared and unique features. Consequently, the features extracted are less interpretable, and the anomaly detection method cannot distinguish the abnormality in the current process signature from the process-wise evolution. Targeting this limitation, this study proposes personalized feature extraction (PFE) to decouple process-wise shared features and transient unique features from a sensor image and further develops process signature characterization and anomaly detection strategies. The PFE algorithm is designed for heterogeneous data with shared features. Supervised and unsupervised anomaly detection strategies are developed upon PFE features to remove the shared features from a process signature and examine the unique features for abnormality. The proposed method is demonstrated on two datasets (i) selected data from the 2018 AM Benchmark Test Series from the National Institute of Standards and Technology (NIST), and (ii) thermal measurements in additive manufacturing of a thin-walled structure of Ti-6Al-4V. The results highlight the power of personalized modeling in extracting features from manufacturing imaging data.

1. Introduction

In manufacturing, *process signatures* refer to the information collected for the physical conditions and outcomes of a manufacturing process or system [1–5]. Typical process signatures for machining processes include heat dissipation, surface integrity, material modification, and material load [1], while process signatures for additive manufacturing (AM) processes mainly consist of melt pool dynamics [5–8], spattering patterns [9,10], fusion area topology [11], and residual stress [12]. Process signatures enable correlation analysis between process parameters and manufacturing outcomes [1]. For example, the analysis of material load signatures builds a correlation between the process parameters and surface integrity in machining [1]; the inspection of melt pool signatures establishes a parameter–signature–quality relationship for metal AM [5]. Indeed, process signatures are valuable

to deepen the understanding of a manufacturing process, optimize the parameters, and facilitate quality improvement.

Mostly, process signatures reflect the dynamic processing results that must be controlled indirectly by modifying the process parameters [13]. Hence, sensing technologies are necessary to capture the process signatures, e.g., thermal imaging [2] and acoustic signaling [14]. In-situ thermal imaging [10,15–17] is especially popular for capturing heat-related process signatures with high resolution and frequency during the manufacturing process. For instance, Fig. 1 shows the in-situ thermal images for spattering patterns during a laser-based metal AM process under regular and irregular situations. These sensing data are crucial evidence for characterizing the process signatures and, more importantly, detecting the abnormality in process signatures caused by process instability, mechanical issues, and systematic failures.

* Corresponding author.

E-mail address: alkontar@umich.edu (R. Al Kontar).

<https://doi.org/10.1016/j.jmansys.2024.04.002>

Received 4 September 2023; Received in revised form 20 February 2024; Accepted 2 April 2024

Available online 15 April 2024

0278-6125/© 2024 The Society of Manufacturing Engineers. Published by Elsevier Ltd. All rights reserved.

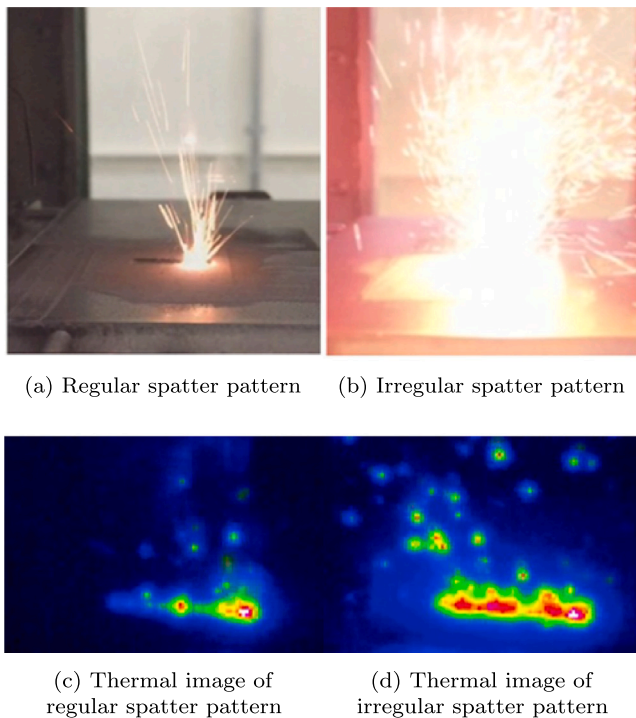


Fig. 1. Examples of regular and irregular (or severe) side view spattering pattern and thermal images.

Source: Adapted from Amano et al. [18].

Nonetheless, process signature characterization and anomaly detection from thermal images are challenging. Process signatures are highly dynamic. The thermal images capture both generic process signatures and transient patterns. Taking Fig. 1 as an example, the plume region with high temperature shows the process signature's generic shape, size, and temperature distribution; the colder region and scattered dots are transient spattering patterns at the printing time. The generic information reveals process-wise features [19,20] and is more relevant to process signature characterization. In contrast, the transient patterns are evidence of instability and unexpected changes and, thus, are more useful for anomaly detection. To characterize the process signature and facilitate anomaly detection, a powerful feature extraction approach is needed to decouple the generic and transient information.

Currently, feature decoupling remains understudied in manufacturing process signature analysis. State-of-the-art literature [9,10,17,21,22] mostly extracted features from the entire thermal images using classic statistical approaches, e.g., Principal Component Analysis (PCA) [16] and tensor decomposition [23], but do not decompose the features into process-wise features (shared by subsequent thermal images) and transient features (unique to a single thermal image). Targeting the research gap, this study explores *personalized feature extraction* (PFE) [24,25] for process signature characterization and anomaly detection in thermal images. The proposed PFE algorithm is developed upon asymmetric matrix factorization [26] and exploits highly efficient gradient descent. Compared with prior works such as PerPCA [24] or JIVE [27], this algorithm significantly improves computational efficiency by removing the need for feature orthonormalization or SVD on large matrices. The goal is to leverage this PFE algorithm for identifying the shared features that are generic for the manufacturing process (or each phase during the process) and, meanwhile, extracting the unique features owned by individual process signatures. Once decoupled, the shared features will unveil high-level information about the underlying physics and are used for process signature characterization; the unique features will become the inputs for anomaly detection strategies that capture irregular, transient patterns

in process signatures. This study proposes two alternative strategies for supervised and unsupervised anomaly detection. They will support decision-making with thermal images either labeled with ground-truth anomaly records or unlabeled.

This work will contribute to manufacturing process signature analysis and, broadly, data-enabled smart manufacturing [28]. First, it identifies a critical research gap in process signature characterization, revealing the need for feature decoupling to separate the generic and transient information in manufacturing thermal images. Second, PFE is introduced into manufacturing data analysis to build a PFE paradigm. Its feature decoupling and subsequent anomaly detection performance will set a benchmark for further exploration, either from a methodology or application perspective. Third, anomaly detection based on PFE features respects the physics of manufacturing processes and generates more interpretable results. By altering the thermal image batches fed to PFE, one can characterize the generic process signatures for specific phases in the manufacturing process and better understand the source of abnormality, i.e., process-wise changes or degradation [29] caused by systemic issues (e.g., machine failures, incorrect parameters) or temporary instability caused by uncertainty [30]. In general, the methodology and case studies from this work will generate new knowledge for manufacturing data analysis and indicate future directions to improve data-enabled smart manufacturing.

The rest of this paper is organized as follows. Section 2 will review state-of-the-art literature for process signature modeling and PFE. Section 3 will elaborate on the technical details of PFE and anomaly detection strategies. Section 4 will apply the proposed methods on two datasets from metal AM [31] and discuss the results. Section 5 will conclude the paper and remark on future research directions.

2. Related work

This section reviews state-of-the-art literature on image-based manufacturing process signature analysis and common statistical methods for low-rank modeling of high-dimensional data. The potential and suitability of PFE for process signature characterization and anomaly detection are highlighted.

2.1. Image-based manufacturing process signature analysis

Depending on the manufacturing processes and data forms, process signature analysis may be physics-based modeling [12,32], correlation evaluation [1,33], or purely data-driven analytics [13,34,35]. Data-driven approaches have seen a surging popularity due to the advances in sensing technologies [5,36,37]. Imaging data is especially widely adopted [5,8,11,38], motivating various studies about image-based manufacturing process signature modeling and anomaly detection. Generic image-based process signature analysis follows four steps:

1. **Data processing:** This step removes the background in images and preserves the image segments containing the process signature, i.e., the so-called “region of interest” (ROI) [9]. Object recognition techniques may be used for locating the ROI in image [10,21,39]. Down-sampling may be done on the extracted ROI to reduce the data size and facilitate model fitting [40].
2. **Feature extraction:** This step defines features describing the shape, size, or temperature distribution of the process signatures and extracts these features from the processed thermal images [9,10,17,21,22]. Some works used statistical dimensionality reduction [16,23] or deep learning (DL) models [41–43] for automatic feature extraction from thermal images, thus saving the effort of manual feature definition and extraction.
3. **Model fitting/training:** This step builds a monitoring statistic (or equivalently, a control chart) [9,10,23,40,44] or trains a machine learning (ML) model [16,17,22,42,45] with the features of ROI. If supervised ML models are selected, the ground-truth anomaly records are also needed as the labels of ROI features [41–43].

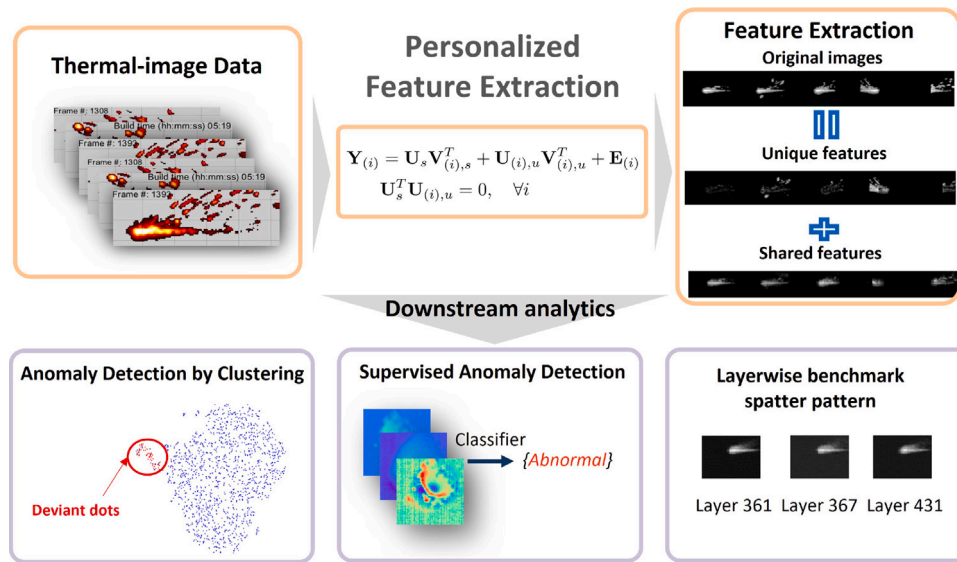


Fig. 2. An overview of our method.

4. **Anomaly detection:** This step implements the fitted or trained model for in-situ anomaly detection [9,10,16,17,22,41,42,44,46], phase-wise (e.g., a printed layer in AM) anomaly detection [19], or prognostic analysis [47–49].

Step 1 is dependent on the data quality. The most critical step is step 2. How well the features describe the process signatures directly impacts the fitness and learning outcome of the anomaly detection model. For manual feature extraction, the features' descriptive power relies on the human understanding of the data and manufacturing processes, which can be uncomprehensive, especially when the operator is not a domain expert. Automatic feature extraction with statistical dimensionality reduction or DL models is a promising direction. It is advantageous in efficiency and information preservation. However, these methods do not decouple the process-wise shared features and unique image-wise features. Consequently, anomaly detection must be done on a mixture of shared and unique features, which is less physically valid or interpretable.

Recently, the coexistence of shared and unique features in manufacturing process signatures has drawn researchers' attention. Preliminary work was done in Yang et al. [23], where the monitoring statistic (i.e., the model in step 3) was built for between-layer anomaly, within-layer anomaly, and between-sample anomaly for AM processes. This is one step forward toward PFE and anomaly detection based on unique features. Nonetheless, the work emphasized a hierarchical design of monitoring statistics rather than the decoupling of phase-wise shared features and unique features. PFE for process signature characterization and anomaly detection with manufacturing images remains an open issue.

2.2. PFE and low-rank modeling

Principal Component Analysis (PCA) is an important unsupervised learning method to extract important features from high-dimensional data. On imaging data, PCA is widely used for compression [50], denoising [51], segmentation [52], and many more. Essentially, PCA finds a low-dimensional subspace that can explain the most variance in high-dimensional data. Therefore, PCA is also closely related to methods devoted to finding low-rank structures (i.e., low-dimensional representations of the original data) from the observation matrix. Numerous works also applied low-rank matrix factorization methods to image processing [53,54].

One limitation of using classic PCA or other low-rank modeling methods in image processing is that they often make holistic assumptions about the images and, thus, are inadequate to differentiate common (or shared) and personalized (or unique) information from different images. A series of algorithms, including joint and individual variance explained (JIVE) [27], common orthogonal basis extraction (COBE) [55], SLIDE [56], and PerPCA [24], propose to extract shared and unique linear features from data. Among them, JIVE introduces an alternating minimization algorithm to find the common and individual low-rank components in data. COBE analyzes the singular vectors of the observation matrices and attempts to identify the common parts of the singular vectors. A caveat of these algorithms is that they are based on heuristics and hence cannot justify the correctness of their outputs. PerPCA [24] is proposed to provably separate the shared and unique features with a Stiefel gradient descent algorithm. PerPCA has strong statistical and convergence guarantees and has been shown to be useful in multiple tasks. However, it only applies to symmetric covariance matrices and involves computationally expensive retraction steps. Different from PerPCA, this work proposes a formulation based on asymmetric matrix factorization to extract shared and unique features while inheriting the strong theoretical guarantees of PerPCA. We use a few numerical studies to illustrate the difference between PFE and existing algorithms in terms of statistical performance and computational efficiency. The feature-extraction capabilities of our approach make it a natural fit to analyze the variations of manufacturing process signatures. We also show that the extracted features provide invaluable information for downstream data analytics.

3. Method development

This section elaborates on the technical details of our personalized feature extraction methodology and its use within downstream analytics, namely process characterization and anomaly detection.

3.1. Method overview

Before delving into the methodology, we hereby highlight our overarching framework in Fig. 2. As shown in Fig. 2, our approach extracts shared and unique features from thermal images. These features are then used for:

- *Unsupervised outlier detection*: Through finding outliers from unique features, we can detect abnormal process signatures. The central philosophy is to emphasize the information in unique features so that differences are more explicit for improved outlier detection.
- *Supervised anomaly detection*: When labeled data are accessible, we can also train classifiers based on the extracted features to predict anomalies in real-time.
- *Building in-control benchmarks*: The extracted shared features from thermal images of a given layer can be used as a benchmark that characterizes the expected behavior of spatter patterns within that layer.

3.2. Personalized feature extraction via matrix factorization

The method development considers thermal or gray-scale images capturing manufacturing process signatures. Assume that, after *pre-processing*, data from each image is represented by $\mathbf{Y}_{(i)} \in \mathbb{R}^{d \times n}$, and the data is collected across N thermal images. For each phase during manufacturing, e.g., a printed layer or one build in AM, the process is relatively stable, so $\{\mathbf{Y}_{(i)}\}_{i \in [N]}$ of the N images share certain common characteristics. Meanwhile, each image contains unique characteristics for the transient spattering behavior at the printing time.

As an effort to model such commonality and heterogeneity, we take a matrix factorization route where we decompose $\mathbf{Y}_{(i)}$ into shared and unique constituents. Specifically, we assume that $\mathbf{Y}_{(i)}$ can be explained by r_1 shared features and r_2 unique features. The notation $\mathbf{U}_s$ represents the shared feature matrix, and $\mathbf{U}_{(i),u}$ represents the unique feature matrix. Then $\mathbf{Y}_{(i)} \in \mathbb{R}^{d \times n}$ is modeled as:

$$\mathbf{Y}_{(i)} = \mathbf{U}_s \mathbf{V}_{(i),s}^T + \mathbf{U}_{(i),u} \mathbf{V}_{(i),u}^T + \mathbf{E}_{(i)} \quad (1)$$

where $\mathbf{U}_s \in \mathbb{R}^{d \times r_1}$, $\mathbf{U}_{(i),u} \in \mathbb{R}^{d \times r_2}$ are feature matrices, $\mathbf{V}_{(i),s} \in \mathbb{R}^{n \times r_1}$, $\mathbf{V}_{(i),u} \in \mathbb{R}^{n \times r_2}$, are the corresponding feature coefficient matrices, and $\mathbf{E}_{(i)} \in \mathbb{R}^{d \times n}$ are noise matrices. The dimension parameters d and n represent the shape of data. The features $\mathbf{U}_s$ are shared for all thermal images, while each image retains its own unique factor matrix $\mathbf{U}_{(i),u}$. However, we allow the shared coefficient matrices $\mathbf{V}_{(i),s}$ to vary across images, as the intensity of shared components may not be the same in all the images. Despite this decomposed formulation, (1) is essentially a linear model. We will discuss non-linear extensions in Section 4.

A key assumption to ensure the separability between shared and unique features is the orthogonality condition below:

$$\mathbf{U}_s^T \mathbf{U}_{(i),u} = 0, \quad \forall i = 1, \dots, N \quad (2)$$

Condition (2) essentially requires that the shared and unique components span orthogonal subspaces to characterize different features. Such inductive bias is intuitive as shared and unique features should be dissimilar by definition. Orthogonality enforces this definition and allows us to maximize the effectiveness of feature decoupling.

Now, to recover the features, we define our objective to minimize the reconstruction error between our model and observed data, under the orthogonality constraint. This is given as:

$$\min_{\{\mathbf{U}_s, \{\mathbf{V}_{(i),s}\}, \{\mathbf{U}_{(i),u}\}, \{\mathbf{V}_{(i),u}\}\}} \sum_{i=1}^N \left[\left\| \mathbf{Y}_{(i)} - \mathbf{U}_s \mathbf{V}_{(i),s}^T - \mathbf{U}_{(i),u} \mathbf{V}_{(i),u}^T \right\|_F^2 \right] \quad (3)$$

such that $\mathbf{U}_s^T \mathbf{U}_{(i),u} = 0, \quad \forall i$

As reiterated in (3), the shared features $\mathbf{U}_s$ are the same among all the images, while the unique features $\mathbf{U}_{(i),u}$ can differ for different images.

To optimize (3), a natural route to take is gradient descent that respects the orthogonality constraint. To do so, for thermal image i , we define $f_i(\mathbf{U}_{s,\tau}, \mathbf{U}_{(i),u,\tau}, \mathbf{V}_{(i),s,\tau}, \mathbf{V}_{(i),u,\tau}) = \left\| \mathbf{Y}_{(i)} - \mathbf{U}_{s,\tau} \mathbf{V}_{(i),s,\tau}^T - \mathbf{U}_{(i),u,\tau} \mathbf{V}_{(i),u,\tau}^T \right\|_F^2$,

where we use $\mathbf{U}_{s,\tau}$, $\mathbf{U}_{(i),u,\tau}$, $\mathbf{V}_{(i),s,\tau}$, $\mathbf{V}_{(i),u,\tau}$ to denote the update at epoch τ . Then the gradients of the f_i are:

$$\begin{cases} \nabla_{\mathbf{U}_s} f_i = 2 \left(\mathbf{U}_{s,\tau} \mathbf{V}_{(i),s,\tau}^T + \mathbf{U}_{(i),u,\tau} \mathbf{V}_{(i),u,\tau}^T - \mathbf{Y}_{(i)} \right) \mathbf{V}_{(i),s,\tau} \\ \nabla_{\mathbf{V}_{(i),s}} f_i = 2 \left(\mathbf{U}_{s,\tau} \mathbf{V}_{(i),s,\tau}^T + \mathbf{U}_{(i),u,\tau} \mathbf{V}_{(i),u,\tau}^T - \mathbf{Y}_{(i)} \right)^T \mathbf{U}_{s,\tau} \\ \nabla_{\mathbf{U}_{(i),u}} f_i = 2 \left(\mathbf{U}_{s,\tau} \mathbf{V}_{(i),s,\tau}^T + \mathbf{U}_{(i),u,\tau} \mathbf{V}_{(i),u,\tau}^T - \mathbf{Y}_{(i)} \right) \mathbf{V}_{(i),u,\tau} \\ \nabla_{\mathbf{V}_{(i),u}} f_i = 2 \left(\mathbf{U}_{s,\tau} \mathbf{V}_{(i),s,\tau}^T + \mathbf{U}_{(i),u,\tau} \mathbf{V}_{(i),u,\tau}^T - \mathbf{Y}_{(i)} \right)^T \mathbf{U}_{(i),u,\tau} \end{cases} \quad (4)$$

To handle the orthogonal constraints, we can use a correction step to explicitly orthogonalize $\mathbf{U}_s$ and $\mathbf{U}_{(i),u}$. More specifically, we apply the projection $\mathbf{I} - \mathbf{U}_{s,\tau} \left(\mathbf{U}_{s,\tau}^T \mathbf{U}_{s,\tau} \right)^{-1} \mathbf{U}_{s,\tau}^T$ to project unique features to the orthogonal space of shared features $\mathbf{U}_s$. The resulting $\mathbf{U}_{(i),u,\tau} = \left(\mathbf{I} - \mathbf{U}_{s,\tau} \left(\mathbf{U}_{s,\tau}^T \mathbf{U}_{s,\tau} \right)^{-1} \mathbf{U}_{s,\tau}^T \right) \mathbf{U}_{(i),u,\tau - \frac{1}{2}}$ is orthogonal to $\mathbf{U}_{s,\tau}$, where we use $\mathbf{U}_{(i),u,\tau - \frac{1}{2}}$, $\mathbf{V}_{(i),s,\tau - \frac{1}{2}}$ to denote intermediate variables between two full rounds of updates. Thus, the constraint is satisfied. However, such a correction step can change the function value f_i and be counterproductive to the optimization. To maintain the same value of f_i , we also introduce a correction step on $\mathbf{V}_{(i),s,\tau}$ that $\mathbf{V}_{(i),s,\tau} = \mathbf{V}_{(i),s,\tau - \frac{1}{2}} + \mathbf{V}_{(i),s,\tau} \mathbf{U}_{(i),u,\tau}^T \mathbf{U}_{s,\tau} \left(\mathbf{U}_{s,\tau}^T \mathbf{U}_{s,\tau} \right)^{-1}$. One can easily check that f_i remains the same before and after the correction step, i.e., $f_i(\mathbf{U}_{s,\tau}, \mathbf{V}_{(i),s,\tau}, \mathbf{U}_{(i),u,\tau}, \mathbf{V}_{(i),u,\tau}) = f_i(\mathbf{U}_{s,\tau}, \mathbf{V}_{(i),s,\tau - \frac{1}{2}}, \mathbf{U}_{(i),u,\tau - \frac{1}{2}}, \mathbf{V}_{(i),u,\tau})$. The computation cost of $\mathbf{V}_{(i),s,\tau}$ correction is comparable to that of $\mathbf{U}_{(i),u,\tau}$ correction. Hence, the additional computation cost associated with $\mathbf{V}_{(i),s,\tau}$ correction is small.

The correction step, along with the gradients in (4), induces a natural iterative algorithm to solve (3). At the beginning of epoch τ , we deflate the unique feature matrix $\mathbf{U}_{(i),u}$ to make it orthogonal to the shared feature matrix $\mathbf{U}_s$, and correct $\mathbf{V}_{(i),s}$ accordingly. This step guarantees that the constraints in (3) are satisfied at the beginning of each epoch. Then, we perform gradient descent using the gradients in (4) to minimize the fitting error. Afterward, we take the average of all updated $\mathbf{U}_s$ at the end of each epoch to balance the updates from all images. The pseudo-code of the projected gradient descent algorithm is shown in Algorithm 1.

Algorithm 1 Gradient descent to optimize (3).

Input matrices $\{\mathbf{Y}_{(i)}\}_{i=1}^N$, stepsize η_τ
Initialize $\mathbf{U}_{s,1}$, $\mathbf{V}_{(i),s,\frac{1}{2}}$, $\mathbf{U}_{(i),u,\frac{1}{2}}$, $\mathbf{V}_{(i),u,1}$ to be small random matrices.
for Epoch $\tau = 1, \dots, R$ **do**
 for Time $i = 1, \dots, N$ **do**
 Correct $\mathbf{V}_{(i),s,\tau} = \mathbf{V}_{(i),s,\tau - \frac{1}{2}} + \mathbf{V}_{(i),s,\tau} \mathbf{U}_{(i),u,\tau}^T \mathbf{U}_{s,\tau} \left(\mathbf{U}_{s,\tau}^T \mathbf{U}_{s,\tau} \right)^{-1}$
 Correct $\mathbf{U}_{(i),u,\tau} = \mathbf{U}_{(i),u,\tau - \frac{1}{2}} - \mathbf{U}_{s,\tau} \left(\mathbf{U}_{s,\tau}^T \mathbf{U}_{s,\tau} \right)^{-1} \mathbf{U}_{s,\tau}^T \mathbf{U}_{(i),u,\tau - \frac{1}{2}}$
 Update $\mathbf{U}_{(i),s,\tau+1} = \mathbf{U}_{s,\tau} - \eta_\tau \nabla_{\mathbf{U}_s} f_i$
 Update $\mathbf{V}_{(i),s,\tau + \frac{1}{2}} = \mathbf{V}_{(i),s,\tau} - \eta_\tau \nabla_{\mathbf{V}_{(i),s}} f_i$
 Update $\mathbf{U}_{(i),u,\tau + \frac{1}{2}} = \mathbf{U}_{(i),u,\tau} - \eta_\tau \nabla_{\mathbf{U}_{(i),u}} f_i$
 Update $\mathbf{V}_{(i),u,\tau+1} = \mathbf{V}_{(i),u,\tau} - \eta_\tau \nabla_{\mathbf{V}_{(i),u}} f_i$
 end for
 Calculate $\mathbf{U}_{s,\tau+1} = \frac{1}{N} \sum_{i=1}^N \mathbf{U}_{(i),s,\tau+1}$
end for
Return $\mathbf{U}_{s,R}$, $\mathbf{V}_{(i),s,R}$, $\mathbf{U}_{(i),u,R}$, $\mathbf{V}_{(i),u,R}$.

As for the initialization of Algorithm 1, entries of all updates are sampled from an i.i.d. Gaussian distribution with mean 0 and standard deviation 0.01. We find such simple initialization works well on all of our numerical experiments.

3.3. Convergence guarantee

Now we discuss the convergence of Algorithm 1. For notational simplicity, we use f to denote the objective $f(\mathbf{U}_{s,\tau}, \{\mathbf{V}_{(i),s,\tau}, \mathbf{U}_{(i),u,\tau}, \mathbf{V}_{(i),u,\tau}\})$

$= \sum_{i=1}^N f_i(\mathbf{U}_{s,\tau}, \mathbf{V}_{(i),s,\tau}, \mathbf{U}_{(i),u,\tau}, \mathbf{V}_{(i),u,\tau})$. The following theorem indicates that the gradient of the objective will converge to 0 at rate $O\left(\frac{1}{R}\right)$.

Theorem 1. Under the conditions

1. The norm of $\mathbf{Y}_{(i)}$'s is upper bounded by a constant G such that $\|\mathbf{Y}_{(i)}\| \leq G$.
2. There exists constant $B_1, B_2 > 0$ that upper bound the norm of iterates, i.e., $\|\mathbf{U}_{s,\tau}\|, \|\mathbf{U}_{(i),u,\tau}\|_F \leq B_1$ and $\|\mathbf{V}_{(i),s,\tau}\|, \|\mathbf{V}_{(i),u,\tau}\| \leq B_2$.
3. The stepsize is constant $\eta_\tau = \eta = O\left(\frac{1}{G+B_1^2+B_2^2}\right)$.

Algorithm 1 converges into stationary points,

$$\min_{\tau \in \{1 \dots R\}} \left\| \nabla f(\mathbf{U}_{s,\tau}, \{\mathbf{V}_{(i),s,\tau}, \mathbf{U}_{(i),u,\tau}, \mathbf{V}_{(i),u,\tau}\}) \right\|_F^2 = O\left(\frac{1}{R}\right) \quad (5)$$

Intuitively, Theorem 1 shows that when the norms of the data matrix $\mathbf{Y}_{(i)}$ are upper bounded, and the stepsize is chosen smaller than a threshold inversely proportional to the maximum norm of $\mathbf{Y}_{(i)}$, Algorithm 1 can converge into stationary points. We hope to emphasize that the $\mathbf{V}_{(i),s,\tau}$ and $\mathbf{U}_{(i),u,\tau}$ correction steps are indispensable to the convergence guarantee in Theorem 1. The proof for Theorem 1 is relegated to Appendix A.

3.4. Process signature visualization

After retrieving the shared and unique features, we are able to use them to visualize the process signatures. Manufacturing process signatures can be highly dynamic, and each image reflects the process signature at a specific timestamp. Hence, unique features in an image naturally characterize the transient behavior. Correspondingly, $\mathbf{U}_{(i),u} \mathbf{V}_{(i),u}^T$ captures the process signature in time i . We thus can visualize the process signature by plotting $\mathbf{U}_{(i),u} \mathbf{V}_{(i),u}^T$. Notice that such information is recovered purely through data-driven methods and does not require domain expertise.

3.5. Process signature characterization

The shared features $\mathbf{U}_s$ capture the generic information for the entire phase (or process); thus, they can be averaged to construct a *benchmark process signature* for the process, which is calculated as the mean $\frac{1}{N} \sum_{i=1}^N \mathbf{U}_s \mathbf{V}_{(i),s}^T$. As our approach can be applied to images from subsequent phases, such a benchmark process signature can be constructed for each phase. Together, they characterize the process signature evolution of the entire manufacturing process.

3.6. Anomaly detection: Unsupervised

If the process signatures come as in-situ images, they are usually not “labeled”, i.e., there is no ground truth for training a classification model for supervised anomaly detection. Therefore, we propose clustering analysis on the unique features extracted from process signatures for unsupervised anomaly (i.e., outlier) detection. To evaluate the proximity of two images, we propose two distance metrics.

Metric I: Similarity Angle

The principal angle can measure the similarity between the unique features of two spatter patterns. More specifically, for unique feature matrices $\mathbf{U}_{(i),u}$ and $\mathbf{U}_{(j),u}$, we use $\mathbf{Q}_{\mathbf{U}_{(i),u}}$ and $\mathbf{Q}_{\mathbf{U}_{(j),u}}$ to represent their corresponding Q -factors in the QR factorization. As $\mathbf{Q}_{\mathbf{U}_{(i),u}}$ and $\mathbf{Q}_{\mathbf{U}_{(j),u}}$ are orthonormal, the largest singular value $\sigma_1(\mathbf{Q}_{\mathbf{U}_{(i),u}}^T \mathbf{Q}_{\mathbf{U}_{(j),u}})$ is upper bounded by 1. We use d_{ij} to denote the difference between the largest singular value and 1.

$$d_{ij} = 1 - \sigma_1(\mathbf{Q}_{\mathbf{U}_{(i),u}}^T \mathbf{Q}_{\mathbf{U}_{(j),u}}) \quad (6)$$

d_{ij} is related to the smallest principal angle θ_{ij} between the subspace spanned by the columns of $\mathbf{U}_{(i),u}$ and $\mathbf{U}_{(j),u}$:

$$d_{ij} = 1 - \cos \theta_{ij}$$

When d_{ij} is larger, the subspaces are more dissimilar, and the spattering patterns are also less alike. $\sigma_1(\cdot)$ in (6) can be calculated by SVD. The similarity angle metric is only determined by unique features $\mathbf{U}_{(i),u}$, thus less affected by inaccurate estimates of coefficients $\mathbf{V}_{(i),u}$. However, calculating (6) requires performing QR decomposition on $\mathbf{U}_{(i),u}$, then SVD on $\mathbf{Q}_{\mathbf{U}_{(i),u}}^T \mathbf{Q}_{\mathbf{U}_{(j),u}}$, which can be time-consuming for large datasets.

Metric II: Frobenius norm of difference The principal angle only measures the difference between $\mathbf{U}_{(i),u}$ features. However, it is possible that the feature coefficient matrices $\mathbf{V}_{(i),u}$ and $\mathbf{V}_{(j),s}$ are also important in the similarity calculation. To this end, we can define the distance as,

$$d_{ij} = \|\mathbf{Y}_{(i)} - \mathbf{Y}_{(j)}\|_F + \|\mathbf{U}_s \mathbf{V}_{(i),s}^T - \mathbf{U}_s \mathbf{V}_{(j),s}^T\|_F + \|\mathbf{U}_{(i),u} \mathbf{V}_{(i),u}^T - \mathbf{U}_{(j),u} \mathbf{V}_{(j),u}^T\|_F \quad (7)$$

The distance measures the similarity between raw data and the shared and unique signals between two data matrices $\mathbf{Y}_{(i)}$ and $\mathbf{Y}_{(j)}$. Compared with similarity angle, Frobenius norm of difference can be faster to calculate on modern GPUs.

To visualize the relative distance relations among all the figures, we can calculate the distance of all (i, j) thermal image pairs to form an $N \times N$ distance matrix D , then perform t-SNE on D to visualize it on a 2-dimensional plane. t-SNE [57] will attempt to find a 2-dimensional representation of the images whose similarity structure will resemble the similarity structure in D locally. Such a visualization can demonstrate the relations among different spattering patterns.

3.7. Anomaly detection: Supervised

If the ground truth of whether an image is “regular” or “irregular” is known, then supervised anomaly detection can be implemented by training classification models with the extracted features.

More specifically, we denote the anomaly label by $y_i \in \{0, 1\}$. After running Algorithm 1 and obtaining optimal $(\mathbf{U}_s, \{\mathbf{V}_{(i),s}, \mathbf{U}_{(i),u}, \mathbf{V}_{(i),u}\})$, one can train a classifier h that maps $[\mathbf{Y}_{(i)}, \mathbf{U}_s \mathbf{V}_{(i),s}^T, \mathbf{U}_{(i),u} \mathbf{V}_{(i),u}^T]$ to the label y_i to minimize the empirical loss on the training set.

$$h: \mathbb{R}^{d \times 3n} \rightarrow \{0, 1\} \quad (8)$$

h can be any binary classifier, e.g., a neural network or a random forest classifier. The input of h contains not only the original data but also the extracted shared and unique information in each image because they can enrich the features of spatter patterns.

Real-time inference. With $\mathbf{U}_s$ extracted from historical data, when a new image $\mathbf{Y}_{(new)}$ comes in, one can efficiently calculate its corresponding shared coefficients $\mathbf{V}_{(new),s}$, unique features $\mathbf{U}_{(new),u}$ and the corresponding coefficients $\mathbf{V}_{(new),u}$ by,

$$\begin{aligned} \mathbf{V}_{(new),s} &= \mathbf{Y}_{(new)}^T \mathbf{U}_s (\mathbf{U}_s^T \mathbf{U}_s)^{-1} \\ \mathbf{U}_{(new),u} &= \text{SVD}_{r_2}(\mathbf{Y}_{(new)} - \mathbf{U}_s (\mathbf{U}_s^T \mathbf{U}_s)^{-1} \mathbf{U}_s^T \mathbf{Y}_{(new)}) \\ \mathbf{V}_{(new),u} &= (\mathbf{U}_{(new),u}^T \mathbf{U}_{(new),u})^{-1} \mathbf{U}_{(new),u}^T \mathbf{Y}_{(new)} \end{aligned}$$

where SVD denoted singular value decomposition and SVD_{r_2} denotes the top r_2 singular column vectors of a matrix.

From calculated feature matrices $\mathbf{V}_{(new),s}$, $\mathbf{U}_{(new),u}$, and $\mathbf{V}_{(new),u}$, the input matrices $[\mathbf{Y}_{(new)}, \mathbf{U}_s \mathbf{V}_{(new),s}^T, \mathbf{U}_{(new),u} \mathbf{V}_{(new),u}^T]$ can be constructed. Then, the trained classifier could make predictions based on the input matrices. The classification procedure extracts the shared and unique components from $\mathbf{Y}_{new}$, thus augmenting the feature space and leading to potentially better prediction performance.

4. Case study

In this section, we apply the proposed method to a synthetic dataset and two real-life metal AM datasets. On the synthetic dataset, we showcase the statistical and computational advantages of PFE. The first AM dataset is the 2018 AM Benchmark Test Series (NIST) [31]. It is an unlabeled dataset. Therefore, we analyze it through the lens of process signature characterization and unsupervised anomaly detection. The second contains thermal images of the AM process of a thin-walled structure [58]. The dataset is labeled according to whether pores are existent at the time thermal images are taken. We study the data by supervised anomaly detection.

In this section, we use a few popular feature extraction algorithms from the literature in Section 2 as benchmark methods. These methods are summarized below.

- Dual control chart [59]: The method applies multilinear principal component analysis (MPCA) to extract orthonormal features, then uses two control charts on the monitoring feature and residual for anomaly detection.
- Boundary feature extraction [17]: The method first parameterizes the iso-thermal contour of the melt pool in a polar coordinate, then uses the parametrized contour to perform statistical analysis.
- JIVE [27]: We apply JIVE to extract the shared and unique features from different frames.
- Standard PCA: As a comparison, we also use standard PCA on the pooled video frames to find the principal components.

We choose benchmark methods focusing on the feature extraction procedures as they are most relevant to PFE. Code to generate the results in this section is available in the linked [Github repository](#).

4.1. Synthetic dataset

In this section, we compare the performance of PFE, JIVE, and the standard version of PerPCA [24] on the synthetic data $\{Y_{(i)}\}$. For data generation, we first sample the features U_s , $U_{(i),u}$ and their corresponding coefficients $V_{(i),s}$, $V_{(i),u}$ from Gaussian distributions, then generate $Y_{(i)}$ according to model (1). We set $d = 100$, $N = 100$, and $n = 100$ to simulate a moderate-scale dataset. We run three algorithms from the same initialization and calculate the subspace error

$$\left\| U_{s,\tau} \left(U_{s,\tau}^T U_{s,\tau} \right)^{-1} U_{s,\tau}^T - U_s \left(U_s^T U_s \right)^{-1} U_s^T \right\|_F^2 + \frac{1}{N} \sum_{i=1}^N \left\| U_{(i),u,\tau} \left(U_{(i),u,\tau}^T U_{(i),u,\tau} \right)^{-1} U_{(i),u,\tau}^T - U_{(i),u} \left(U_{(i),u}^T U_{(i),u} \right)^{-1} U_{(i),u}^T \right\|_F^2$$

at every iteration τ . The subspace error measures the difference between estimated shared and unique components and the ground truth. We plot the loss curve in the left graph of Fig. 3. To compare the runtime, we also collect the wall clock time for each iteration and plot the mean and standard deviation of the per-iteration runtime in the middle graph of Fig. 3. The error to time curve is plotted in the right graph of Fig. 3.

From the left graph of Fig. 3, it is clear that PerPCA and PFE have similar final subspace errors, which are lower than that of JIVE. This is expected as PFE and the standard version of PerPCA have the same statistical error guarantee, while the JIVE algorithm is based on heuristics. The middle graph of Fig. 3 shows PFE is much faster than the standard PerPCA: PFE takes around 0.03 s for each iteration, while PerPCA takes around 0.09 s, which is almost 3 times slower. The right graph of Fig. 3 shows the combined performance of statistical and computational efficiency. One can see that, indeed, PFE converges faster than standard PerPCA. The comparisons in Fig. 3 highlight the statistical and computational advantages of PFE.

4.2. AM benchmark test series

4.2.1. Data description

The 2018 AM Benchmark Test Series from NIST provides in-situ thermography of an L-PBF-ed metal bridge structure made with IN625 and stainless-steel 15-5. A sample thermal image from the dataset is shown in Fig. 4. The bridge structure was completed with two builds; each build fabricated four parts (eight parts in total). In-situ thermal measurements of a select region on one of the parts within each build were acquired at 1800 frames per second. The part had 12 legs of varying size (5 mm $\times$ 5 mm, 5 mm $\times$ 2.5 mm, and 0.5 mm $\times$ 5 mm), where each leg was 5 mm tall, then used a 45-degree overhang to transition into the bridge structure with a constant cross-section. Each part was manufactured using 0.02 mm layer thickness, a programmed laser power of 195 W traveling at a scan speed of 800 mm/s, and the hatch spacing was 0.1 mm. The part was completed with 624 layers and a total build time of nearly 9.5 h [31].

The proposed PFE and anomaly detection methods are applied to in-situ thermal videos of selected layers for the bridge body from build 1. One in-situ thermal video is available for each layer. Per the build time of the layer, the video length (or, equivalently, the number of frames) may differ, ranging from (roughly) 400 to > 3000 frames. Each frame is a gray-scale image with size 126 $\times$ 360, recording the spattering behavior (or a transient process signature) at a particular printing time. The pixel values represent the radiant temperature (not accounting for emissivity) measured during each layer.

4.2.2. Data preprocessing

In a video, the laser scan direction could vary, so the moving direction of spatter tracks could be either vertical or horizontal. Occasionally, the spatter tracks might move outside the camera view, leading to some “void” or noisy frames containing no process signatures. This case study focuses on the thermal images showing a complete spatter pattern. The noisy frames in a video do not provide meaningful information about LBMAM and are not good for demonstrating personalized feature extraction. By examining the data, we find that these frames usually correspond to some sparse noise, and their total temperature is low. Therefore, we use a thresholding criterion to filter the noisy frames and only retain the “complete” frames: we calculate the summation of temperature recorded on each pixel and only keep frames with total temperature larger than 10^6 . Such a thresholding procedure will remove the pictures that only contain weak signals like background noise.

We use thermal images from Layer 361 from build 1 in Fig. 4 as an example of our layer-wise analysis. One can see that the comet-shaped high-temperature region only occupies one small part of the entire domain of measurement. To focus our analysis on the high-temperature region, we first clip the entire thermal image to a smaller 100 $\times$ 200 image that contains the high-temperature region.

To focus on the spatial patterns in the clipped thermal images, we adopt the image patching technique [60,61]. The 100 $\times$ 200 image is separated into 200 10 $\times$ 10 block patches. We then flatten all pixels in one patch into a row vector and stack all row vectors to construct the *folded matrix*. It is straightforward to calculate that the folded matrix has dimensions 200 by 100. The construction of a folding matrix essentially permutes all the pixels in the clipped thermal image. The image patching allows us to extract several features from an image that can characterize the spatial patterns of the frame. As one folded column vector is constructed by concatenating pixels in one specific position in each block patch, it distills the high-level information from the entire image. Such information is descriptive of the common and unique patterns in the image, thus suitable for the analysis of personalized modeling. It is worth noting that the image patching mechanism is essentially a permutation of pixels, thus deterministic and easily invertible. We call the inverse of matrix folding *unfolding*. Fig. 5 is an illustration of the folding procedure.

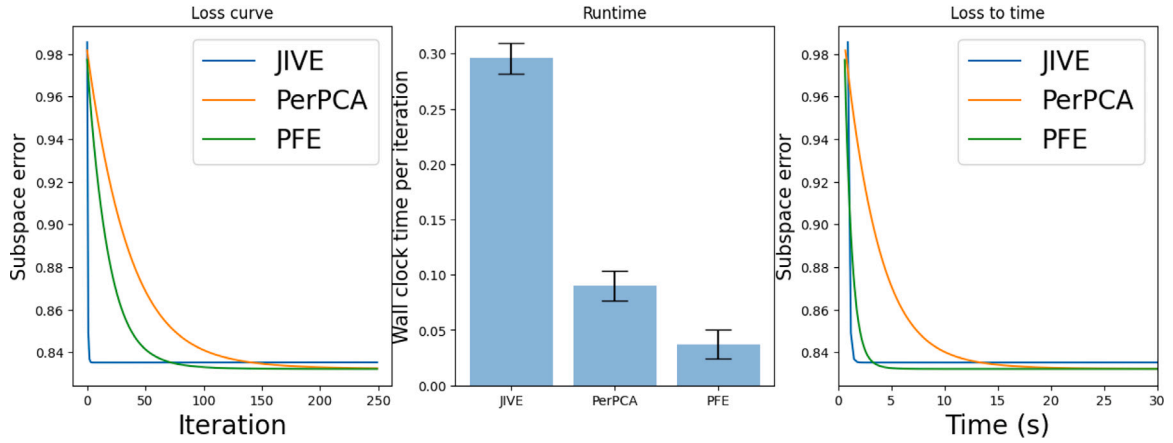


Fig. 3. Left: Loss to iteration. Middle: Wall clock time (second) per iteration. Right: Loss to wall clock time.

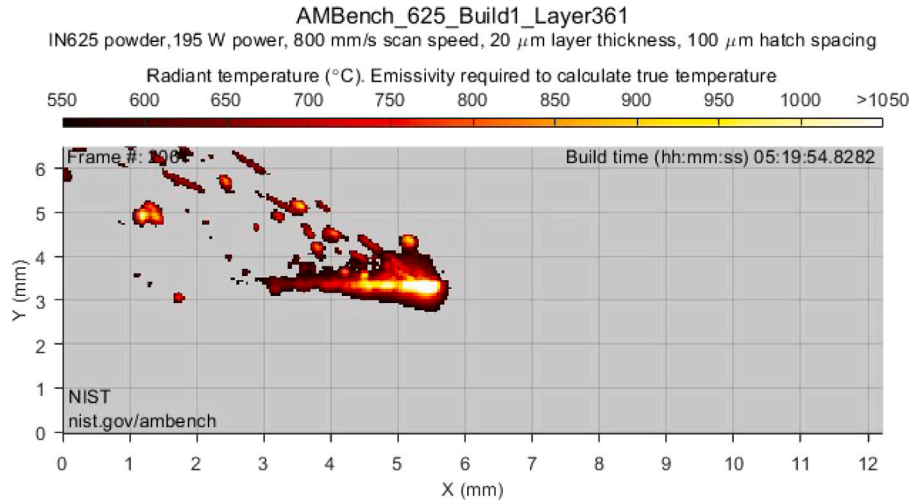


Fig. 4. A sample of thermal-image at one time [31].

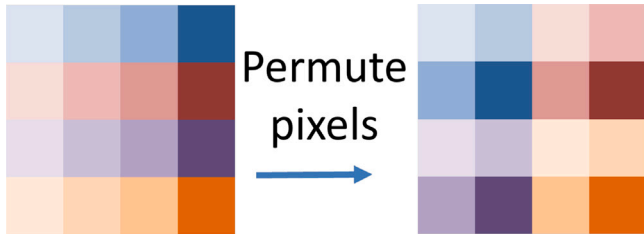


Fig. 5. An illustration of the folding process. The left figure is an example of a thermal image. It is subdivided into 4 2×2 blocks. The right figure is the folded matrix.

Such construction procedure exploits the philosophy of subsampling [62] and dilation [63,64], thus can encode rich knowledge about the geometric properties of high radiant temperature region. Notice that after the matrix folding, the resulting $\mathbf{Y}_{(i)}$ has dimension d by n , where d is the number of patches, and n is the number of pixels in each block patch.

4.2.3. Unique features vs. Shared features

With the folded matrices, we can follow the procedures in Section 3.2 to extract shared and unique features. We first normalize the folded matrices by row, then apply Algorithm 1 with constant stepsize $\eta_\tau = 0.05$ and $R = 400$. We choose $r_1 = 15$ and $r_2 = 50$. The rank of the unique feature matrix is higher as unique features should capture

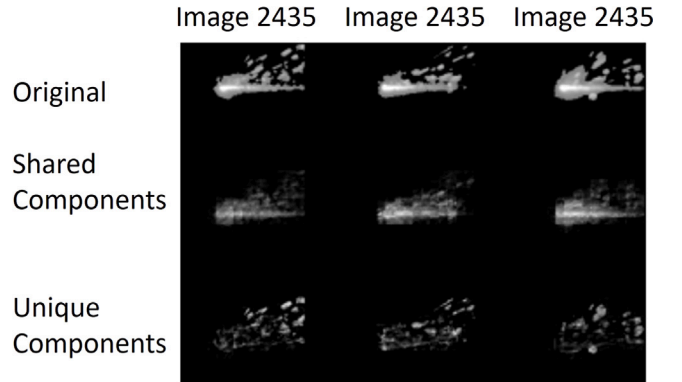


Fig. 6. The shared and unique components of three sample thermal images.

the detailed structure of spattering patterns, which often have irregular shapes.

One result is visualized in Fig. 6. The first row shows the clipped thermal image. The second row shows the recovered $\mathbf{U}_s \mathbf{V}_{(i),s}^T$ after unfolding. And the third row shows the recovered $\mathbf{U}_{(i),u} \mathbf{V}_{(i),u}^T$ after unfolding.

It is natural to see that the shared components capture the shape information about the comet-shaped high radiant temperature region in

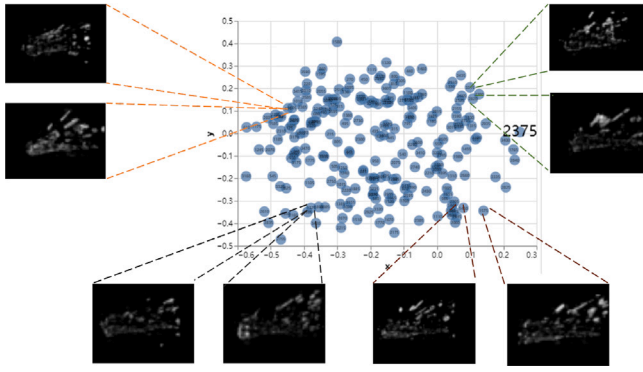


Fig. 7. t-SNE of all the unique features.

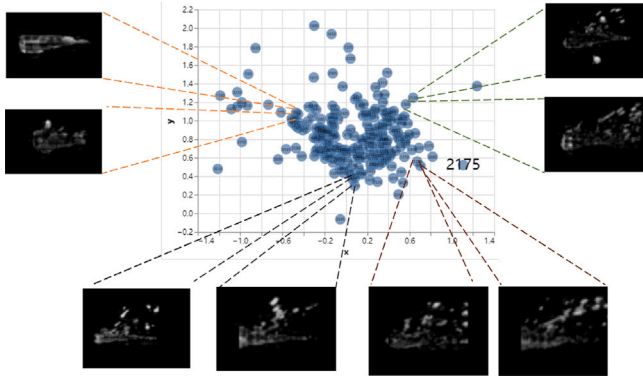


Fig. 8. t-SNE based on the top eigenvectors of each individual thermal image.

the middle, while the unique components can find the spatter patterns scattered around the image.

4.2.4. Anomaly detection performance

In this section, we present the results of using the extracted unique features to understand the similarity of spatter patterns and detect the outliers. We calculate the pairwise principal angle distance d_{ij} between time frame i and j defined in (6). Then, we use t-SNE to embed the high-dimensional distance information onto a 2-dimensional plane and visualize the results in Fig. 7. The perplexity of t-SNE is set to 200.

In Fig. 7, every point represents a thermal image. The closeness of the two points indicates the similarity between the unique features of the two thermal images. One can tell that the points close to each other have similar spattering patterns, and the points that are far from other points have spattering patterns different from other thermal images. The similarity structure shown in Fig. 7 can justify the outlier detection procedures discussed in Section 3.6.

In comparison, we also run standard PCA on each individual thermal image independently to extract the top eigenvectors, calculate the distance between different images similarly, and then draw the t-SNE of the estimated distance. Results are shown in Fig. 8.

Points in Fig. 8 are more concentrated in the middle area, suggesting that PCA-extracted features are less discriminating in analyzing the spattering patterns. Also, adjacent points in Fig. 8 are less alike compared to Fig. 7. To highlight the disparities between the PFE and PCA approaches, we pick two close images from t-SNE of each method. Results are plotted in Fig. 9.

In the first row of Fig. 9, the two neighbors found by PFE have similar spatter patterns, with consistent placement and orientation in the top right corner. However, the two neighbors found by PCA exhibit quite distinct spatter patterns. Notably, one image contains a large spatter in the lower section, while its close neighboring image

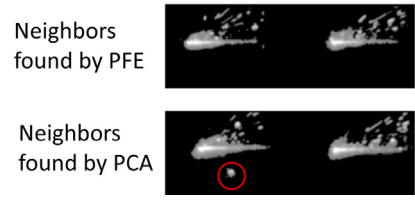


Fig. 9. Two neighbors found by PFE and PCA. One image has a spatter in the circled region. Its close neighbor does not have any spatter at the same location.

Table 1

The average spattering pattern on eight different layers.

361	363	365	367
411	413	415	417

lacks any spatter at that location. The comparison again highlights the importance of unique features extracted from PFE.

4.2.5. Layer-wise benchmark process signatures

As described in Section 3.5, the shared features recovered by Algorithm 1 contain the archetypal spatter patterns for the specific layer. We thus calculate the average spatter pattern to construct the layer-wise benchmark process signature as introduced in Section 3.5. The results from different layers are plotted in Table 1.

The process signature data were selected from adjacent layers of the bridge body from build 1. In Table 1, one can see that the benchmark process signatures from these layers are similar, indicating a notable but relatively small variation among these layers. The result indicates region-wise stability of the benchmark process signatures during L-PBF printing.

4.2.6. Kernel version

Though formulation (1) models linear features, it is straightforward to extend to nonlinear ones by exploiting kernel methods. We use the Gaussian kernel to construct the kernel covariance matrices. More specifically, for two matrices $\mathbf{X}_1 \in \mathbb{R}^{q \times n_1}$ and $\mathbf{X}_2 \in \mathbb{R}^{q \times n_2}$, we define $K(\mathbf{X}_1, \mathbf{X}_2) \in \mathbb{R}^{n_1 \times n_2}$ to be a matrix whose ij -th entry is $K(\mathbf{X}_1, \mathbf{X}_2)_{ij} = \exp\left(-\frac{\|(\mathbf{X}_1)_i - (\mathbf{X}_2)_j\|^2}{2\sigma^2}\right)$, where $(\mathbf{X}_1)_i$ is the i th row of $\mathbf{X}_1$. We use inducing points to characterize the spatial features in each frame. The inducing points serve as representatives of thermal image frames. In this experiment, we generate 10000 random sparse inducing points $\mathbf{Y}_s \in \mathbb{R}^{10000 \times 200}$ so that each inducing point has only 1 nonzero element. The sparse inducing points capture localized features in video frames. Then we calculate $\tilde{\mathbf{Y}}_{(i)} = K(\mathbf{Y}_s, \mathbf{Y}_{(i)}) \in \mathbb{R}^{10000 \times 100}$. Next, we run Algorithm 1 on $\{\tilde{\mathbf{Y}}_{(i)}\}$, and retrieve the estimates of shared and unique features, then apply the technique in [65] to reconstruct the pre-images associated with the features. We show two frames and the reconstructed shared and unique components in Fig. 10.

In Fig. 10, it is clear that the shared components recovered by kernel PFE have fewer artifacts compared with the standard PFE. Also, the unique components in kernel PFE have smoother boundaries, thus better reflecting the geometric properties of the spatters. The results are understandable as the kernel version of Algorithm 1 relaxes the implicit assumption of linear feature decomposition in (1) and thus is less restrictive for real data when the “feature linearity” assumption unnecessarily holds.

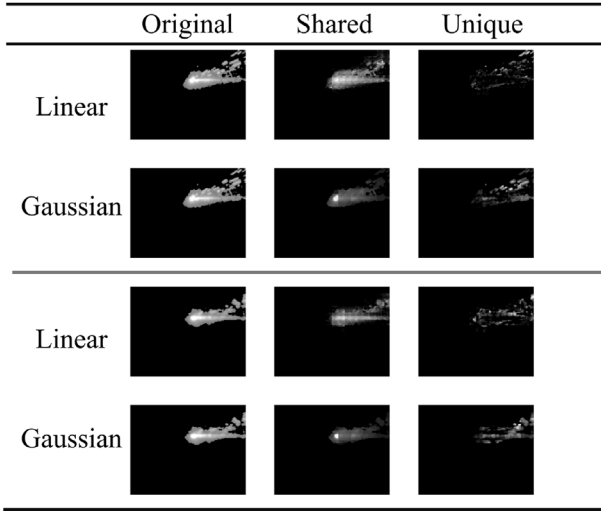


Fig. 10. Two frames and the recovered shared and unique components from linear PFE and kernel PFE with a Gaussian kernel.

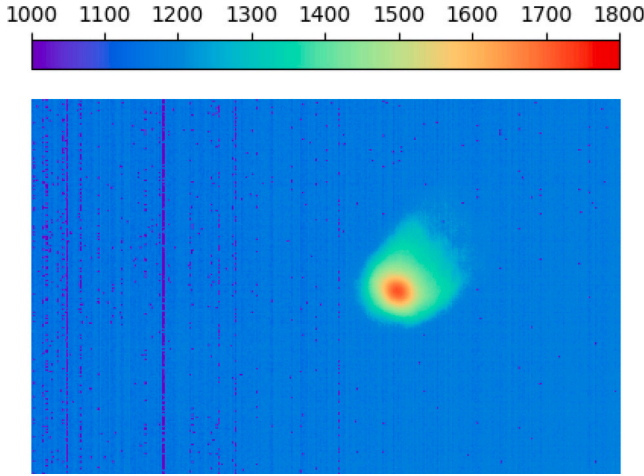


Fig. 11. A sample of thermal-image from the dataset [58].

4.3. Ti-6Al-4V thin-walled structure

4.3.1. Data description

The dataset contains thermal measurements collected by a Stratonic Dual-wavelength pyrometer during the direct energy deposition (DED) process of a thin-walled structure [58]. The pyrometer captured a top-down view of the melt pool, which was the heat-affected zone under the deposition head during printing. The pyrometer thermal images were provided in Comma Separated Values format centered around the melt pool with temperatures > 1000 °C, indicative of the shape and distribution of temperature values. After the printing process, Nikon X-ray Computed Tomography (XCT) XT H225 captured internal porosity reflective of a lack of fusion. XCT porosity labels of sizes between 0.05 mm to 1.00 mm were registered within 0.5 mm of the melt pool image coordinate, thus labeling a pyrometer thermal image with its porosity records (i.e., “no porosity”/“porous”). Images with a “porous” label were considered “abnormal”. The labeled thermal images can be used to train supervised anomaly detection methods (see Fig. 11).

4.3.2. Data preprocess

The shape of the original thermal image is 752×480 . Still, it can be seen that only a small portion of the entire thermal image contains the

high-temperature melt pool. Thus, as a simple data dimension reduction technique, we also clip the thermal image into a 200×200 sub-image encapsulating the ROI. Then, we also use the folding and unfolding procedures described in Section 4.2.2 to build the folded matrices.

4.3.3. Visualization of extracted features

We then apply PFE on the folded matrices to extract the shared and unique features. We set r_1 to 10 and r_2 to 20, and choose the stepsize η_τ to be 10^{-5} . We choose a smaller stepsize than the NIST dataset as the norm $\|Y_{(i)}\|$ is larger. The extracted shared and unique components are plotted in Fig. 12.

For each thermal image, we plot the original (processed) data and the shared and unique components in one column. All images in one row are plotted in one color scale. We plot 2 normal samples (not porous) and 6 abnormal samples (porous) in Fig. 12. One can see that the melt pools of normal samples tend to have regular shapes. Also, the shared components can capture most information about the thermal pattern, while the unique components do not have a strong signal in the central region, suggesting that these images do not have large “eccentricity”. In contrast, for the abnormal samples, the melt pool shapes are more irregular in the original data. It can also be seen that the variability is not well-represented by shared features. In the central parts of the unique components, there are often regions with high or low temperatures, suggesting that the melt pool’s thermal behavior is very different from others in some instances. The irregular thermal behavior is caused by process instability. The low temperature or smaller size of abnormal melt pool is highly associated with unstable laser input and results in a lack of fusion porosity [66] (or other less common defects beyond the scope of this dataset).

4.3.4. Feature visualization

The recovered shared and unique features can also characterize similar information among different frames. In this section, we perform a clustering analysis based on the distance metrics defined in Section 3.6. We calculate the distance d_{ij} between all pairs of images i and j according to the Frobenius norm of difference metric in (7) to build an $N \times N$ distance matrix D . From D , we perform t-SNE with perplexity 40 to visualize the similarity structure on a 2D plane. The result is shown in Fig. 14. As a benchmark, we also present the t-SNE plot of the raw data in Fig. 13.

Fig. 13 has a clearer clustering structure than Fig. 14 as the abnormal samples are clustered on the left, and the normal samples are more uniformly distributed on the right. While in Fig. 14, the clustering pattern is less conspicuous. The comparison highlights the discriminating power of features learned by Algorithm 1.

Also, we introduce a numerical metric to measure the quality of the t-SNE results. As there are two groups of samples, the normal group and the abnormal group, we can separate the total variance of t-SNE-extracted embeddings into the summation of inter-group variance and variance of each group.

Total variance = Inter-group variance + Variance within each group

Inspired by ANOVA, we introduce a t-SNE score that measures the portion of inter-group variance in total variance,

$$\text{Variance Ratio Score} = \frac{\text{Inter-group variance}}{\text{Total variance}} \quad (9)$$

The score will take a value between 0 and 1. When the score is higher, the two groups of samples are more distinct. Thus, the t-SNE embeddings have higher quality.

The calculated VRS for PFE and several benchmark methods are reported in the last column of Table 2. PFE stands out with the highest VRS, demonstrating the better quality of its t-SNE embeddings.

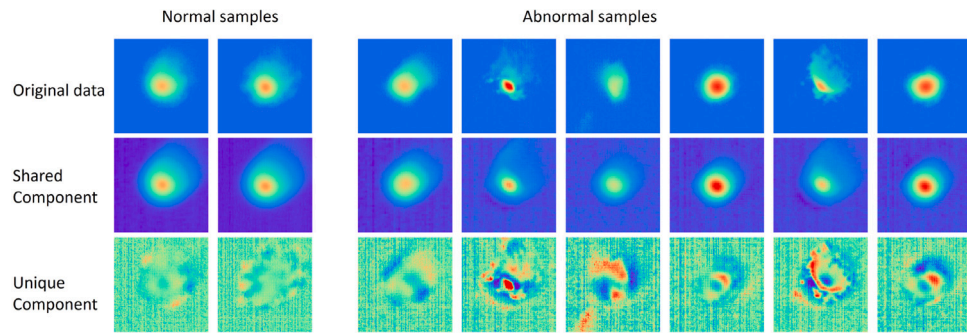


Fig. 12. The shared and unique components of two normal and six abnormal thermal images. Red denotes high temperature, and purple-blue denotes low temperature. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

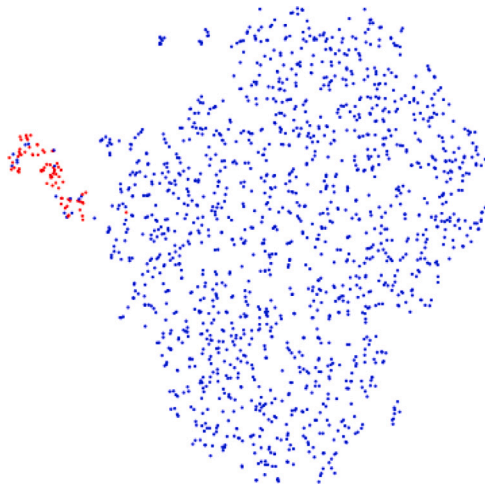


Fig. 13. t-SNE of the extracted features.

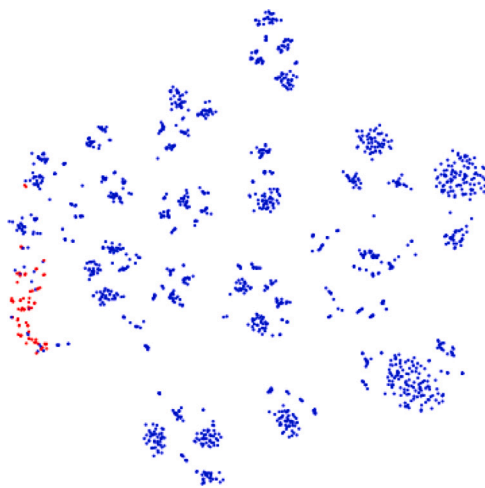


Fig. 14. t-SNE of raw images.

4.3.5. Porosity prediction

Since the anomaly labels are available, we can train classifiers to predict the anomalies. As described in Section 3.7, we concatenate the learned shared and unique components with the original data to construct the input matrix for image i as $[\mathbf{Y}_{(i)}, \mathbf{U}_s \mathbf{V}_{(i),s}^T, \mathbf{U}_{(i),u} \mathbf{V}_{(i),u}^T]$. Then, we randomly split the entire dataset into a 90% training set and a 10% test set. We train a random forest classifier from the input matrix

Table 2

Anomaly detection and clustering performance of different algorithms. VRS stands for Variance Ratio Score introduced in (9).

	Precision	Recall	F- score	VRS
Dual control charts [59]	0.97	0.62	0.76	–
Boundary feature [17]	0.78 ± 0.08	0.17 ± 0.02	0.27 ± 0.03	0.0009
Original data	0.85 ± 0.02	0.74 ± 0.02	0.78 ± 0.03	0.08
PCA	0.84 ± 0.02	0.88 ± 0.02	0.85 ± 0.02	0.05
JIVE [27]	0.82 ± 0.02	0.94 ± 0.02	0.87 ± 0.02	0.13
PFE	0.81 ± 0.02	0.95 ± 0.02	0.87 ± 0.02	0.15

Table 3

Wall clock time for two feature extraction algorithms.

Method	JIVE	PFE
Time (s)	755	275

to the anomaly label on the training set, then evaluate the predicting performance on the test set.

As the dataset consists of 1493 normal images and 71 abnormal images, the proportions of normal and abnormal images are highly unbalanced. Classification accuracy is not a good metric for evaluating the predictive performance of such unbalanced data. Thus, we exploit a confusion matrix and report recall, precision, and the F-score. The higher recall and precision are, the better performance the classifier has. F-score is the harmonic mean of recall and precision, hence serving as a comprehensive metric for classification performance. We run each experiment from 18 different random seeds and calculate the evaluation metrics for each experiment. For benchmarks, we also extract features using boundary feature extraction [17], JIVE [27], standard PCA, and exact original images and train a random forest classifier on these features to predict anomaly labels.

A summary table was added for a quantitative demonstration of the performance. The Variance Ratio Score is a deterministic metric that does not involve randomness, so we do not report its standard deviation.

Table 2 shows that among all the algorithms, PFE has the highest F-score, validating its superior performance in anomaly detection. Also, in terms of clustering, PFE achieves the highest variance ratio. This effectively emphasizes the discriminative power of the extracted features.

We run another analysis on the computational costs of PFE and JIVE. As a simple comparison, we implement both methods on the same desktop with NVIDIA GeForce RTX 3080 and collect the wall clock time of each method. Results are reported in Table 3.

Table 3 shows significant improvement of PFE: compared with JIVE, PFE saves 64% runtime. The huge decrease in runtime highlights the computational efficiency of PFE.

Therefore, the comparisons showcase PFE's superior statistical and computational performance.

4.4. Method generalization

Results in this section advocate the use of Algorithm 1 (or its kernel version) for shared and unique feature decoupling for manufacturing process signature characterization and anomaly detection. In addition to imaging data, PFE techniques can also be applied to other data forms for manufacturing process signatures where common properties/requirements and personalized characteristics coexist. In manufacturing fields, such a situation is frequently confronted. For example, a number of AM processes can leverage Algorithm 1 and the proposed anomaly detection methods for process monitoring and quality control. Their printing mechanism, which is layer-by-layer accumulation (or line-by-line accumulation [67,68]), naturally creates the coexistence of layer-wise (or line-wise) shared features and unique features in in-situ process signatures. Personalized models may as well become a useful tool for constructing the layer-wise (or line-wise) benchmark process signatures for various AM processes.

5. Conclusion & discussion of future work

This study adopted a new statistical dimensional reduction method for finding shared and unique feature extraction from manufacturing imaging data. It developed a novel PFE method for process signature characterization and anomaly detection. The method has a distinctive capability of extracting the process-wise shared features and unique features simultaneously and separately. Therefore, the extracted features are more interpretable and respect the fundamental physics, i.e., the evaluation of process signatures during manufacturing. The unsupervised anomaly detection strategy proposed upon the recovered shared and unique features achieved remarkable performance on the 2018 AM Benchmark Test Series.

The work has demonstrated the superiority and feasibility of using PFE to improve the performance of manufacturing process monitoring and quality control. Specifically, decoupled features can help uncover insightful knowledge, whether in shared or unique parts. The features enable classification and clustering analysis on unique features, thus yielding better accuracy since differences are more explicit when shared features are removed. By monitoring changes in the unique features, anomalies can be detected better and faster when data is collected over time. Furthermore, PFE will underlie personalized predictive modeling by selectively transferring common knowledge from one data source to another, reducing the negative transfer of knowledge and enhancing personalized models.

In the future, this work will be extended in several directions. First, with the availability of defect records, the detected anomalies can be connected with defect occurrences and build a relationship between the process signatures and the finished parts' quality. This exploration will be our next step on the data availability. Second, PFE and the anomaly detection strategy can be generalized to both machining processes and AM processes, especially AM processes for polymers/ceramics, by replacing process signature data and applications. Third, broader applications of personalized models and the proposed anomaly detection methods will be explored for manufacturing systems, where there are multiple manufacturing processes generating local data and meanwhile fulfilling common requirements or given common resources.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was partially supported by the National Science Foundation CMMI CAREER grant 2144147.

Appendix A. Proof of Theorem 1

In this section, we present the proof of Theorem 1. The proof consists of three steps. We will firstly show that the objective f is gradient Lipschitz continuous. Then, based on Lipschitz continuity, we will show a sufficient decrease inequality with the help of a special property of objective f . Eventually, we will prove Theorem 1.

Gradient Lipschitz continuity is the key property of the objective (3) that leads to the convergence of Algorithm 1. We first review its definition. We use $C \subset \mathbb{R}^n$ to denote a convex set. A differentiable function $f : C \rightarrow \mathbb{R}$ is L -gradient Lipschitz continuous [69] if for any $x, y \in C$, the following holds,

$$\|\nabla f(x) - \nabla f(y)\| \leq L \|x - y\| \quad (\text{A.1})$$

The Lipschitz constant L signifies the rate of gradient change for the function f caused by unit changes in its inputs. It is proved as Lemma 1.2.3 in [69] that if f is L -Lipschitz continuous, then

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2} \|y - x\|^2, \quad \forall x, y \in C \quad (\text{A.2})$$

The property (A.2) will play a critical role in establishing Theorem 1.

Throughout this section, we use $B(\zeta_1, \zeta_2)$ to denote the set of solutions with bounded norms:

$$B(\zeta_1, \zeta_2) = \{U_s, \{U_{(i,u)}\}, \{V_{(i,s)}\}, \{V_{(i,u)}\} \mid \|U_s\|, \|U_{(i,u)}\| \leq \zeta_1, \|V_{(i,s)}\|, \|V_{(i,u)}\| \leq \zeta_2\} \quad (\text{A.3})$$

It is easy to see that $B(\zeta_1, \zeta_2)$ is a convex set.

The following lemma gives an upper bound on the Lipschitz constant when all iterates are bounded.

Lemma A.1 (Lipschitz Continuity). In region $B(B_1, B_2)$ as defined in (A.3), f_i 's are Lipschitz continuous:

$$\begin{aligned} & \|\nabla f_i(U'_s, V'_{(i,s)}, U'_{(i,u)}, V'_{(i,u)}) - \nabla f_i(U_s, V_{(i,s)}, U_{(i,u)}, V_{(i,u)})\|_F \\ & \leq L \sqrt{\|U'_s - U_s\|_F^2 + \|V'_{(i,s)} - V_{(i,s)}\|_F^2 + \|U'_{(i,u)} - U_{(i,u)}\|_F^2 + \|V'_{(i,u)} - V_{(i,u)}\|_F^2} \end{aligned} \quad (\text{A.4})$$

for $\{U'_s, \{V'_{(i,s)}\}, \{U'_{(i,u)}\}, \{V'_{(i,u)}\}\}, \{U_s, \{V_{(i,s)}\}, \{U_{(i,u)}\}, \{V_{(i,u)}\}\} \in B(B_1, B_2)$, where L_g and L are constants dependent on B_1, B_2, G ,

$$L = 2\sqrt{(G + 3B_1B_2)^2 + B_1^2B_2^2 + 2B_1^4 + B_2^4 + B_2^4}$$

Proof. The Lipschitz continuity of f_i can be proved by direct calculation. We will calculate the gradient of f_i over each variable and bound the norm of the difference of the gradients.

$$\begin{aligned} & \nabla_{U_s} f_i(U'_s, V'_{(i,s)}, U'_{(i,u)}, V'_{(i,u)}) - \nabla_{U_s} f_i(U_s, V_{(i,s)}, U_{(i,u)}, V_{(i,u)}) \\ & = 2 \left(U'_s (V'_{(i,s)})^T + U'_{(i,u)} (V'_{(i,u)})^T - Y_{(i)} \right) V'_{(i,s)} \\ & \quad - 2 \left(U_s (V_{(i,s)})^T + U_{(i,u)} (V_{(i,u)})^T - Y_{(i)} \right) V_{(i,s)} \end{aligned}$$

Also by triangle inequalities, when $(U'_s, V'_{(i,s)}, U'_{(i,u)}, V'_{(i,u)})$ and $(U_s, V_{(i,s)}, U_{(i,u)}, V_{(i,u)})$ are in $B(B_1, B_2)$, we have:

$$\begin{aligned} & \|\nabla_{U_s} f_i(U'_s, V'_{(i,s)}, U'_{(i,u)}, V'_{(i,u)}) - \nabla_{U_s} f_i(U_s, V_{(i,s)}, U_{(i,u)}, V_{(i,u)})\|_F \\ & \leq 2B_2^2 \|U'_s - U_s\|_F + 2(G + 3B_1B_2) \|V'_{(i,s)} - V_{(i,s)}\|_F \\ & \quad + 2B_2^2 \|U'_{(i,u)} - U_{(i,u)}\|_F + 2B_1B_2 \|V'_{(i,u)} - V_{(i,u)}\|_F \end{aligned}$$

Then on the derivative over $V_{(i,s)}$,

$$\begin{aligned} & \nabla_{V_{(i,s)}} f_i(U'_s, V'_{(i,s)}, U'_{(i,u)}, V'_{(i,u)}) - \nabla_{V_{(i,s)}} f_i(U_s, V_{(i,s)}, U_{(i,u)}, V_{(i,u)}) \\ & = 2 \left(U'_s (V'_{(i,s)})^T + U'_{(i,u)} (V'_{(i,u)})^T - Y_{(i)} \right)^T U'_s \\ & \quad - 2 \left(U_s (V_{(i,s)})^T + U_{(i,u)} (V_{(i,u)})^T - Y_{(i)} \right)^T U_s \end{aligned}$$

Thus by similar calculations, we have,

$$\begin{aligned} & \left\| \nabla_{\mathbf{V}_{(i),s}} f_i(\mathbf{U}'_s, \mathbf{V}'_{(i),s}, \mathbf{U}'_{(i),u}, \mathbf{V}'_{(i),u}) - \nabla_{\mathbf{V}_{(i),s}} f_i(\mathbf{U}_s, \mathbf{V}_{(i),s}, \mathbf{U}_{(i),u}, \mathbf{V}_{(i),u}) \right\|_F \\ & \leq 2(G + 3B_1B_2) \left\| \mathbf{U}'_s - \mathbf{U}_s \right\|_F + 2B_1^2 \left\| \mathbf{V}'_{(i),s} - \mathbf{V}_{(i),s} \right\|_F \\ & + 2B_1B_2 \left\| \mathbf{U}'_{(i),u} - \mathbf{U}_{(i),u} \right\|_F + 2B_1^2 \left\| \mathbf{V}'_{(i),u} - \mathbf{V}_{(i),u} \right\|_F \end{aligned}$$

And,

$$\begin{aligned} & \left\| \nabla_{\mathbf{U}_{(i),u}} f_i(\mathbf{U}'_s, \mathbf{V}'_{(i),s}, \mathbf{U}'_{(i),u}, \mathbf{V}'_{(i),u}) - \nabla_{\mathbf{U}_{(i),u}} f_i(\mathbf{U}_s, \mathbf{V}_{(i),s}, \mathbf{U}_{(i),u}, \mathbf{V}_{(i),u}) \right\|_F \\ & \leq 2B_2^2 \left\| \mathbf{U}'_s - \mathbf{U}_s \right\|_F + 2B_1B_2 \left\| \mathbf{V}'_{(i),s} - \mathbf{V}_{(i),s} \right\|_F \\ & + 2B_2^2 \left\| \mathbf{U}'_{(i),u} - \mathbf{U}_{(i),u} \right\|_F + 2(G + 3B_1B_2) \left\| \mathbf{V}'_{(i),u} - \mathbf{V}_{(i),u} \right\|_F \end{aligned}$$

Also,

$$\begin{aligned} & \left\| \nabla_{\mathbf{V}_{(i),u}} f_i(\mathbf{U}'_s, \mathbf{V}'_{(i),s}, \mathbf{U}'_{(i),u}, \mathbf{V}'_{(i),u}) - \nabla_{\mathbf{V}_{(i),u}} f_i(\mathbf{U}_s, \mathbf{V}_{(i),s}, \mathbf{U}_{(i),u}, \mathbf{V}_{(i),u}) \right\|_F \\ & \leq 2B_1B_2 \left\| \mathbf{U}'_s - \mathbf{U}_s \right\|_F + 2B_1^2 \left\| \mathbf{V}'_{(i),s} - \mathbf{V}_{(i),s} \right\|_F \\ & + 2(G + 3B_1B_2) \left\| \mathbf{U}'_{(i),u} - \mathbf{U}_{(i),u} \right\|_F + 2B_1^2 \left\| \mathbf{V}'_{(i),u} - \mathbf{V}_{(i),u} \right\|_F \end{aligned}$$

Combining the four inequalities, we have:

$$\begin{aligned} & \left\| \nabla f_i(\mathbf{U}'_s, \mathbf{V}'_{(i),s}, \mathbf{U}'_{(i),u}, \mathbf{V}'_{(i),u}) - \nabla f_i(\mathbf{U}_s, \mathbf{V}_{(i),s}, \mathbf{U}_{(i),u}, \mathbf{V}_{(i),u}) \right\|_F^2 \\ & = \left\| \nabla_{\mathbf{U}_s} f_i(\mathbf{U}'_s, \mathbf{V}'_{(i),s}, \mathbf{U}'_{(i),u}, \mathbf{V}'_{(i),u}) - \nabla_{\mathbf{U}_s} f_i(\mathbf{U}_s, \mathbf{V}_{(i),s}, \mathbf{U}_{(i),u}, \mathbf{V}_{(i),u}) \right\|_F^2 \\ & + \left\| \nabla_{\mathbf{V}_{(i),s}} f_i(\mathbf{U}'_s, \mathbf{V}'_{(i),s}, \mathbf{U}'_{(i),u}, \mathbf{V}'_{(i),u}) - \nabla_{\mathbf{V}_{(i),s}} f_i(\mathbf{U}_s, \mathbf{V}_{(i),s}, \mathbf{U}_{(i),u}, \mathbf{V}_{(i),u}) \right\|_F^2 \\ & + \left\| \nabla_{\mathbf{U}_{(i),u}} f_i(\mathbf{U}'_s, \mathbf{V}'_{(i),s}, \mathbf{U}'_{(i),u}, \mathbf{V}'_{(i),u}) - \nabla_{\mathbf{U}_{(i),u}} f_i(\mathbf{U}_s, \mathbf{V}_{(i),s}, \mathbf{U}_{(i),u}, \mathbf{V}_{(i),u}) \right\|_F^2 \\ & + \left\| \nabla_{\mathbf{V}_{(i),u}} f_i(\mathbf{U}'_s, \mathbf{V}'_{(i),s}, \mathbf{U}'_{(i),u}, \mathbf{V}'_{(i),u}) - \nabla_{\mathbf{V}_{(i),u}} f_i(\mathbf{U}_s, \mathbf{V}_{(i),s}, \mathbf{U}_{(i),u}, \mathbf{V}_{(i),u}) \right\|_F^2 \\ & \leq L^2 \left(\left\| \mathbf{U}'_s - \mathbf{U}_s \right\|_F^2 + \left\| \mathbf{V}'_{(i),s} - \mathbf{V}_{(i),s} \right\|_F^2 + \left\| \mathbf{U}'_{(i),u} - \mathbf{U}_{(i),u} \right\|_F^2 + \left\| \mathbf{V}'_{(i),u} - \mathbf{V}_{(i),u} \right\|_F^2 \right) \end{aligned}$$

where L is a constant defined as,

$$L = 2\sqrt{(G + 3B_1B_2)^2 + B_1^2B_2^2 + 2B_1^4 + B_2^4 + B_2^4} \quad \square$$

Now, we can proceed to prove the sufficient decrease in inequality. An important equality about the objective f_i is that,

$$\begin{aligned} & f_i(\mathbf{U}_{s,\tau+1}, \mathbf{V}_{(i),s,\tau+1}, \mathbf{U}_{(i),u,\tau+1}, \mathbf{V}_{(i),u,\tau+1}) \\ & = \left\| \mathbf{Y}_{(i)} - \mathbf{U}_{s,\tau+1} \mathbf{V}_{(i),s,\tau+1}^T - \mathbf{U}_{(i),u,\tau+1} \mathbf{V}_{(i),u,\tau+1}^T \right\|_F^2 \\ & = \left\| \mathbf{Y}_{(i)} - \mathbf{U}_{s,\tau+1} \left(\mathbf{V}_{(i),s,\tau+1}^T + \mathbf{V}_{(i),u,\tau+1} \mathbf{U}_{s,\tau+1}^T \mathbf{U}_{(i),u,\tau+1}^{-1} \mathbf{U}_{s,\tau+1} \right)^T \right\|_F^2 \\ & - \left(\mathbf{U}_{(i),u,\tau+1}^T - \mathbf{U}_{s,\tau+1}^T \mathbf{U}_{(i),u,\tau+1} \mathbf{U}_{s,\tau+1}^T \mathbf{U}_{(i),u,\tau+1}^{-1} \mathbf{U}_{s,\tau+1} \right)^T \mathbf{V}_{(i),u,\tau+1}^T \left\|_F^2 \right. \\ & = \left\| \mathbf{Y}_{(i)} - \mathbf{U}_{s,\tau+1} \mathbf{V}_{(i),s,\tau+1}^T - \mathbf{U}_{(i),u,\tau+1} \mathbf{V}_{(i),u,\tau+1}^T \right\|_F^2 \\ & = f_i(\mathbf{U}_{s,\tau+1}, \mathbf{V}_{(i),s,\tau+1}, \mathbf{U}_{(i),u,\tau+1}, \mathbf{V}_{(i),u,\tau+1}) \end{aligned}$$

Therefore, considering the gradient descent step and the inequality (A.2), we have,

$$\begin{aligned} & f_i(\mathbf{U}_{s,\tau+1}, \mathbf{V}_{(i),s,\tau+1}, \mathbf{U}_{(i),u,\tau+1}, \mathbf{V}_{(i),u,\tau+1}) - f_i(\mathbf{U}_{s,\tau}, \mathbf{V}_{(i),s,\tau}, \mathbf{U}_{(i),u,\tau}, \mathbf{V}_{(i),u,\tau}) \\ & = f_i(\mathbf{U}_{s,\tau+1}, \mathbf{V}_{(i),s,\tau+1}, \mathbf{U}_{(i),u,\tau+1}, \mathbf{V}_{(i),u,\tau+1}) - f_i(\mathbf{U}_{s,\tau}, \mathbf{V}_{(i),s,\tau}, \mathbf{U}_{(i),u,\tau}, \mathbf{V}_{(i),u,\tau}) \\ & \leq -\eta \left(\left\langle \nabla_{\mathbf{U}_s} f_i, \nabla_{\mathbf{U}_s} \frac{f}{N} \right\rangle + \left\| \nabla_{\mathbf{U}_{(i),u}} f_i \right\|_F^2 + \left\| \nabla_{\mathbf{V}_{(i),s}} f_i \right\|_F^2 + \left\| \nabla_{\mathbf{V}_{(i),u}} f_i \right\|_F^2 \right) \\ & + \eta^2 \frac{L}{2} \left(\left\| \nabla_{\mathbf{U}_s} \frac{f}{N} \right\|_F^2 + \left\| \nabla_{\mathbf{U}_{(i),u}} f_i \right\|_F^2 + \left\| \nabla_{\mathbf{V}_{(i),s}} f_i \right\|_F^2 + \left\| \nabla_{\mathbf{V}_{(i),u}} f_i \right\|_F^2 \right) \end{aligned}$$

Summing both side for i from 1 to N , we have:

$$\begin{aligned} & f(\mathbf{U}_{s,\tau+1}, \{\mathbf{V}_{(i),s,\tau+1}, \mathbf{U}_{(i),u,\tau+1}, \mathbf{V}_{(i),u,\tau+1}\}) - f(\mathbf{U}_{s,\tau}, \{\mathbf{V}_{(i),s,\tau}, \mathbf{U}_{(i),u,\tau}, \mathbf{V}_{(i),u,\tau}\}) \\ & = \sum_{i=1}^N f_i(\mathbf{U}_{s,\tau+1}, \mathbf{V}_{(i),s,\tau+1}, \mathbf{U}_{(i),u,\tau+1}, \mathbf{V}_{(i),u,\tau+1}) - f_i(\mathbf{U}_{s,\tau}, \mathbf{V}_{(i),s,\tau}, \mathbf{U}_{(i),u,\tau}, \mathbf{V}_{(i),u,\tau}) \\ & \leq -\eta \left(\left\| \frac{1}{\sqrt{N}} \nabla_{\mathbf{U}_s} f \right\|_F^2 + \sum_{i=1}^N \left\| \nabla_{\mathbf{V}_{(i),s}} f_i \right\|_F^2 + \left\| \nabla_{\mathbf{U}_{(i),u}} f_i \right\|_F^2 + \left\| \nabla_{\mathbf{V}_{(i),u}} f_i \right\|_F^2 \right) \\ & + \eta^2 \frac{L}{2} \left(\left\| \frac{1}{\sqrt{N}} \nabla_{\mathbf{U}_s} f \right\|_F^2 + \sum_{i=1}^N \left\| \nabla_{\mathbf{V}_{(i),s}} f_i \right\|_F^2 + \left\| \nabla_{\mathbf{U}_{(i),u}} f_i \right\|_F^2 + \left\| \nabla_{\mathbf{V}_{(i),u}} f_i \right\|_F^2 \right) \end{aligned}$$

Therefore, when $\eta \leq \frac{1}{L}$, we have:

$$\begin{aligned} & f(\mathbf{U}_{s,\tau+1}, \{\mathbf{V}_{(i),s,\tau+1}, \mathbf{U}_{(i),u,\tau+1}, \mathbf{V}_{(i),u,\tau+1}\}) - f(\mathbf{U}_{s,\tau}, \{\mathbf{V}_{(i),s,\tau}, \mathbf{U}_{(i),u,\tau}, \mathbf{V}_{(i),u,\tau}\}) \\ & \leq -\frac{\eta}{2} \left(\left\| \frac{1}{\sqrt{N}} \nabla_{\mathbf{U}_s} f \right\|_F^2 + \sum_{i=1}^N \left\| \nabla_{\mathbf{V}_{(i),s}} f_i \right\|_F^2 + \left\| \nabla_{\mathbf{U}_{(i),u}} f_i \right\|_F^2 + \left\| \nabla_{\mathbf{V}_{(i),u}} f_i \right\|_F^2 \right) \end{aligned}$$

Summing up both sides for τ from 1 to R , we have,

$$\begin{aligned} & \min_{\tau \in \{1, \dots, R\}} \left[\left\| \frac{1}{\sqrt{N}} \nabla_{\mathbf{U}_s} f \right\|_F^2 + \sum_{i=1}^N \left\| \nabla_{\mathbf{V}_{(i),s}} f_i \right\|_F^2 + \left\| \nabla_{\mathbf{U}_{(i),u}} f_i \right\|_F^2 + \left\| \nabla_{\mathbf{V}_{(i),u}} f_i \right\|_F^2 \right] \\ & \leq \frac{2}{R\eta} \left(f(\mathbf{U}_{s,1}, \{\mathbf{V}_{(i),s,1}, \mathbf{U}_{(i),u,1}, \mathbf{V}_{(i),u,1}\}) - f(\mathbf{U}_{s,R+1}, \{\mathbf{V}_{(i),s,R+1}, \mathbf{U}_{(i),u,R+1}, \mathbf{V}_{(i),u,R+1}\}) \right) \end{aligned}$$

Since the objective f is apparently nonnegative, the right-hand side is upper bounded by

$$\begin{aligned} & \min_{\tau \in \{1, \dots, R\}} \left[\left\| \frac{1}{\sqrt{N}} \nabla_{\mathbf{U}_s} f \right\|_F^2 + \sum_{i=1}^N \left\| \nabla_{\mathbf{V}_{(i),s}} f_i \right\|_F^2 + \left\| \nabla_{\mathbf{U}_{(i),u}} f_i \right\|_F^2 + \left\| \nabla_{\mathbf{V}_{(i),u}} f_i \right\|_F^2 \right] \\ & \leq \frac{2}{R\eta} f(\mathbf{U}_{s,1}, \{\mathbf{V}_{(i),s,1}, \mathbf{U}_{(i),u,1}, \mathbf{V}_{(i),u,1}\}) \\ & = O\left(\frac{1}{R}\right) \end{aligned}$$

This completes the proof of Theorem 1.

Appendix B. Connection of (3) to PerPCA

In this section we will show that the formulation in (3) is essentially equivalent to the formulation (5) in [24]. To see this we can first optimize $\mathbf{V}_{(i),u}$'s and $\mathbf{V}_{(i),s}$'s when $\mathbf{U}_s$ and $\mathbf{U}_{(i),u}$'s are given (and orthogonal). The solution is:

$$\begin{cases} \mathbf{V}_{(i),s}^{\star T} = (\mathbf{U}_s^T \mathbf{U}_s)^{-1} \mathbf{U}_s^T \mathbf{Y}_{(i)} \\ \mathbf{V}_{(i),u}^{\star T} = (\mathbf{U}_{(i),u}^T \mathbf{U}_{(i),u})^{-1} \mathbf{U}_{(i),u}^T \mathbf{Y}_{(i)} \end{cases} \quad (\text{B.1})$$

Plugging this into (3), the objective becomes

$$\left\| \mathbf{Y}_{(i)} - \mathbf{U}_s (\mathbf{U}_s^T \mathbf{U}_s)^{-1} \mathbf{U}_s^T \mathbf{Y}_{(i)} - \mathbf{U}_{(i),u} (\mathbf{U}_{(i),u}^T \mathbf{U}_{(i),u})^{-1} \mathbf{U}_{(i),u}^T \mathbf{Y}_{(i)} \right\|_F^2 \quad (\text{B.2})$$

If we define $\tilde{\mathbf{U}}_s = \mathbf{U}_s (\mathbf{U}_s^T \mathbf{U}_s)^{-\frac{1}{2}}$ and $\tilde{\mathbf{U}}_{(i),u} = \mathbf{U}_{(i),u} (\mathbf{U}_{(i),u}^T \mathbf{U}_{(i),u})^{-\frac{1}{2}}$, then $\tilde{\mathbf{U}}_s$ and $\tilde{\mathbf{U}}_{(i),u}$ are orthonormal, and (B.2) transforms to the objective (5) in [24].

It is proved in [24] that the optimal solutions are statistically consistent estimators of true shared and unique subspaces under mild identifiability assumptions. As a result, the column spaces of optimal $\tilde{\mathbf{U}}_s$ and $\tilde{\mathbf{U}}_{(i),u}$'s, thus $\mathbf{U}_s$ and $\mathbf{U}_{(i),u}$'s in (3), are also close to the true shared and unique subspaces. This justifies the formulation (3).

Appendix C. Sensitivity analysis

The stepsize η is an important hyper-parameter in Algorithm 1. Theorem 1 in the main paper provides an upper bound of the stepsize: it

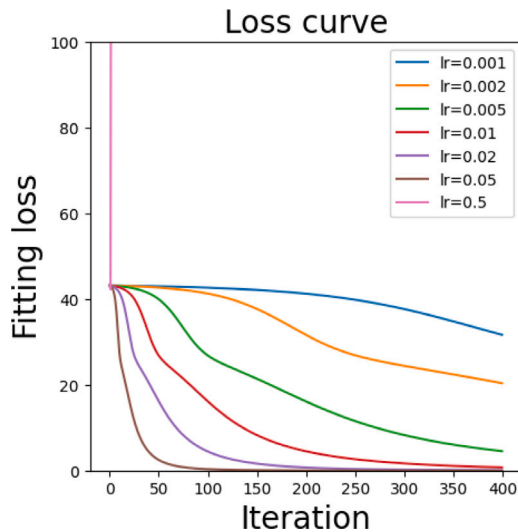


Fig. C.15. Loss curves under different stepsizes (lr in the figure).

should not be larger than the inverse of the maximum operator norm of $\mathbf{Y}_{(i)}$, i.e., $\eta = O\left(\frac{1}{\max_i \|\mathbf{Y}_{(i)}\|}\right)$. Such an upper limit of stepsize is needed to ensure the convergence of Algorithm 1.

We perform a sensitivity analysis on stepsize η to demonstrate its effects on convergence and statistical error. In the experiment, we run Algorithm 1 with different η from the same initialization on the NIST data and collect the fitting error at each iteration. The results are plotted in Fig. C.15.

There are a few interesting observations in Fig. C.15. First, a larger stepsize often corresponds to a faster convergence. However, if the stepsize η is set too large (0.5), the algorithm will diverge. Therefore, one should choose an appropriate stepsize to achieve fast and stable convergence. Second, when the stepsize is not too large, the fitting loss will eventually converge to 0, though at speed dependent on the stepsize. Such a result indicates that the final outcome of the algorithm is not sensitive to the choice of stepsize.

In practice, we recommend practitioners choose the stepsize according to the norm of data $\|\mathbf{Y}_{(i)}\|$. One can also use cross-validation to find the most appropriate stepsize for fast convergence.

References

- [1] Brinksmeier E, Klocke F, Lucca DA, Sölter J, Meyer D. Process signatures – a new approach to solve the inverse surface integrity problem in machining processes. *Proc CIRP* 2014;13:429–34. <http://dx.doi.org/10.1016/j.procir.2014.04.073>, 2nd CIRP Conference on Surface Integrity (CSI).
- [2] Klink A. Process signatures of EDM and ECM processes – overview from part functionality and surface modification point of view. *Proc CIRP* 2016;42:240–5. <http://dx.doi.org/10.1016/j.procir.2016.02.279>.
- [3] Moroni G, Pacella M. An approach based on process signature modeling for roundness evaluation of manufactured items. *J Comput Inf Sci Eng* 2008;8(2):021003. <http://dx.doi.org/10.1115/1.2904923>.
- [4] Fergani O, Berto F, Welo T, Liang SY. Analytical modelling of residual stress in additive manufacturing. *Fatigue Fract Eng Mater Struct* 2017;40(6):971–8. <http://dx.doi.org/10.1111/ffe.12560>.
- [5] Mani M, Lane BM, Donmez MA, Feng SC, Moylan SP. A review on measurement science needs for real-time control of additive manufacturing metal powder bed fusion processes. *Int J Prod Res* 2017;55(5):1400–18. <http://dx.doi.org/10.1080/00207543.2016.1223378>.
- [6] Scime L, Beuth J. Using machine learning to identify in-situ melt pool signatures indicative of flaw formation in a laser powder bed fusion additive manufacturing process. *Addit Manuf* 2019;25:151–65. <http://dx.doi.org/10.1016/j.addma.2018.11.010>.
- [7] Guo S, Guo W, Bian L, Guo YB. A deep-learning-based surrogate model for thermal signature prediction in laser metal deposition. *IEEE Trans Autom Sci Eng* 2023;20(1):482–94. <http://dx.doi.org/10.1109/TASE.2022.3158204>.
- [8] Guo S, Agarwal M, Cooper C, Tian Q, Gao RX, Guo W, et al. Machine learning for metal additive manufacturing: Towards a physics-informed data-driven paradigm. *J Manuf Syst* 2022;62:145–63. <http://dx.doi.org/10.1016/j.jmsy.2021.11.003>.
- [9] Grasso M, Colosimo B. A statistical learning method for image-based monitoring of the plume signature in laser powder bed fusion. *Robot Comput-Integr Manuf* 2019;57:103–15. <http://dx.doi.org/10.1016/j.rcim.2018.11.007>, URL <https://www.sciencedirect.com/science/article/pii/S073658451830139X>.
- [10] Grasso M, Demir A, Previtali B, Colosimo B. In situ monitoring of selective laser melting of zinc powder via infrared imaging of the process plume. *Robot Comput-Integr Manuf* 2018;49:229–39. <http://dx.doi.org/10.1016/j.rcim.2017.07.001>, URL <https://www.sciencedirect.com/science/article/pii/S0736584517300583>.
- [11] Li Z, Liu X, Wen S, He P, Zhong K, Wei Q, et al. In situ 3D monitoring of geometric signatures in the powder-bed-fusion additive manufacturing process via vision sensing methods. *Sensors* 2018;18(4). <http://dx.doi.org/10.3390/s18041180>.
- [12] Fergani O, Berto F, Welo T, Liang SY. Analytical modelling of residual stress in additive manufacturing. *Fatigue Fract Eng Mater Struct* 2017;40(6):971–8. <http://dx.doi.org/10.1111/ffe.12560>.
- [13] Liu C, Le Roux L, Körner C, Tabaste O, Lacan F, Bigot S. Digital twin-enabled collaborative data management for metal additive manufacturing systems. *J Manuf Syst* 2022;62:857–74. <http://dx.doi.org/10.1016/j.jmsy.2020.05.010>.
- [14] Koester LW, Taheri H, Bigelow TA, Bond LJ, Faierson EJ. In-situ acoustic signature monitoring in additive manufacturing processes. *AIP Conf Proc* 2018;1949(1):020006. <http://dx.doi.org/10.1063/1.5031503>.
- [15] Williams RJ, Piglion A, Ronneberg T, Jones C, Pham M-S, Davies CM, et al. In situ thermography for laser powder bed fusion: Effects of layer temperature on porosity, microstructure and mechanical properties. *Addit Manuf* 2019;30:100880. <http://dx.doi.org/10.1016/j.addma.2019.100880>, URL <https://www.sciencedirect.com/science/article/pii/S2214860419303410>.
- [16] Grasso M, Laguzza V, Semeraro Q, Colosimo BM. In-process monitoring of selective laser melting: Spatial detection of defects via image data analysis. *J Manuf Sci Eng* 2016;139(5). <http://dx.doi.org/10.1115/1.4034715>, arXiv:https://asmedigitalcollection.asme.org/manufacturingscience/article-pdf/139/5/051001/6272156/manu_139_05_051001.pdf, 051001.
- [17] Khanzadeh M, Chowdhury S, Tschopp MA, Doude HR, Marufuzzaman M, Bian L. In-situ monitoring of melt pool images for porosity prediction in directed energy deposition processes. *IIEE Trans* 2019;51(5):437–55. <http://dx.doi.org/10.1080/24725854.2017.1417656>.
- [18] Amano H, Yamaguchi Y, Ishimoto T, Nakano T. Reduction of spatter generation using atmospheric gas in laser powder bed fusion of Ti–6Al–4V. *Mater Trans* 2021;62(8):1225–30. <http://dx.doi.org/10.2320/matertrans.MT-M2021059>.
- [19] Mahmoudi M, Ezzat AA, Elwany A. Layerwise anomaly detection in laser powder-bed fusion metal additive manufacturing. *J Manuf Sci Eng* 2019;141(3):031002. <http://dx.doi.org/10.1115/1.4042108>, arXiv:https://asmedigitalcollection.asme.org/manufacturingscience/article-pdf/141/3/031002/6278795/manu_141_03_031002.pdf.
- [20] Guo S, Guo W, Bian L, Guo YB. A deep-learning-based surrogate model for thermal signature prediction in laser metal deposition. *IEEE Trans Autom Sci Eng* 2022;1–13. <http://dx.doi.org/10.1109/TASE.2022.3158204>.
- [21] Zhang Y, Hong GS, Ye D, Zhu K, Fuh JY. Extraction and evaluation of melt pool, plume and spatter information for powder-bed fusion AM process monitoring. *Mater Des* 2018;156:458–69. <http://dx.doi.org/10.1016/j.mates.2018.07.002>, URL <https://www.sciencedirect.com/science/article/pii/S026412751830532X>.
- [22] Repossini G, Laguzza V, Grasso M, Colosimo BM. On the use of spatter signature for in-situ monitoring of laser powder bed fusion. *Addit Manuf* 2017;16:35–48. <http://dx.doi.org/10.1016/j.addma.2017.05.004>, URL <https://www.sciencedirect.com/science/article/pii/S2214860416303402>.
- [23] Yang W, Grasso M, Colosimo BM, Paynabar K. A tensor-based hierarchical process monitoring approach for anomaly detection in additive manufacturing. *Qual Reliab Eng Int*. <http://dx.doi.org/10.1002/qre.3223>. URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/qre.3223>.
- [24] Shi N, Kontar RA. Personalized PCA: Decoupling shared and unique features. *J Mach Learn Res* 2024.
- [25] Shi N, Kontar RA. Personalized federated learning via domain adaptation with an application to distributed 3D printing. *Technometrics* 2023;1–12. <http://dx.doi.org/10.1080/00401706.2022.2157882>.
- [26] Bhatia R. Matrix analysis. vol. 169, Springer Science & Business Media; 2013.
- [27] Lock EF, Hoadley KA, Marron JS, Nobel AB. Joint and individual variation explained (JIVE) for integrated analysis of multiple data types. *Ann Appl Stat* 2013;7(1):523.
- [28] Yang H, Kumara S, Bukkapatnam ST, Tsung F. The internet of things for smart manufacturing: A review. *IIEE Trans* 2019;51(11):1190–216. <http://dx.doi.org/10.1080/24725854.2018.1555383>.
- [29] Hao L, Bian L, Gebrael N, Shi J. Residual life prediction of multistage manufacturing processes with interaction between tool wear and product quality degradation. *IEEE Trans Autom Sci Eng* 2017;14(2):1211–24. <http://dx.doi.org/10.1109/TASE.2015.2513208>.
- [30] Mahadevan S, Nath P, Hu Z. Uncertainty quantification for additive manufacturing process improvement: Recent advances. *ASCE-ASME J Risk and Uncert in Engrg Sys Part B Mech Engrg* 2022;8(1):010801. <http://dx.doi.org/10.1115/1.4053184>.

- [31] Heigel JC, Lane B, Levine L, Phan T, Whiting J. In situ thermography of the metal bridge structures fabricated for the 2018 additive manufacturing benchmark test series (AM-bench 2018). *J Res Natl Inst Stand Technol* 2020;125.
- [32] Frerichs F, Meyer H, Strunk R, Kolkwitz B, Epp J. Development of a process signature for manufacturing processes with thermal loads. *Metall Mater Trans A* 2018;49:3419–29.
- [33] Karpuschewski B, Kinner-Becker T, Klink A, Langenhorst L, Mayer J, Meyer D, et al. Process signatures—knowledge-based approach towards function-oriented manufacturing. *Proc CIRP* 2022;108:624–9. <http://dx.doi.org/10.1016/j.procir.2022.01.001>, 6th CIRP Conference on Surface Integrity.
- [34] Eppinger SD, Huber CD, Pham VH. A methodology for manufacturing process signature analysis. *J Manuf Syst* 1995;14(1):20–34. [http://dx.doi.org/10.1016/0278-6125\(95\)98898-G](http://dx.doi.org/10.1016/0278-6125(95)98898-G).
- [35] Fang T, Jafari M, Bakhadyrov I, Safari A, Danforth S, Langrana N. Online defect detection in layered manufacturing using process signature. In: SMC'98 conference proceedings. 1998 IEEE international conference on systems, man, and cybernetics (cat. no.98CH36218). Vol. 5, 1998, p. 4373–8. <http://dx.doi.org/10.1109/ICSMC.1998.727536>.
- [36] Spears TG, Gold SA. In-process sensing in selective laser melting (SLM) additive manufacturing. *Integr Mater Manuf Innov* 2016;5(1):16–40.
- [37] Kong L, Peng X, Chen Y, Wang P, Xu M. Multi-sensor measurement and data fusion technology for manufacturing process monitoring: a literature review. *Int J Extreme Manuf* 2020;2(2):022001.
- [38] Wang C, Tan X, Tor S, Lim C. Machine learning in additive manufacturing: State-of-the-art and perspectives. *Addit Manuf* 2020;36:101538. <http://dx.doi.org/10.1016/j.addma.2020.101538>.
- [39] Sikandar Iquebal A, Yadav A, Botcha B, Krishna Gorthi R, Bukkapatnam S. Tracking and quantifying spatter characteristics in a laser directed energy deposition process using Kalman filter. *Manuf Lett* 2022;33:692–700. <http://dx.doi.org/10.1016/j.mfglet.2022.07.086>, URL <https://www.sciencedirect.com/science/article/pii/S2213846322001171>, 50th SME North American Manufacturing Research Conference (NAMRC 50, 2022).
- [40] Guo S, Guo WG, Bain L. Hierarchical spatial-temporal modeling and monitoring of melt pool evolution in laser-based additive manufacturing. *IIEE Trans* 2020;52(9):977–97. <http://dx.doi.org/10.1080/24725854.2019.1704465>.
- [41] Lee H, Heogh W, Yang J, Yoon J, Park J, Ji S, et al. Deep learning for in-situ powder stream fault detection in directed energy deposition process. *J Manuf Syst* 2022;62:575–87. <http://dx.doi.org/10.1016/j.jmsy.2022.01.013>, URL <https://www.sciencedirect.com/science/article/pii/S0278612522000127>.
- [42] Tan Z, Fang Q, Li H, Liu S, Zhu W, Yang D. Neural network based image segmentation for spatter extraction during laser-based powder bed fusion processing. *Opt Laser Technol* 2020;130:106347. <http://dx.doi.org/10.1016/j.optlastec.2020.106347>, URL <https://www.sciencedirect.com/science/article/pii/S0030399220309804>.
- [43] Tian Q, Guo S, Melder E, Bian L, Guo WG. Deep learning-based data fusion method for in situ porosity detection in laser-based additive manufacturing. *J Manuf Sci Eng* 2020;143(4). <http://dx.doi.org/10.1115/1.4048957>, 041011.
- [44] Yan H, Grasso M, Paynabar K, Colosimo BM. Real-time detection of clustered events in video-imaging data with applications to additive manufacturing. *IIEE Trans* 2022;54(5):464–80. <http://dx.doi.org/10.1080/24725854.2021.1882013>, arXiv:<https://www.tandfonline.com/doi/pdf/10.1080/24725854.2021.1882013>, URL <https://www.tandfonline.com/doi/abs/10.1080/24725854.2021.1882013>.
- [45] Bugatti M, Colosimo BM. Towards real-time in-situ monitoring of hot-spot defects in L-PBF: a new classification-based method for fast video-imaging data analysis. *J Intell Manuf* 2022;33(1):293–309.
- [46] Gobert C, Reutzel EW, Petrich J, Nassar AR, Phoha S. Application of supervised machine learning for defect detection during metallic powder bed fusion additive manufacturing using high resolution imaging. *Addit Manuf* 2018;21:517–28. <http://dx.doi.org/10.1016/j.addma.2018.04.005>.
- [47] Jin X, Sun Y, Que Z, Wang Y, Chow TW. Anomaly detection and fault prognosis for bearings. *IEEE Trans Instrum Meas* 2016;65(9):2046–54.
- [48] Nakazawa T, Kulkarni DV. Anomaly detection and segmentation for wafer defect patterns using deep convolutional encoder-decoder neural network architectures in semiconductor manufacturing. *IEEE Trans Semicond Manuf* 2019;32(2):250–6. <http://dx.doi.org/10.1109/TSM.2019.2897690>.
- [49] Wen L, Li X, Gao L, Zhang Y. A new convolutional neural network-based data-driven fault diagnosis method. *IEEE Trans Ind Electron* 2018;65(7):5990–8. <http://dx.doi.org/10.1109/TIE.2017.2774777>.
- [50] Mofarreh-Bonab M, Mofarreh-Bonab M. A new technique for image compression using PCA. *Int J Comput Sci Commun Netw* 2012;2(1):111–6.
- [51] Deledalle C-A, Salmon J, Dalalyan AS, et al. Image denoising with patch based PCA: local versus global. In: *BMVC*. Vol. 81, 2011, p. 425–55.
- [52] Han Y, Feng X-C, Baci G. Variational and PCA based natural image segmentation. *Pattern Recognit* 2013;46(7):1971–84.
- [53] Hoyer PO. Non-negative matrix factorization with sparseness constraints. *J Mach Learn Res* 2004;5(9).
- [54] Haeffele B, Young E, Vidal R. Structured low-rank matrix factorization: Optimality, algorithm, and applications to image processing. In: Xing EP, Jbara T, editors. *Proceedings of the 31st international conference on machine learning*. Proceedings of machine learning research, Vol. 32, (2):Beijing, China: PMLR; 2014, p. 2007–15, URL <https://proceedings.mlr.press/v32/haeffele14.html>.
- [55] Zhou G, Cichocki A, Zhang Y, Mandic DP. Group component analysis for multiblock data: Common and individual feature extraction. *IEEE Trans Neural Netw Learn Syst* 2015;27(11):2426–39.
- [56] Gaynanova I, Li G. Structural learning and integrative decomposition of multi-view data. *Biometrics* 2019;75(4):1121–32.
- [57] van der Maaten L, Hinton G. Visualizing data using t-SNE. *J Mach Learn Res* 2008;9(86):2579–605, URL <http://jmlr.org/papers/v9/vandermaaten08a.html>.
- [58] Marshall GJ, Thompson SM, Shamsaei N. Data indicating temperature response of Ti-6Al-4V thin-walled structure during its additive manufacture via Laser Engineered Net Shaping. *Data Brief* 2016;7:697–703. <http://dx.doi.org/10.1016/j.dib.2016.02.084>, URL <https://www.sciencedirect.com/science/article/pii/S2352340916301081>.
- [59] Khanzadeh M, Tian W, Yadollahi A, Doude HR, Tschopp MA, Bian L. Dual process monitoring of metal-based additive manufacturing using tensor decomposition of thermal image streams. *Addit Manuf* 2018;23:443–56.
- [60] Lowe DG. Distinctive image features from scale-invariant keypoints. *Int J Comput Vis* 2004;60:91–110.
- [61] Aharon M, Elad M, Bruckstein A. K-SVD: An algorithm for designing over-complete dictionaries for sparse representation. *IEEE Trans Signal Process* 2006;54(11):4311–22. <http://dx.doi.org/10.1109/TSP.2006.881199>.
- [62] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, p. 770–8.
- [63] Yu F, Koltun V. Multi-scale context aggregation by dilated convolutions. 2015, arXiv preprint [arXiv:1511.07122](https://arxiv.org/abs/1511.07122).
- [64] Zhang Z, Wang X, Jung C. DCSR: Dilated convolutions for single image super-resolution. *IEEE Trans Image Process* 2018;28(4):1625–35.
- [65] Mika S, Schölkopf B, Smola A, Müller K-R, Scholz M, Rätsch G. Kernel PCA and de-noising in feature spaces. *Adv Neural Inf Process Syst* 1998;11.
- [66] Svetlizky D, Das M, Zheng B, Vyatskikh AL, Bose S, Bandyopadhyay A, et al. Directed energy deposition (DED) additive manufacturing: Physical characteristics, defects, challenges and applications. *Mater Today* 2021;49:271–95. <http://dx.doi.org/10.1016/j.mattod.2021.03.020>.
- [67] Wilkinson N, Smith M, Kay R, Harris R. A review of aerosol jet printing—a non-traditional hybrid process for micro-manufacturing. *Int J Adv Manuf Technol* 2019;105(11):4599–619.
- [68] Singh M, Haverinen HM, Dhagat P, Jabbour GE. Inkjet printing—Process and its applications. *Adv Mater* 2010;22(6):673–85. <http://dx.doi.org/10.1002/adma.200901141>, URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/adma.200901141>.
- [69] Nesterov Y, et al. *Lectures on convex optimization*. vol. 137, Springer; 2018.