



An LSTM-autoencoder based online side channel monitoring approach for cyber-physical attack detection in additive manufacturing

Zhangyue Shi¹ · Abdullah Al Mamun² · Chen Kan³ · Wenmeng Tian² · Chenang Liu¹

Received: 28 May 2021 / Accepted: 8 November 2021 / Published online: 17 January 2022
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2021

Abstract

Additive manufacturing (AM) has gained increasing popularity in a large variety of mission-critical fields, such as aerospace, medical, and transportation. The layer-by-layer fabrication scheme of the AM significantly enhances fabrication flexibility, resulting in the expanded vulnerability space of cyber-physical AM systems. This potentially leads to altered AM parts with compromised mechanical properties and functionalities. Furthermore, those internal alterations in the AM builds are very challenging to detect using the traditional geometric dimensioning and tolerancing (GD&T) features. Therefore, how to effectively monitor and accurately detect cyber-physical attacks becomes a critical barrier for the broader adoption of AM technology. To address this issue, this paper proposes a machine learning-driven online side channel monitoring approach for AM process authentication. A data-driven feature extraction approach based on the LSTM-autoencoder is developed to detect the unintended process/product alterations caused by cyber-physical attacks. Both supervised and unsupervised monitoring schemes are implemented based on the extracted features. To validate the effectiveness of the proposed method, real-world case studies were conducted using a fused filament fabrication (FFF) platform equipped with two accelerometers. In the case study, two different types of cyber-physical attacks are implemented to mimic the potential real-world process alterations. Experimental results demonstrate that the proposed method outperforms conventional process monitoring methods, and it can effectively detect part geometry and layer thickness alterations in a real-time manner.

Keywords Additive manufacturing · Cyber-physical security · LSTM-autoencoder · Online attack detection · Process authentication · Side channel

Introduction

Background

The great flexibility of additive manufacturing (AM) has significantly enlarged the product design space, enabling to fabricate complex geometries that are not feasible for conventional manufacturing technologies. Nowadays, AM has been adopted in an increasing number of fields where lightweight structures are highly desirable, such as aerospace, medi-

cal, and transportation (Beyer, 2014; Huang et al., 2015). Furthermore, cyber-enabled platforms are incorporated into AM systems, which allows rapid communication, iteration, and information sharing regarding product design, machine setup, and processing parameters (Lee et al., 2016). This facilitates the cyber-physical AM where critical information can be transferred remotely for manufacturing. However, the information sharing between cyber and physical domains is exposed to the risks of cyber-physical attacks, which poses a tremendous threat in cyber-physical AM systems (Sturm et al., 2017; Wells et al., 2014; Wu et al., 2018).

In general, the files containing important product information in the AM digital thread, such as STL, AMF, 3MF, and G-code files, are at a risk of cyber-physical attacks (Bonnard et al., 2018, 2019a, 2019b; Shi et al., 2021). More critically, the layer-by-layer fashion of AM processes significantly expands the product alteration space. As such, cyber-physical attacks could result in compromised functionality of the fabricated parts without even altering its

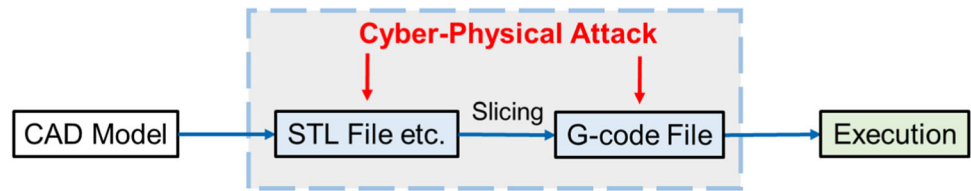
✉ Chenang Liu
chenang.liu@okstate.edu

¹ The School of Industrial Engineering and Management, Oklahoma State University, Stillwater, OK, USA

² Department of Industrial and Systems Engineering, Mississippi State University, Mississippi State, MS, USA

³ Department of Industrial, Manufacturing and Systems Engineering, The University of Texas at Arlington, Arlington, TX, USA

Fig. 1 A demonstration of cyber-physical attacks in AM processes



external geometric dimensioning and tolerancing (GD&T) features. As demonstrated in Fig. 1, a cyber-physical attack can either alter the design (e.g., insert a small void) in the STL/AMF/3MF file, or manipulate the G-code file which is sliced from STL/AMF/3MF file by modifying the process parameters (e.g., increase the nozzle temperature, printing speed, and printing path) and product geometry. The alterations due to cyber-physical attacks may not be easily detected by traditional quality control methods, but it will significantly deteriorate the mechanical properties of AM products, leading to compromised product quality and reliability, or even life-threatening hazards in human-involved operations. Therefore, it is urgently needed to develop *in-situ* process authentication tools for the timely detection of product/process alterations in AM.

Motivation and objectives

With the development of online sensing technologies, sensor-based process monitoring provides new opportunities to achieve effective *in-situ* process authentication, which verifies if the actual process/product is consistent with its design intent (Komolafe et al., 2019; Xiao et al., 2015). With the signals collected from heterogeneous sensors, i.e., side channels, appropriate data analytics methods can be developed by coupling AM process knowledge and online sensor data. However, most recent AM process monitoring methods mainly focus on the detection of process anomalies, such as material under or over flow (Liu et al., 2019), and overheating/lack-of-fusion problems (Khanzadeh et al., 2018, 2019). Product/process alterations induced by cyber-physical attacks, on the other hand, are usually regarded as a normal process, and thus cannot be easily detected by the state-of-the-art *in-situ* anomaly detection methods.

The objective of this study is to extract key process features through heterogeneous sensor fusion to facilitate *in-situ* process alteration detection. Subsequently, various *in-situ* monitoring and shift detection approaches could be implemented based on the extracted process features. In practice, when the attack types of interests are known and the labeled data under attack are available, supervised monitoring approaches, e.g., classification algorithms, can be applied to detect the attack occurrence. However, as discussed, the potential attack space is almost infinite. Thus, it is also common that the type of attack is unknown, and there are no

labeled data under attack for model training. Under this circumstance, unsupervised monitoring approaches are needed.

There are three major technical challenges to realize this objective: (1) some process attributes are technically difficult or expensive to be directly measured by side channels, such as the alteration on infill rate; (2) the process physics of AM are usually highly complex, resulting in challenges to recognize the underlying patterns from the sensor data; and (3) the potential process alteration space is almost infinite in practice, so that the cost of obtaining a comprehensive training dataset is prohibitively high.

To address these challenges, this study developed a feature extraction approach based on LSTM-autoencoder to effectively capture the attack-induced alterations from the online collected sensor signals. As a powerful self-supervised representation learning approach, autoencoder is capable of extracting effective features through identifying a low dimensional representation from the original complex high dimensional data. Moreover, the embedding of LSTM hidden layer further enables the autoencoder to extract more accurate features from the sequential sensor signals to incorporate the temporal information. Using the extracted features, both unsupervised and supervised monitoring approaches are further developed and incorporated for online cyber-physical attack detection. The rest of the paper is organized as follows. A brief review of the related work is provided in Sect. 2. The proposed research methodology is introduced in detail in Sect. 3. To demonstrate the effectiveness of the proposed method, Sect. 4 presents the experimental setup and case study results using a fused filament fabrication (FFF) platform. Finally, the conclusions and future work are discussed in Sect. 5.

Literature review

As discussed in Sect. 1, this study is motivated by the cyber-physical security monitoring for AM processes using side channels. Therefore, this section first introduces relevant existing studies regarding sensor fusion for online anomaly detection (Sect. 2.1), and then followed by a brief review of the existing approaches regarding the side channel monitoring for AM processes (Sect. 2.2) and related studies in the post-process part authentication (Sect. 2.3). Accordingly, the limitations in the current literature are also identified.

Sensor-based anomaly detection approaches

In the literature, sensor-based *in-situ* process monitoring for process/product anomaly detection has been thoroughly investigated. Heterogeneous sensor signals, such as acceleration, temperature, and acoustic emission, are fused for process anomaly detection for various AM processes (Villalobos et al., 2021; Wu et al., 2019). For example, Pittino et al. applied acceleration signals to automatically detect the process anomaly (Pittino et al., 2020). Furthermore, advanced sensing technologies, such as optical camera, infrared imaging, and 3D scan, generate high dimensional data in the format of infrared and optical images or even 3D point cloud, providing rich information for *in-situ* process monitoring (Dastoorian & Wells, 2021; Larsen & Hooper, 2021; Rao et al., 2015; Wang et al., 2021). For example, Liu et al. proposed a close-loop quality control approach based on image data (Liu et al., 2019). Ye et al. proposed an *in-situ* layer-wise monitoring method based on point cloud fusion for additive manufacturing (Ye et al., 2021). In terms of data analytics methodologies, various advanced analytical methods have been tailored for the specific data formats, such as Bayesian Dirichlet process (DP) mixture model (Rao et al., 2015), multifractal analysis (Chen et al., 2020; Imani et al., 2019), deep learning (Li et al., 2021; Ye et al., 2021), and various variants of principal component analysis as well as manifold learning methods (Colosimo & Grasso, 2018; Khanzadeh et al., 2018; Liu et al., 2021). However, most state-of-the-art *in-situ* AM process monitoring methods only focus on process defect detection, making it cumbersome to detect unintentional alterations when the process is still stable.

Side channel monitoring for AM processes

In the cybersecurity area, side channels can be exploited to infer process information (Hospodar et al., 2011). These side channels contain fruitful information and can be used for cyber-physical attack detection, because side channels can capture intrinsic process/product behaviors, and do not necessarily focus on detecting pre-specified alterations (Vincent et al., 2015). In cyber-physical systems, side channels are defined as systems that capture information directly related to the current manufacturing process, which can be used for part and process authentication and verification.

There are several existing solutions to detect cyber-attacks in industrial control systems, for example, neural network-based methods (Kravchik et al., 2021), and the rule-based approaches (Yılmaz & Gönen, 2018). However, these methods do not take the physical domain into consideration, which could not effectively utilize all information in the industrial control systems. Thus, the detection performance may be limited in AM practice. Furthermore, for the rule-based methods, since the potential AM process alteration space is usually

almost infinite, it is also impractical to list all types of cyber-attack in the rule table. During the manufacturing phase, side channel analysis can be categorized into single-sensor side channel analysis and heterogeneous-sensor side channel analysis. In single-sensor side channel analysis, typical side channels exploited include acoustic emission (Belikovetsky et al., 2018; Chhetri et al., 2016; Huang & Kovacevic, 2011; Mativo et al., 2018), thermal related signals (Al Faruque et al., 2016; Slaughter et al., 2017), power consumption (Gatlin et al., 2019; Moore et al., 2017), acceleration (Chhetri et al., 2016; Liu et al., 2020), and image series (Al Mamun et al., 2021; Kwon et al., 2020; Prakash et al., 2020). Recently, electromechanical impedance measurements are used as side channel information for various manufacturing processes (Komolafe et al., 2019; Sturm et al., 2016; Tenney et al., 2019, 2020). The transducer can be attached to or embedded in the AM build during the process to generate *in-situ* part signature for part alteration detection. In heterogeneous-sensor side channel analysis, multiple types of sensors are incorporated to detect cyber-physical attacks to the AM process. For example, Gao et al. (2018) integrated heterogeneous sensor information (i.e., accelerometer, magnetometer, and camera) for real-time printing parameters inference (i.e., infill path, printing speed, layer thickness, and fan speed) to detect any process/part alterations. Chhetri et al. conducted a case study to investigate the integration of acoustic, vibration, power, and magnetic sensors for 3D model reconstruction (Chhetri & Al Faruque, 2017). Machine learning-based supervised monitoring methods (Mahato et al., 2020) and control chart-based unsupervised monitoring methods (Huang et al., 2020) are widely applied in both single-sensor and heterogeneous-sensor based side channel analysis in AM. Furthermore, this strategy has also been adopted by other manufacturing processes. For example, Wu et al. extracted features from acoustic sensor signal and applied random forest classifier to detect the alterations of geometry design and machine parameters in the CNC milling system (Wu et al., 2019). However, the current side channel monitoring methods highly rely on a comprehensive dataset to properly train the model for real-time estimation/detection. When latent information becomes complicated, these methods reviewed above may not be capable of extracting effective features and the monitoring performance may be deteriorated.

Post-process part/process authentication

As another common way to assess the AM product, post-process part/process authentication is also widely applied in practice. Process authentication means the verification of the manufacturing product to its design intent (Komolafe et al., 2019; Xiao et al., 2015), including product geometry, surface roughness, and material properties. During the testing phase in AM processes, multiple part authentica-

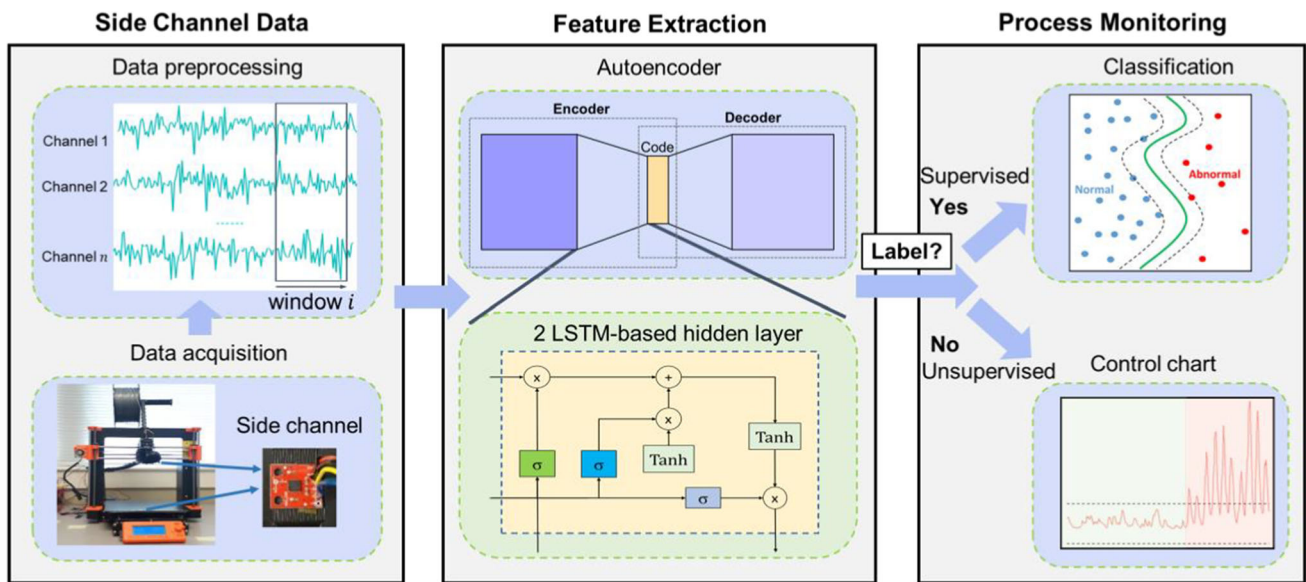


Fig. 2 The overall research framework of the developed online cyber-physical attack detection approach for AM processes

tion methods can be used to verify the part structure and performance, which can be categorized into destructive and nondestructive methods. In destructive testing, the final part will be destroyed during either the testing procedure or the sample preparation procedure. For example, Sturm et al. altered an STL file of a tensile test specimen to reduce the tensile strength which remained undetected until the specimen was destructively tested (Sturm et al., 2017; Zeltmann et al., 2016). In addition, various microstructure characterization can be used to verify the material properties of the AM part (Yampolskiy et al., 2014). In nondestructive evaluation, multiple traditional post-characterization methods, such as ultrasonic testing and x-ray CT testing, can be used (Waller et al., 2014). In addition, recent studies have proposed to use some more convenient approaches for part authentication by applying sensing techniques such as impedance analysis (Komolafe et al., 2019; Tenney et al., 2017, 2019), and chemical taggants (Flank et al., 2017). Unfortunately, compared with *in-situ* monitoring methods, post-process authentication methods cannot effectively utilize the in-process information, particularly, the internal structure fabrication information. Meanwhile, due to its high flexibility requirements for customization, it is usually very costly in terms of time, material, and equipment, or even infeasible, to produce a lot of identical reference parts as training data using AM processes.

Proposed research methodology

As shown in Fig. 2, the overall proposed research methodology in this study consists of three steps: (1) time window-

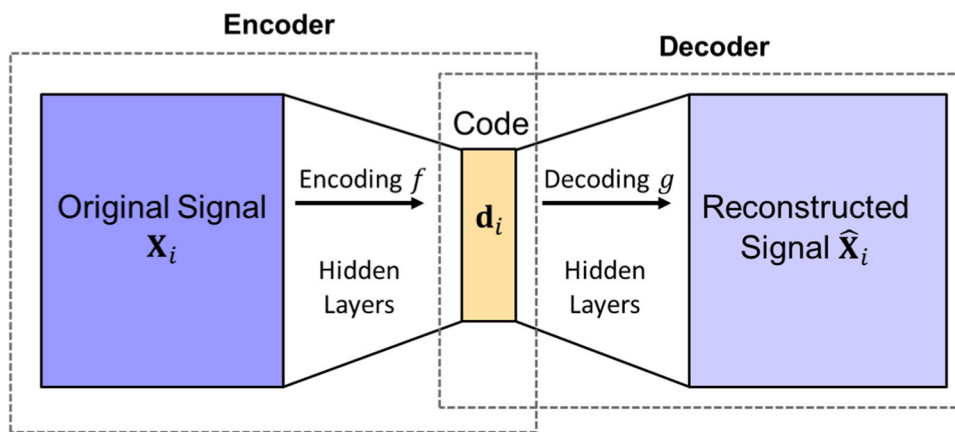
based side channel data representation (Sect. 3.1); (2) data-driven feature extraction method based on the LSTM-autoencoder (Sect. 3.2); and (3) online cyber-physical attack detection framework for AM processes using the extracted features based on supervised and unsupervised method (Sect. 3.3).

Data representation

As discussed in Sect. 1, this study is focused on the cyber-physical attack detection using side channels. In practice, the sensor signals collected from side channels. The raw signals can be represented in the form of a time series $\{\mathbf{x}_t\}_{t=1}^T$, where t is the time index and \mathbf{x}_t is a $p \times 1$ vector and p denotes the dimension of the side channels. For example, in the case study two vibration sensors are mounted on printing bed and extruder with sampling frequency 3 Hz (see details in Sect. 4.1), resulting in a time series with 6 channels, i.e., $p = 6$. Without loss of generality, it is assumed that all sensor channels have the same sampling frequency and are synchronized. In practice, pre-processing techniques such as up/down-sampling approaches (Proakis, 2001) can be potentially used if side channels have different sampling frequency.

The objective is to extract the inherent pattern information of $\{\mathbf{x}_t\}$ and detect the process change caused by cyber-physical attacks in AM. Since the proposed methodology is targeted for *in-situ* application, a natural and cost-effective option is to analyze the sensor data sequentially using a time window. As defined in Eq. (1), the raw signals $\{\mathbf{x}_t\}$ from side channels can be represented as a sequence of overlapping

Fig. 3 A demonstration of the autoencoder architecture



time windows $\{\mathbf{X}_i\}$ (i is the time index of each window) with the dimension $p \times n$,

$$\mathbf{X}_i = \begin{pmatrix} \mathbf{x}^{(i-1) \times (n-v)+1} \\ \mathbf{x}^{(i-1) \times (n-v)+2} \\ \vdots \\ \mathbf{x}^{(i-1) \times (n-v)+n} \end{pmatrix}^T \tag{1}$$

where n is the user-defined window size and v represents the number of overlapping observations between two consecutive windows. In practice, the selection of n and v can be empirically determined based on existing studies (Tootooni et al., 2016). In this study, the window size and overlap are selected by conducting an offline study for the collected training data and identifying an appropriate value. Notably, other approaches, such as the autocorrelation function (ACF) or automutual information (AMI) (Kantz & Schreiber, 2004; Nayfeh & Balachandran, 2008), can be further incorporated as well. Specifically, the first zero crossing ACF and the first minimum of the AMI can be used to determine the window size.

Data-driven feature extraction using LSTM-Autoencoder

To analyze the multi-channel sensor data represented by a series of windows $\{\mathbf{X}_i\}$, this study proposes to incorporate a data-driven feature extraction approach based on the integration of autoencoder and long short-term memory (LSTM), which is capable of effectively capturing the underlying pattern variation of $\{\mathbf{X}_i\}$ when process alterations occur.

Widely applied as a feature extraction approach, autoencoder is a self-supervised neural network that learns accurate low dimensional representation from high dimensional data through an encoding–decoding architecture (Hinton & Salakhutdinov, 2006). The advantages of autoencoder lie in

two main aspects: (1) it is able to capture the inherent non-linear patterns from the data; and (2) it can be conducted in an unsupervised manner without knowing the label information. In general, an autoencoder consists of two parts, namely, an encoder and a decoder. As demonstrated in Fig. 3, an encoder is a part of neural network that embeds the high-dimensional input data \mathbf{X}_i into a low-dimensional representation \mathbf{d}_i in a latent space (i.e., in the code layer). Mathematically, it can be represented as,

$$\mathbf{d}_i = f(\mathbf{X}_i) \tag{2}$$

where $f(\cdot)$ is an encoding function. On the other hand, a decoder is to reconstruct the input from the latent space representation \mathbf{d}_i using a highly symmetric structure with encoder. Similar to the encoder, the decoder can also be represented by a decoding function $g(\cdot)$,

$$\hat{\mathbf{X}}_i = g(\mathbf{d}_i) \tag{3}$$

As such, the autoencoder model can be described by combining Eq. (2) and Eq. (3),

$$\hat{\mathbf{X}}_i = g(f(\mathbf{X}_i)) \tag{4}$$

where the goal is to make the reconstructed signal $\hat{\mathbf{X}}_i$ as close as the original input \mathbf{X}_i , and the reconstruction error is defined as follows,

$$e_i = \|\hat{\mathbf{X}}_i - \mathbf{X}_i\| \tag{5}$$

where $\|\cdot\|$ represents the norm to quantify the error. Consequently, the autoencoder training can be achieved by minimizing the mean square error (MSE) of the reconstruction through all training samples. Moreover, various regularization terms, such as L2 regularization and sparsity regularization, can be incorporated in the loss function (Tschannen et al., 2018). With the defined loss function,

two groups of hyper-parameters need to be pre-defined: (1) dimension of the latent space representation vector, i.e., \mathbf{d}_i , which represents the extent of dimension reduction; and (2) the structure of the hidden layers, e.g., the number of hidden layers and the number of neurons in each layer. Subsequently, the autoencoder can be trained via backpropagation, which follows the same way as the conventional multilayer perceptron. However, such autoencoders that are based on feedforward neural networks that do not fully consider the sequential information in the sensor signals, which may make feature extraction ineffective and result in inaccurate attack detection.

To address this limitation, this study incorporated the LSTM networks (Hochreiter & Schmidhuber, 1997) into the encoder-decoder architecture, namely, LSTM-autoencoder. As a powerful machine learning technique, LSTM is one type of recurrent neural network (RNN) that are capable of learning order dependency in sequential data (Xu & Yoneda, 2019). Due to its excellent performance, LSTM has been widely applied to many types of the sequential data, e.g., video, text, and audio data. In this study, data collected by the side channel during the process are also sequential. Thus, it is natural to integrate the LSTM in the framework of autoencoder (i.e., to fit f and g using LSTM) by considering the inherent temporal relationship within each \mathbf{X}_i .

Compared to the structure of regular neural networks, the hidden layer updates in LSTM are replaced by a special type of neuron, namely, memory cell, which is able to store the useful information over an arbitrary period of time. As shown in Fig. 4a, a memory cell in LSTM consists of three gates, i.e., forget gate, input gate, and output gate. These gates are used to determine the value of C_k and \mathbf{h}_k which represent the state and output vector of the k th memory cell, respectively. Specifically, \mathbf{r}_k represents the forget gate, a layer that usually uses the sigmoid function σ as activation function to determine how long to retain the information. Once the stored data are outdated, \mathbf{r}_k will erase it by resetting to zero. Mathematically, it can be formulated as,

$$\mathbf{r}_k = \sigma(\mathbf{W}_r[\mathbf{h}_{k-1}, \mathbf{x}_k] + \mathbf{b}_r) \quad (6)$$

where \mathbf{h}_{k-1} denotes the output vector of the previous cell and \mathbf{x}_k denotes the input vector of this cell. \mathbf{W}_r and \mathbf{b}_r are the weight matrix and bias vector, respectively. The input gate \mathbf{i}_k is another critical layer in the memory cell to determine which information should be stored. It also uses the sigmoid function as activation function and can be written as,

$$\mathbf{i}_k = \sigma(\mathbf{W}_{in}[\mathbf{h}_{k-1}, \mathbf{x}_k] + \mathbf{b}_{in}) \quad (7)$$

where \mathbf{W}_{in} and \mathbf{b}_{in} denote the weight matrix and bias vector of the input gate, respectively. Besides, the candidate state

of the cell, denoted as \tilde{C}_k , which is the combination of \mathbf{h}_{k-1} and \mathbf{x}_k , can be calculated through,

$$\tilde{C}_k = \tanh(\mathbf{W}_C[\mathbf{h}_{k-1}, \mathbf{x}_k] + \mathbf{b}_C) \quad (8)$$

where \mathbf{W}_C and \mathbf{b}_C denote the weight matrix and bias vector of the input gate, respectively. Meanwhile, the hyperbolic tangent function is applied as activation function. Based on \mathbf{r}_k and \mathbf{i}_k , the state of cell C_k can be updated through

$$C_k = C_{k-1}\mathbf{r}_k + \tilde{C}_k\mathbf{i}_k \quad (9)$$

Then the output vector \mathbf{h}_k of the cell can be determined by the output gate and the cell state, as demonstrated in Eq. (10),

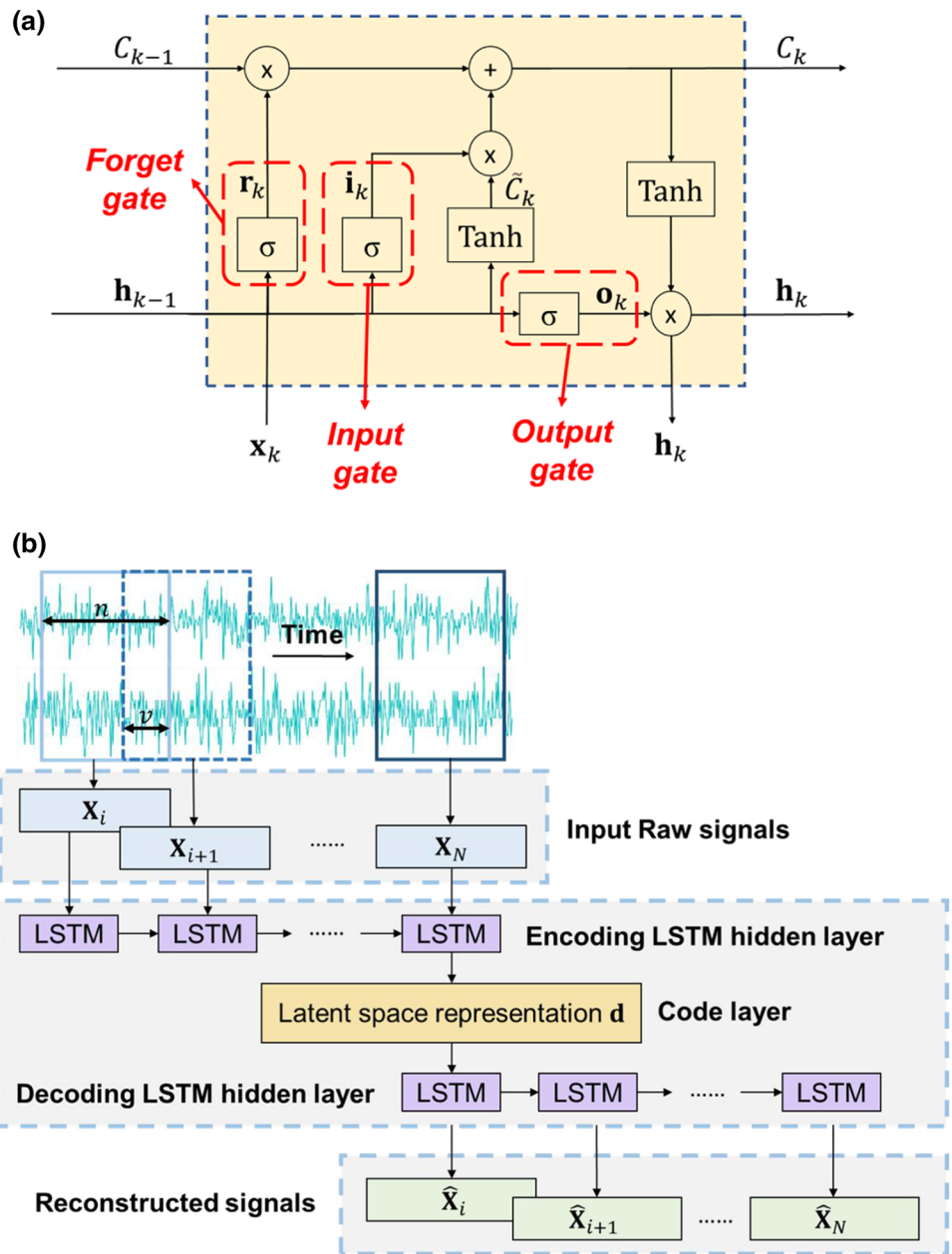
$$\begin{aligned} \mathbf{o}_k &= \sigma(\mathbf{W}_o[\mathbf{h}_{k-1}, \mathbf{x}_k] + \mathbf{b}_o) \\ \mathbf{h}_k &= \mathbf{o}_k \tanh(C_k) \end{aligned} \quad (10)$$

where \mathbf{W}_o and \mathbf{b}_o are the weight matrix and bias vector of the output gate, respectively.

Thereby, as illustrated in Fig. 4b, the LSTM-autoencoder essentially combines the framework of autoencoder and the usage of LSTM networks (Gensler et al., 2016). The LSTM network is embedded in the encoding function (i.e., $f(\cdot)$) and decoding function (i.e., $g(\cdot)$) so that the temporal information in the input sequential data can be effectively considered and utilized. In this study, 2 LSTM hidden layers are incorporated in both encoder and decoder, with 128 and 64 LSTM units, respectively. The dimension of code layer is 60. Since the input data are collected in a window-based format with temporal order, the suggested sequence length in LSTM layer is the same as the window size so that the underlying temporal relationship within each sample could be effectively utilized.

In practice, to apply the LSTM-autoencoder to extract features from side channels for online monitoring, it needs to be trained by using pre-collected data under normal condition (i.e., no attack). Then the newly observed signals from the side channels will be processed using the trained LSTM-autoencoder model and thereby features can be extracted for *in-situ* process authentication. In this study, two groups of features are utilized for both supervised and unsupervised process monitoring. First, the low dimensional latent space representation $\{\mathbf{d}_i\}$, as defined by Eq. (2), is applied to identify the change of the inherent signal patterns between normal and attacked conditions. If unexpected changes occur during process, the variations in the pattern of $\{\mathbf{d}_i\}$ will reflect the pattern change in $\{\mathbf{X}_i\}$. Second, the reconstruction error $\{e_i\}$, which is defined in Eq. (5), is selected for monitoring as well. If the new online side channel data come from the same distribution with the training data, the reconstruction error will also follow the same distribution as that of the training data. Otherwise, unintended process alterations may occur if the level of reconstruction error significantly increases. Based

Fig. 4 a A demonstration of the LSTM unit (memory cell); **b** the architecture of LSTM-autoencoder, which embeds the LSTM network to the encoding and decoding functions



upon these two groups of features, an online cyber-physical attack detection methodology is developed in Sect. 3.3.

Online monitoring for cyber-physical attack detection

In this section, an online monitoring approach for cyber-physical attack detection is further developed using the features $\{d_i\}$ and $\{e_i\}$, which are extracted by the LSTM-autoencoder (see more explanations in Sect. 3.2), as demonstrated in Fig. 5. The overall framework of the proposed online attack detection methodology consists of the process

monitoring from both supervised (Sect. 3.3.1) and unsupervised (Sect. 3.3.2) perspectives.

Supervised process monitoring through classification

If labeled sensor data under attacks are available for training, i.e., possible attack patterns are known, then supervised machine learning, i.e., classification, can be applied to distinguish the attacked status from normal status. In practice, using the fused extracted features $\{(d_i, e_i)\}$ with labels y_i , e.g., normal ($y_i = 0$) or attacked ($y_i = 1$), advanced machine learning-based classifiers can be trained and an appropriate one will be selected for online application after comparison.

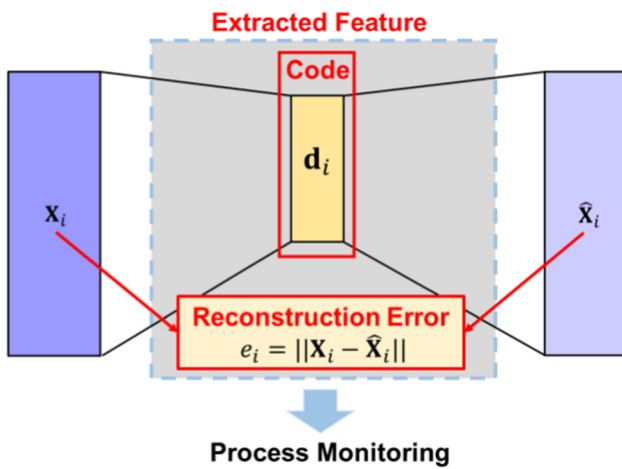


Fig. 5 The features extracted from the developed methodological framework for online cyber-physical attack detection

In this study, through comparison (see details in Sect. 4), a popular ensemble learning method, namely, adaptive boosting (also known as AdaBoost) (Hastie et al., 2009), is applied. AdaBoost uses the initial training data to generate a weak learner, e.g., a decision tree. Then based on the performance of the initial weak learner’s prediction, AdaBoost adjusts the weight of training data for the next round weak learner training. During this process, training samples with low prediction accuracy in the previous round will receive more attention. Eventually, these weak learners are integrated together with different weights to form a strong learner (Feng et al., 2020). After training an AdaBoost model $\text{Ada}(\cdot)$, the labels y_i for

\mathbf{X}_i can be then predicted using the extracted features (\mathbf{d}_i, e_i) , as shown in Eq. (11),

$$\hat{y}_i = \text{Ada}(\mathbf{d}_i, e_i) \tag{11}$$

where \hat{y}_i is the predicted label.

To improve the detection robustness, a voting-based decision-making framework is further implemented based on the prediction results from classification. Specifically, to make the attack detection more accurate at time i , k consecutive samples, $\mathbf{X}_{t-k+1}, \mathbf{X}_{t-k+2}, \dots, \mathbf{X}_t$, will be used together, and the classification result from them with the highest votes will be used to make decision, i.e., normal or abnormal (see Fig. 6). For example, if $k = 5$, then the prediction outcomes from the current sample and 4 consecutive immediate past samples are used, and the alteration will be reported when at least three of these five samples are predicted as abnormal. The number of consecutive samples and sampling frequency are inter-correlated in order to make status predictions in a constant time interval. When the sampling frequency is high, the number of consecutive samples should be large and vice versa. In practice, the sampling frequency of sensor is usually fixed so the only hyperparameter is the number of consecutive samples k . Thus, in this study, an appropriate value of k could be selected through offline tuning in the model training stage. The advantage of supervised process monitoring is that the type of attack can be clearly identified. However, more training data and label information is required in practice to ensure the detection accuracy.

Fig. 6 Supervised process monitoring using AdaBoost with voting-based decision-making framework

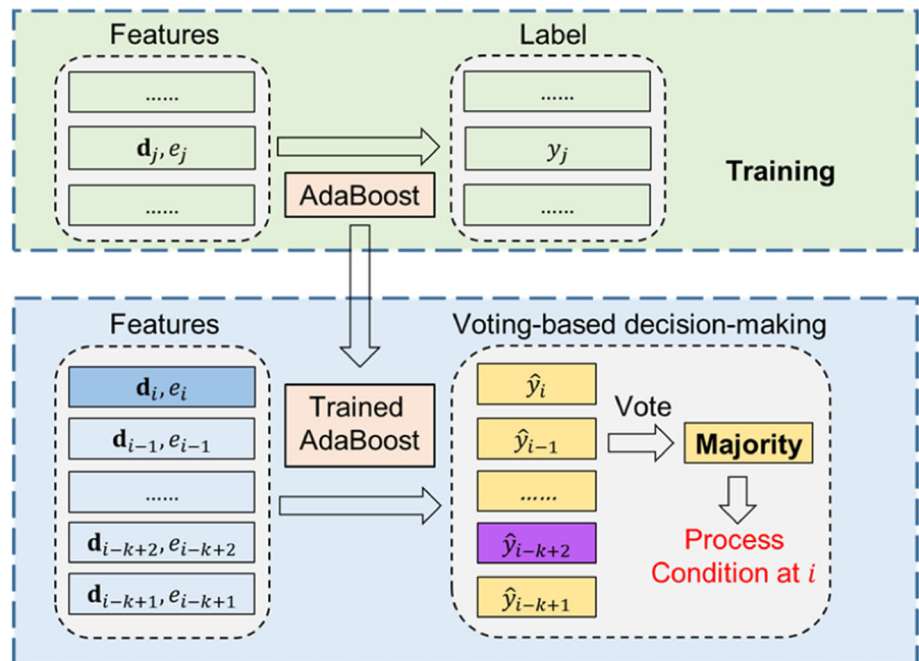
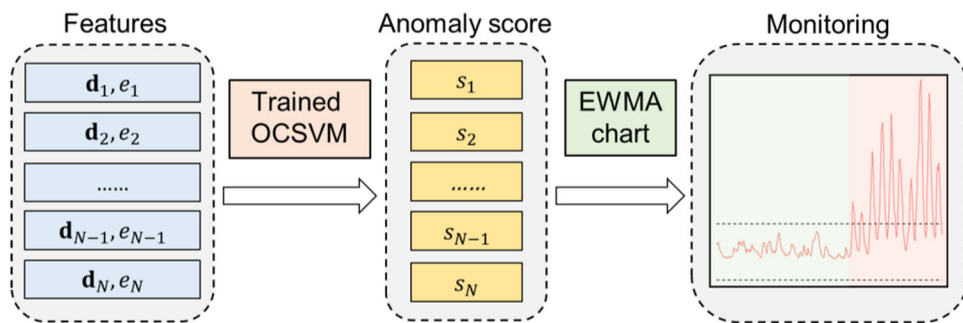


Fig. 7 Unsupervised process monitoring based on EWMA control chart using anomaly score estimated from OCSVM



Unsupervised process monitoring incorporating novelty detection and control chart

On the other hand, as discussed in Sect. 1.2, if the attack information is unknown and only the data without attack, i.e., under normal condition, are available, an unsupervised monitoring approach is then needed for the attack detection. Control chart is an effective and widely applied tool for unsupervised monitoring in engineering practice. However, in this study, directly applying control chart may not work since the collected sensor data are with high dimension and the complex underlying distribution is also unknown. To address this issue, as a popular novelty detection approach, one-class support vector machine (OCSVM) is incorporated, since it is capable of dealing with high-dimensional data and it does not require any distribution assumptions. Thus, this study proposes to integrate these two approaches, in which the OCSVM outputs an anomaly score for every high dimensional sample, and the EWMA control chart is applied to detect the shift of the anomaly scores.

As an extension of SVM, OCSVM is often applied to detect anomalies without over-fitting (Manevitz & Yousef, 2001). OCSVM maps the input (i.e. training data) to a high-dimensional feature space and creates a region within which most of the mapped points fall, while the remaining points lie outside and become outliers (Hoang et al., 2019). Then based on the trained model, the testing data points are mapped to the feature space and labeled accordingly. Furthermore, OCSVM can estimate an anomaly score s_i for each sample, which is calculated by a predefined decision function, denoted by $A_s(\cdot)$,

$$s_i = A_s(\mathbf{d}_i, e_i) \tag{12}$$

The anomaly extent is inversely proportional to the score and it can be identified by the upper bound on the fraction of training error and a lower bound of the fraction of support vectors. For more details of OCSVM, please refer to the Ref (Manevitz & Yousef, 2001). It is worth noting that the performance of OCSVM will be affected by kernel function selection. In this study, different kernel functions, including Radial basis function (RBF), polynomial function, linear

function, and sigmoid function, are examined. The RBF kernel is selected since it can help to achieve the best monitoring performance according to the experiments on the training data.

In practice, if there is no attack, the rate of the detected “alteration” should be at a low level, i.e., the distribution of the calculated anomaly score will not change significantly. However, once an attack occurs, the trained autoencoder will be no longer suitable for the data under attack, hence the anomaly scores calculated from OCSVM follow a different distribution. Based on this hypothesis, this work embedded the OCSVM model into an exponentially weighted moving average (EWMA) control chart, namely, OCSVM-EWMA chart, to achieve effective unsupervised monitoring. As shown in Fig. 7, the developed OCSVM-EWMA chart uses the anomaly score estimated by OCSVM as the quality characteristic and detects its shift through an EWMA chart, which is a well investigated and commonly applied control chart. Mathematically, this could be formulated as

$$z_i = \lambda s_i + (1 - \lambda)z_{i-1} \tag{13}$$

where s_i denotes the score of each point calculated by OCSVM with proper linear transformation, and λ is the parameter of EWMA. There are two major reasons to select EWMA chart in this study for change detection: (1) it is not sensitive for the normal assumption (Huang et al., 2014); and (2) it has strong capability to detect the small variations (Montgomery, 2009). In practice, the parameters of EWMA chart can be determined by balancing false alarms and missed detection rate. Then the control limits and decision rules also can be determined accordingly.

Case study

To validate the effectiveness of the proposed method, this section examines the performance of the proposed methodology based on an actual AM process, fused filament fabrication (FFF), by using the *in-situ* accelerometers as side channels. The experimental setup and data collection are introduced

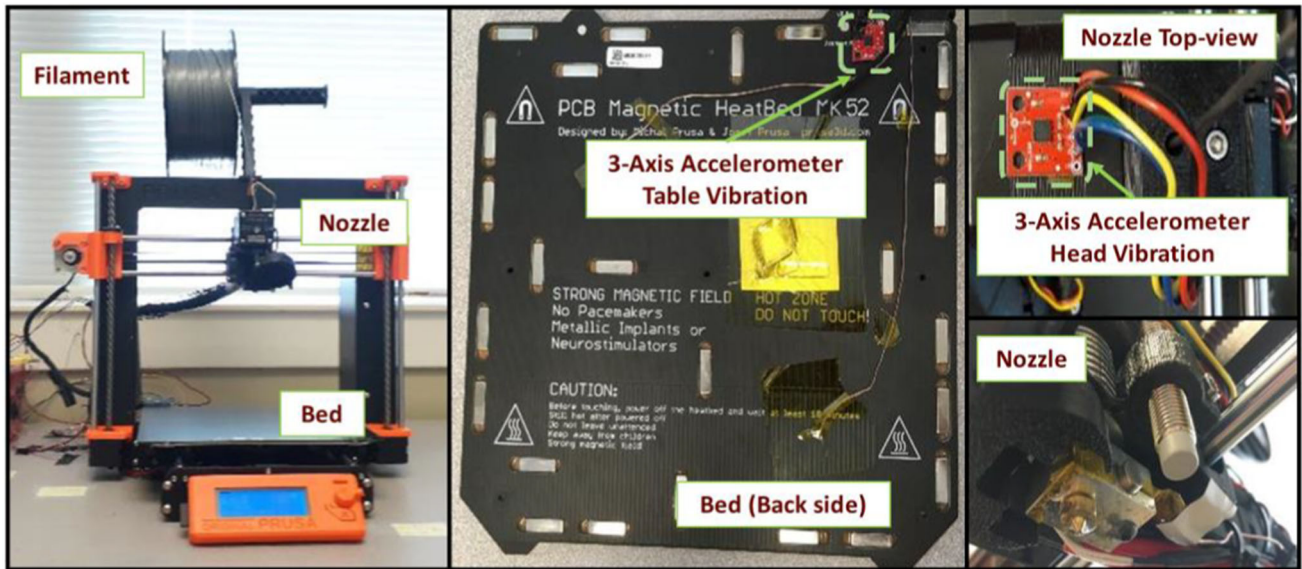


Fig. 8 Demonstration of the experimental setup

in Sect. 4.1, followed by the results summary, interpretation, and discussion in Sect. 4.2.

Experimental setup

In this case study, a desktop FFF-based 3D printer (Prusa i3 MK3S) was used for data collection. Vibration sensors are selected to detect the motion-related alteration during the 3D printing process, which is directly associated with G-code alterations and thus can be used as side channels to detect changes in the G-code (Chhetri et al., 2016). Based on the AM process knowledge, the printing paths specified in the G-code are implemented through the relative motion between the extruder and the printing bed. Therefore, to detect AM process alterations, two vibration sensors (i.e., MEMS accelerometers) are mounted to the extruder and printing bed (see Fig. 8), respectively, which are capable of recording the real-time vibrations of the extruder and the printing bed in the three axes with approximately 3 Hz sampling frequency. ARDUINO MEGA 2560 REV3 microcontroller was used for data acquisition from all the side channels (i.e., the vibration sensors in this study).

In this study, a solid cube (with the edge dimension 2 cm) was used as the nominal design with the process parameters summarized in Table 1. The feedstock material used was polylactic acid (PLA) filament. To validate the performance of the proposed monitoring approach, two cases based on two different potential cyber-physical attacks in AM were investigated in this study, and the detailed design parameters are summarized in Table 2:

Table 1 The design parameters of nominal parts

Design Parameters	Value
Printing Speed	40 mm/s
Layer Thickness	0.3 mm
Nozzle Temperature	215 °C
Bed Temperature	60 °C

- Case 1: The design geometry was attacked, in “STL” stage, to insert a small square-shaped void in the design.
- Case 2: The layer thickness was attacked, in the “slicing” stage, to alter the thickness of specific layers.

These two altered cases are designed to represent two major categories of G-code alterations. Case 1 simulates the scenario where an internal feature (i.e., void) is added to the design, which cannot be detected by traditional quality inspection methods if the internal feature is concealed. On the other hand, Case 2 simulates the scenarios where the extrusion speed and the motion in the z-axis changes. Both cases will lead to compromised mechanical properties of the final product. In addition, it is worth noting that advanced cyber-physical attacks were designed to ensure that the build time of original and altered designs are comparable. Therefore, with no side channel information, the process change cannot be trivially detected by tracking the build time. To conduct the case studies, 6 trials were performed in each case and the sensor data were collected. In each trial, about 4600 sample points (sensor signals) were collected in about 30 min, i.e., the time to print a test part. Afterwards, according to the Sect. 3.1, the raw data were represented in time window-based for-

Table 2 The design parameters of two altered cases

Design Parameters	Case 1	Case 2
Layer Design	Layer # 1~41: solid Layer # 42~67: a square hole inside	Solid
Layer Thickness	0.3 mm	Layer # 1~40: 0.3 mm Layer # 41~56: 0.25 mm Layer # 57~67: 0.35 mm
Printing Speed	40 mm/s	
Nozzle Temperature	215 °C	
Bed Temperature	60 °C	

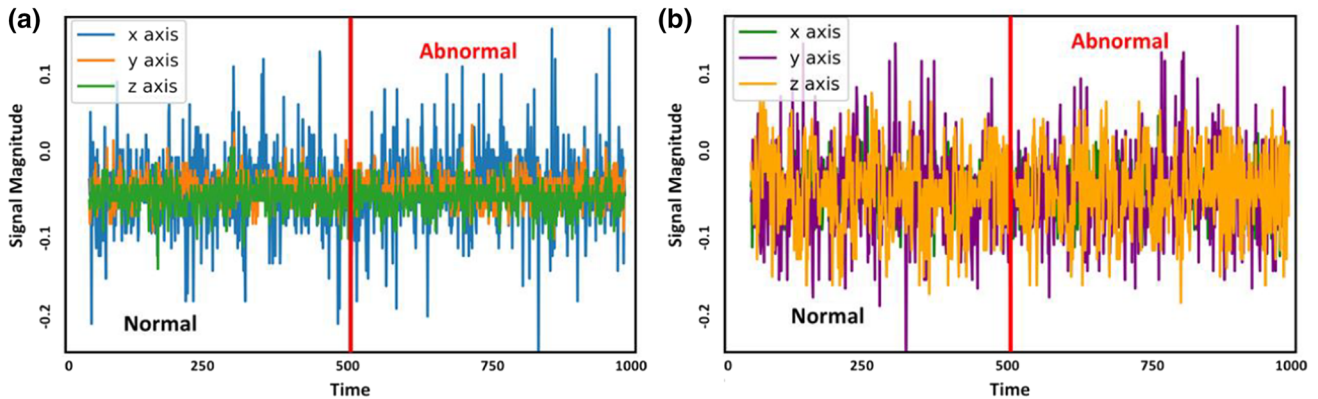
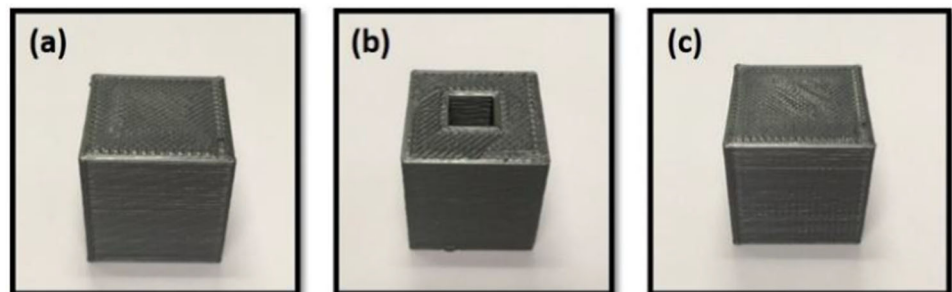


Fig. 9 Normal sample vibration signals collected from **a** extruder, and **b** printing bed

Fig. 10 Sample parts, **a** a nominal part, **b** an attacked part of Case 1, and **c** an attacked part of Case 2



mat with window size $n = 30$ and window overlap $v = 25$. The exemplary raw signals collected from each sensor are shown in Fig. 9 and three sample parts are demonstrated in Fig. 10.

Results and discussion of cyber-physical attack detection performance

As described in Sect. 4.1, the side channel data collected from the two different simulated potential cyber-physical attack cases (i.e., Case 1 and Case 2) were applied to perform the online attack detection analysis. Specifically, a preliminary trial was conducted first as pre-collected data for model training, and then for each case five trials were conducted to test the detection performance. Both supervised and unsupervised online monitoring methods (see details in Sect. 3.3)

were applied to these two cases, and the detection performance is evaluated and discussed as well.

Parameter setup and selected benchmark methods

In the proposed method, several parameters need to be determined in the following four setup steps: (1) time window setup; (2) LSTM-autoencoder setup; (3) classifier (AdaBoost) setup; and (4) OCSVM-EWMA control chart setup. To select appropriate parameters, this study tested different combinations and select the best one based on the attack detection performance. The selected parameters setup for the proposed method is listed in Table 3.

Furthermore, to demonstrate the effectiveness of the LSTM-based feature extraction framework, three state-of-the-art feature extraction methods in the related area, namely,

Table 3 The parameter setup of the proposed method

Step	Parameter	Value
Window setup	Window size (n)	30
	Window overlap (v)	25
LSTM-autoencoder setup	LSTM hidden layer size in encoder	(128,64)
	LSTM hidden layer size in decoder	(64,128)
	Latent space dimension	60
	Transfer function for encoder and decoder	tanh
AdaBoost setup	Number of estimators	400
	Learning rate	0.2
OCSVM-EWMA setup	Kernel function in OCSVM	RBF
	Upper bound on the fraction of training errors and lower bound of the fraction of support vectors	0.1
	λ in EWMA	0.3

Table 4 The parameter setup of the benchmark methods

Methods	Parameter	Value
AdaBoost	Number of estimators	400
	Learning rate	0.2
RF	Number of trees	300
GB	Number of estimators	300
	Learning rate	0.01
SVM	Kernel function	RBF
MEWMA	Smoothing parameter (λ)	0.2
	Confidence level for control limit	0.975
Wavelet + AdaBoost	Wavelet basis	Daubechies 4 (db4)
	Decomposition level	2
	Threshold value used for data compression	0.05
Simple autoencoder + AdaBoost	Latent space dimension	100
	Transfer function for the encoder and decoder	sigmoid

Fast Fourier Transform (FFT) (Mironovova & Bíla, 2015), wavelet transformation (Jiang et al., 2008), and simple autoencoder (Hinton & Salakhutdinov, 2006), were selected as benchmark methods for comparison in this study. Meanwhile, for the supervised monitoring, the popular classification algorithms, including AdaBoost, random forest (RF), gradient boosting (GB), and support vector machine (SVM), were also applied to directly classify the normal status and attacked status for comparison. Regarding the unsupervised monitoring scheme, the multivariate EWMA (MEWMA) chart, which is a very powerful multivariate process monitoring tool, was applied as the benchmark to compare with the developed OCSVM-EWMA chart. The parameter selection used for both benchmark and the proposed methods are summarized in Table 4. Meanwhile, the computation time for online application was evaluated as well. In this study, feature extraction and supervised learning were performed under Python 3.7.3 and TensorFlow 1.13.1 (Abadi et al., 2016). The control chart part was performed under R 4.1.0 with package qcc 2.7 (Scrucca, 2004). The computer CPU

and GPU used in the experiments are Intel Xeon Processor W-2123 (3.6 GHz) and NVIDIA Quadro P2000, respectively.

Supervised monitoring results

The classification results of Case 1 and Case 2 are presented in Tables 5 and 6, respectively. The classification performance is evaluated by the value of precision, recall, and F-score (Powers, 2011). Precision and recall are directly related to the level of type I and type II errors, respectively. F-score is a combination of precision and recall, which can be formulated by Eq. (14),

$$F - \text{score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (14)$$

The results demonstrate that the proposed method achieves the best classification performance in both Case 1 and Case 2. Meanwhile, the computational efficiency of the proposed method is also sufficient for online attack detection since the computational speed (> 2000 Hz) is much faster

Table 5 The comparison of classification results (supervised monitoring)—Case 1

Methods	Precision	Recall	F-score	Computation time (ms)
AdaBoost	0.7424	0.8117	0.7755	0.27
Random Forest	0.7024	0.8428	0.7662	0.28
Gradient Boosting	0.7061	0.8612	0.7759	0.06
SVM	0.6837	0.9535	0.7954	0.27
FFT + AdaBoost	0.8122	0.9820	0.8891	2.5
Wavelet + AdaBoost	0.9100	0.9600	0.9343	2
AE + AdaBoost	0.8947	0.9090	0.9018	0.72
LSTM-AE + AdaBoost	0.9220	0.9570	0.9392	0.72
LSTM-AE (consider error) + AdaBoost (Proposed)	0.9479	0.9460	0.9469	0.81

Table 6 The comparison of classification results (supervised monitoring)—Case 2

Methods	Precision	Recall	F-score	Computation time (ms)
AdaBoost	0.7655	0.8111	0.7876	0.27
Random Forest	0.7322	0.7994	0.7643	0.29
Gradient Boosting	0.7368	0.8231	0.7776	0.06
SVM	0.6667	0.9999	0.8	0.27
FFT + AdaBoost	0.9961	0.7580	0.8609	2.5
Wavelet + AdaBoost	0.8593	0.8730	0.8661	2.1
AE + AdaBoost	0.9371	0.9690	0.9528	0.67
LSTM-AE + AdaBoost	0.9513	0.9770	0.9640	0.8
LSTM-AE (consider error) + AdaBoost (Proposed)	0.9477	0.9790	0.9631	0.85

than the sampling frequency (about 3 Hz) during the printing process. Although using classification algorithm directly for monitoring is much faster, the detection accuracy will be significantly reduced. The three benchmark feature extraction approaches also demonstrate the relatively high detection accuracy, but none of them can achieve competitive accuracy for both Case 1 and Case 2. For example, the wavelet-based method achieved very high accuracy in Case 1, but its F-score in Case 2 is significantly lower than the proposed method. Meanwhile, its computation cost for online monitoring is much larger, which may not be suitable if the sampling frequency or the number of channels increased significantly. Similarly, if the simple autoencoder is applied for feature extraction, the detection accuracy is very competitive in Case 2, however, its performance in Case 1 is not satisfactory. Additionally, the results also show that the consideration of reconstruction error can help to improve the detection accuracy in Case 1.

Unsupervised monitoring results

The results of proposed LSTM-AE + OCSVM-EWMA method under unsupervised monitoring scenario are visualized in Fig. 11, where the dash lines represent the upper control limit (UCL) and lower control limit (LCL), respec-

tively. In addition, the red solid vertical line denotes the onset of cyber-physical attack. For both Case 1 (Fig. 11a) and Case 2 (Fig. 11b), the results demonstrated that the proposed method is able to effectively detect the attack in a short time with a relatively low false alarm rate.

Subsequently, to further compare the proposed approach with the benchmark methods, three metrics are applied to quantify the detection performance, listed below:

- (1) *False alarm rate*: the rate of false alarms under normal condition, i.e., an estimation of type I error α . Lower false alarm rate indicates the method is more robust when the process is under normal condition.
- (2) *Averaged attack response time*: for the five testing trials of each case, when attack occurs, the average number of samples until the control chart first signals, i.e., an estimation of the out-of-control average run length. In general, shorter averaged attack response time implies higher sensitivity to out-of-control samples.
- (3) *True detection rate*: the rate of the true alarms under attacked condition. i.e., an estimation of $1 - \beta$, where β represents the type II error.

Based on the above mentioned three evaluation metrics, the comparison results between the proposed method and

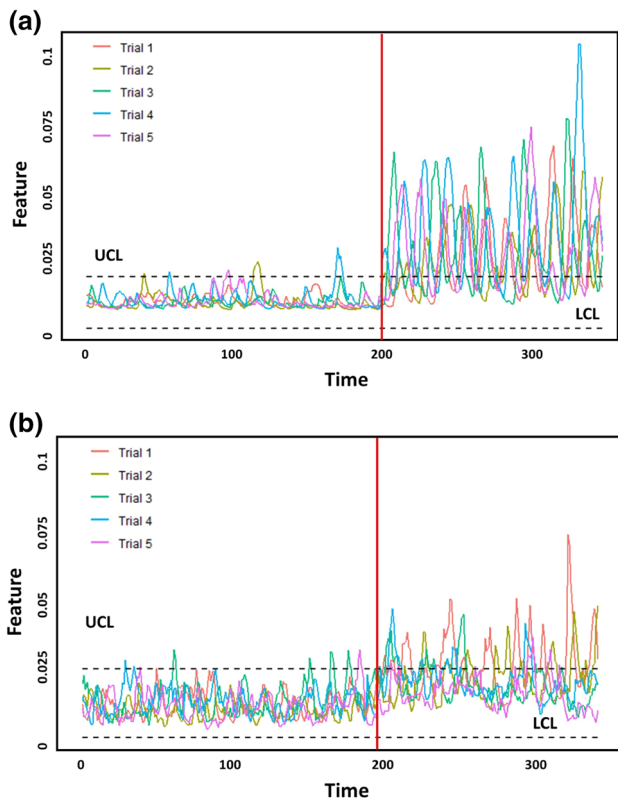


Fig. 11 The online attack detection performance using the proposed LSTM-AE + OCSVM-EWMA method for unsupervised monitoring. **a** Case 1; and **b** Case 2

benchmark methods are presented in Tables 7 and 8. The control limits for each method are determined based on the same training data (i.e., phase I data) used in Sect. 4.2.2, which controls the level of α around 0.025 for the phase I

data. Afterwards, the control limits are applied for the testing data (phase II data), which are also the same with the testing data in Sect. 4.2.2.

Overall, the results demonstrate that the proposed method can achieve a superior performance for both cases than the benchmark methods. Specifically, FFT-based monitoring has very long attack response time and low true detection rate, which is significantly worse than the proposed method. Although the wavelet-based monitoring can achieve competitive attack response time for both cases and similar true detection rate with the proposed method, its false alarm rate is much higher than all other methods and also its true detection rate in Case 2 is low. The simple autoencoder-based approach is also not competitive. For both cases, it has much higher false alarm rates, longer attack response times, and lower true detection rates than the proposed method. In addition, the results based on the direct application of OCSVM-EWMA (without the proposed effective feature extraction) and the LSTM-AE + MEWMA (without the proposed OCSVM-EWMA control chart) based monitoring also show that our proposed feature extraction and control charting framework can indeed reduce both false alarm rate and miss detection rate. Meanwhile, the results also show that the consideration of reconstruction error can help to improve the detection performance significantly in Case 1. Therefore, it can be concluded that the proposed method is very promising to handle the AM process alteration caused by cyber-physical attack.

Conclusions and future work

This paper develops a machine learning-based online monitoring methodology using side channels to detect process

Table 7 The comparison of the unsupervised monitoring results—Case 1

Methods	False alarm rate	Averaged attack response time	True detection rate	Computational time (ms)/unit
FFT + OCSVM-EWMA	0	35.8	0.144	1.06
Wavelet + OCSVM-EWMA	0.198	4.2	0.831	1.588
AE + OCSVM-EWMA	0.170	7.6	0.309	0.494
LSTM-AE + MEWMA	0.051	16	0.149	5.613
OCSVM-EWMA	0.096	5.4	0.716	0.262
LSTM-AE + OCSVM-EWMA (not consider error)	0.063	9.2	0.400	0.426
LSTM-AE + OCSVM-EWMA (Proposed)	0.091	4.6	0.857	0.478

Table 8 The comparison of the unsupervised monitoring results—Case 2

Methods	False alarm rate	Averaged attack response time	True detection rate	Computational time (ms) /unit
FFT + OCSVM-EWMA	0.200	93.8	0.255	0.984
Wavelet + OCSVM-EWMA	0.214	8	0.255	1.482
AE + OCSVM-EWMA	0.017	89.8	0.012	0.556
LSTM-AE + MEWMA	0.099	3.2	0.197	6.116
OCSVM-EWMA	0.155	20.4	0.337	0.303
LSTM-AE + OCSVM-EWMA (not consider Error)	0.024	4	0.423	0.410
LSTM-AE + OCSVM-EWMA (Proposed)	0.033	7.4	0.444	0.432

alterations in AM due to the malicious cyber-physical attacks. A data-driven approach based on the LSTM-autoencoder is developed for feature extraction from the multi-dimensional side channel data. By incorporating an AdaBoost-based classification approach and a developed novelty detection embedded control chart OCSVM-EWMA, the extracted features can be applied for both supervised and unsupervised *in-situ* monitoring, respectively. Two cyber-physical attack cases in FFF are designed to validate the effectiveness of the proposed method. Overall, the conclusions can be summarized as three points: (1) side channels contain useful information to detect process alterations; (2) the proposed LSTM-autoencoder based feature extraction is able to effectively capture the variation induced by process alterations; and (3) the developed attack detection approach using the extract features can detect process alterations in a timely and accurate manner under both supervised and unsupervised scenarios. Furthermore, the proposed method also has the potential to be applied in other additive manufacturing processes as well. To extend it to other AM processes, suitable sensors should be explored and installed due to the different mechanisms among different AM processes. Afterwards, the model setup, such as the neural network architecture also needs to be further optimized in order to achieve a satisfactory performance. In short, it is very promising for the future application of online cyber-physical attack detection in AM.

This work is a preliminary study for side channel-based online process cyber-physical attack detection in AM. Thus, several limitations need to be addressed in the future work. On the one hand, currently only the vibration signals are utilized, so the capability of the proposed method for other types of signals should be further investigated. On the other hand, the design of the test parts is relatively simple in this

study. As the geometry of AM product becomes more complicated, the robustness of the proposed method will be more important and it needs to be investigated as well. Therefore, future directions of this study include the following aspects. First, other types of side channel sensors (e.g., thermocouple, acoustic sensors, etc.) will be explored to improve detection capability for all different types of potential process alterations (such as the temperature and/or speed change). The fusion of heterogeneous sensor data is also expected to provide better performance to the proposed alteration detection framework. Second, when fabricating designs with complicated geometries, the robustness of the proposed method to complex printing paths needs to be further investigated. Last but not the least, from the methodological perspective, the accuracy and efficiency improvement of the feature extraction and monitoring algorithms will be further investigated as well.

References

- Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., & Devin, M. (2016). Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467.
- Al Faruque, M. A., Chhetri, S. R., Canedo, A., & Wan, J. (2016). *Forensics of thermal side-channel in additive manufacturing systems*. University of California.
- Al Mamun, A., Liu, C., Kan, C., & Tian, W. (2021). Real-time process authentication for additive manufacturing processes based on in-situ video analysis. *Procedia Manufacturing*, 53, 697–704.
- Belikovetsky, S., Solewicz, Y. A., Yampolskiy, M., Toh, J., & Elovici, Y. (2018). Digital audio signature for 3D printing integrity. *IEEE Transactions on Information Forensics Security*, 14(5), 1127–1141.

- Beyer, C. (2014). Strategic implications of current trends in additive manufacturing. *Journal of Manufacturing Science and Engineering*, 136(6), 064701.
- Bonnard, R., Hascoët, J.-Y., & Mognol, P. (2019a). Data model for additive manufacturing digital thread: State of the art and perspectives. *International Journal of Computer Integrated Manufacturing*, 32(12), 1170–1191.
- Bonnard, R., Hascoët, J.-Y., Mognol, P., & Stroud, I. (2018). STEP-NC digital thread for additive manufacturing: Data model, implementation and validation. *International Journal of Computer Integrated Manufacturing*, 31(11), 1141–1160.
- Bonnard, R., Hascoët, J.-Y., Mognol, P., Zancul, E., & Alvares, A. J. (2019b). Hierarchical object-oriented model (HOOM) for additive manufacturing digital thread. *Journal of Manufacturing Systems*, 50, 36–52.
- Chen, Y., Li, H., Hou, L., Bu, X., Ye, S., & Chen, D. (2020). Chatter detection for milling using novel p-leader multifractal features. *Journal of Intelligent Manufacturing*, 1–15.
- Chhetri, S. R., & Al Faruque, M. A. (2017). Side channels of cyber-physical systems: Case study in additive manufacturing. *IEEE Design & Test*, 34(4), 18–25.
- Chhetri, S. R., Canedo, A., & Al Faruque, M. A. (2016). Kcad: kinetic cyber-attack detection method for cyber-physical additive manufacturing systems. 2016 IEEE/ACM International Conference on Computer-Aided Design (ICCAD).
- Colosimo, B. M., & Grasso, M. (2018). Spatially weighted PCA for monitoring video image data with application to additive manufacturing. *Journal of Quality Technology*, 50(4), 391–417.
- Dastoorian, R., & Wells, L. J. (2021). A hybrid off-line/on-line quality control approach for real-time monitoring of high-density datasets. *Journal of Intelligent Manufacturing*, 1–14.
- Feng, D.-C., Liu, Z.-T., Wang, X.-D., Chen, Y., Chang, J.-Q., Wei, D.-F., & Jiang, Z.-M. (2020). Machine learning-based compressive strength prediction for concrete: An adaptive boosting approach. *Construction and Building Materials*, 230, 117000.
- Flank, S., Nassar, A. R., Simpson, T. W., Valentine, N., & Elburn, E. (2017). Fast authentication of metal additive manufacturing. *3D Printing and Additive Manufacturing*, 4(3), 143–148.
- Gao, Y., Li, B., Wang, W., Xu, W., Zhou, C., & Jin, Z. (2018). Watching and safeguarding your 3D printer: Online process monitoring against cyber-physical attacks. *Proceedings of the ACM on Interactive, Mobile, Wearable Ubiquitous Technologies*, 2(3), 108.
- Gatlin, J., Belikovetsky, S., Moore, S. B., Solewicz, Y., Elovici, Y., & Yampolskiy, M. (2019). Detecting sabotage attacks in additive manufacturing using actuator power signatures. *IEEE Access*, 7, 133421–133432.
- Gensler, A., Henze, J., Sick, B., & Raabe, N. (2016). Deep Learning for solar power forecasting—An approach using AutoEncoder and LSTM Neural Networks. 2016 IEEE international conference on systems, man, and cybernetics (SMC).
- Hastie, T., Rosset, S., Zhu, J., & Zou, H. (2009). Multi-class adaboost. *Statistics and Its Interface*, 2(3), 349–360.
- Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, 313(5786), 504–507.
- Hoang, T. M., Nguyen, N. M., & Duong, T. Q. (2019). Detection of eavesdropping attack in UAV-aided wireless systems: Unsupervised learning with one-class SVM and k-means clustering. *IEEE Wireless Communications Letters*, 9(2), 139–142.
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780.
- Hospodar, G., Gierlichs, B., De Mulder, E., Verbrauwhe, I., & Vandewalle, J. (2011). Machine learning in side-channel analysis: A first study. *Journal of Cryptographic Engineering*, 1(4), 293.
- Huang, S., Kong, Z., & Huang, W. (2014). High-dimensional process monitoring and change point detection using embedding distributions in reproducing kernel Hilbert space. *IIE Transactions*, 46(10), 999–1016.
- Huang, T., Wang, S., Yang, S., & Dai, W. (2020). Statistical process monitoring in a specified period for the image data of fused deposition modeling parts with consistent layers. *Journal of Intelligent Manufacturing*, 1–16.
- Huang, W., & Kovacevic, R. (2011). A neural network and multiple regression method for the characterization of the depth of weld penetration in laser welding based on acoustic signatures. *Journal of Intelligent Manufacturing*, 22(2), 131–143.
- Huang, Y., Leu, M. C., Mazumder, J., & Donmez, A. (2015). Additive manufacturing: Current state, future potential, gaps and needs, and recommendations. *Journal of Manufacturing Science and Engineering*, 137(1), 014001.
- Imani, F., Yao, B., Chen, R., Rao, P., & Yang, H. (2019). Joint multifractal and lacunarity analysis of image profiles for manufacturing quality control. *Journal of Manufacturing Science Engineering*, 141(4), 044501.
- Jiang, X., Scott, P., & Whitehouse, D. (2008). Wavelets and their applications for surface metrology. *CIRP Annals*, 57(1), 555–558.
- Kantz, H., & Schreiber, T. (2004). *Nonlinear time series analysis* (7th ed.). Cambridge: Cambridge University Press.
- Khanzadeh, M., Chowdhury, S., Tschopp, M. A., Doude, H. R., Maruffzaman, M., & Bian, L. (2019). In-situ monitoring of melt pool images for porosity prediction in directed energy deposition processes. *IIE Transactions*, 51(5), 437–455.
- Khanzadeh, M., Tian, W., Yadollahi, A., Doude, H. R., Tschopp, M. A., & Bian, L. (2018). Dual process monitoring of metal-based additive manufacturing using tensor decomposition of thermal image streams. *Additive Manufacturing*, 23, 443–456.
- Komolafe, T., Tian, W., Purdy, G. T., Albakri, M., Tarazaga, P., & Camelio, J. (2019). Repeatable part authentication using impedance based analysis for side-channel monitoring. *Journal of Manufacturing Systems*, 51, 42–51.
- Kravchik, M., Biggio, B., & Shabtai, A. (2021). Poisoning attacks on cyber attack detectors for industrial control systems. *Proceedings of the 36th Annual ACM Symposium on Applied Computing*.
- Kwon, O., Kim, H. G., Ham, M. J., Kim, W., Kim, G.-H., Cho, J.-H., Kim, N. I., & Kim, K. (2020). A deep neural network for classification of melt-pool images in metal additive manufacturing. *Journal of Intelligent Manufacturing*, 31(2), 375–386.
- Larsen, S., & Hooper, P. A. (2021). Deep semi-supervised learning of dynamics for anomaly detection in laser powder bed fusion. *Journal of Intelligent Manufacturing*, pp. 1–15.
- Lee, J., Bagheri, B., & Jin, C. (2016). Introduction to Cyber Manufacturing. *Manufacturing Letters*, 8, 11–15.
- Li, Y., Shi, Z., Liu, C., Tian, W., Kong, Z., & Williams, C. (2021). Augmented Time Regularized Generative Adversarial Network (ATR-GAN) for Data Augmentation in Online Process Anomaly Detection. *IEEE Transactions on Automation Science and Engineering*.
- Liu, C., Kan, C., & Tian, W. (2020). An online side channel monitoring approach for cyber-physical attack detection of additive manufacturing. *ASME 2020 15th International Manufacturing Science and Engineering Conference*, Cincinnati, Ohio.
- Liu, C., Kong, Z., Babu, S., Joslin, C., & Ferguson, J. (2021). An integrated manifold learning approach for high-dimensional data feature extractions and its applications to online process monitoring of additive manufacturing. *IIE Transactions*, 53(11), 1215–1230.
- Liu, C., Law, A. C. C., Roberson, D., & Kong, Z. J. (2019). Image analysis-based closed loop quality control for additive manufacturing with fused filament fabrication. *Journal of Manufacturing Systems*, 51, 75–86.

- Mahato, V., Obeidi, M. A., Brabazon, D., & Cunningham, P. (2020). Detecting voids in 3D printing using melt pool time series data. *Journal of Intelligent Manufacturing*, pp. 1–8.
- Manevitz, L. M., & Yousef, M. (2001). One-class SVMs for document classification. *Journal of Machine Learning Research*, 2(Dec), 139–154.
- Matavo, T., Fritz, C., & Fidan, I. (2018). Cyber acoustic analysis of additively manufactured objects. *The International Journal of Advanced Manufacturing Technology*, 96(1–4), 581–586.
- Mironovova, M., & Bíla, J. (2015). Fast fourier transform for feature extraction and neural network for classification of electrocardiogram signals. 2015 Fourth International Conference on Future Generation Communication Technology (FGCT).
- Montgomery, D. C. (2009). *Statistical quality control* 7. New York: Wiley.
- Moore, S. B., Gatlin, J., Belikovetsky, S., Yampolskiy, M., King, W. E., & Elovici, Y. (2017). Power consumption-based detection of sabotage attacks in additive manufacturing. arXiv preprint arXiv:01822.
- Nayfeh, A. H., & Balachandran, B. (2008). *Applied nonlinear dynamics: Analytical, computational, and experimental methods*. Wiley.
- Pittino, F., Puggl, M., Moldaschl, T., & Hirschl, C. (2020). Automatic anomaly detection on in-production manufacturing machines using statistical learning methods. *Sensors*, 20(8), 2344.
- Powers, D. M. (2011). Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation.
- Prakash, A., Kumar, S., Mahan, T., Williams, G., McComb, C., Menold, J., & Tucker, C. S. (2020). Detection of system compromise in additive manufacturing using video motion magnification. *Journal of Mechanical Design*, vol 142, no 3.
- Proakis, J. G. (2001). *Digital signal processing: principles algorithms and applications*. Pearson Education India.
- Rao, P. K., Liu, J. P., Roberson, D., Kong, Z. J., & Williams, C. (2015). Online real-time quality monitoring in additive manufacturing processes using heterogeneous sensors. *Journal of Manufacturing Science and Engineering*, vol 137, no 6.
- Scrucca, L. (2004). qcc: An R package for quality control charting and statistical process control. *Dim Pistonrings*, 1(200), 3.
- Shi, Z., Kan, C., Tian, W., & Liu, C. (2021). A blockchain-based G-code protection approach for cyber-physical security in additive manufacturing. *Journal of Computing and Information Science in Engineering*, 21(4), 041007.
- Slaughter, A., Yampolskiy, M., Matthews, M., King, W. E., Guss, G., & Elovici, Y. (2017). How to ensure bad quality in metal additive manufacturing: In-situ infrared thermography from the security perspective. In *Proceedings of the 12th International Conference on Availability, Reliability and Security*.
- Sturm, L., Albakri, M., Williams, C. B., & Tarazaga, P. (2016). In-situ detection of build defects in additive manufacturing via impedance-based monitoring. *27th Annual International Solid Freeform Fabrication Symposium—An Additive Manufacturing Conference*.
- Sturm, L. D., Williams, C. B., Camelio, J. A., White, J., & Parker, R. (2017). Cyber-physical vulnerabilities in additive manufacturing systems: A case study attack on the. STL file with human subjects. *Journal of Manufacturing Systems*, 44, 154–164.
- Tenney, C., Albakri, M., Williams, C., & Tarazaga, P. (2019). NDE of additively manufactured parts via directly bonded and mechanically attached electromechanical impedance sensors. In *Dynamics of Civil Structures, Volume 2* (pp. 263–271). Springer.
- Tenney, C., Albakri, M. I., Kubalak, J., Sturm, L. D., Williams, C. B., & Tarazaga, P. A. (2017). Internal porosity detection in additively manufactured parts via electromechanical impedance measurements. In *ASME 2017 Conference on Smart Materials, Adaptive Structures and Intelligent Systems*.
- Tenney, C. M., Albakri, M. A., Williams, C. B., & Tarazaga, P. A. (2020). Clamping force effects on the performance of mechanically attached piezoelectric transducers for impedance-based NDE. In *Sensors and Instrumentation, Aircraft/Aerospace, Energy Harvesting & Dynamic Environments Testing, Volume 7* (pp. 355–363). Springer.
- Tootooni, M. S., Rao, P. K., Chou, C.-A., & Kong, Z. J. (2016). A spectral graph theoretic approach for monitoring multivariate time series data from complex dynamical processes. *IEEE Transactions on Automation Science and Engineering*, 15(1), 127–144.
- Tschannen, M., Bachem, O., & Lucic, M. (2018). Recent advances in autoencoder-based representation learning. arXiv preprint arXiv:05069.
- Villalobos, K., Suykens, J., & Illarramendi, A. (2021). A flexible alarm prediction system for smart manufacturing scenarios following a forecaster-analyzer approach. *Journal of Intelligent Manufacturing*, 32(5), 1323–1344.
- Vincent, H., Wells, L., Tarazaga, P., & Camelio, J. (2015). Trojan detection and side-channel analyses for cyber-security in cyber-physical manufacturing systems. *Procedia Manufacturing*, 1, 77–85.
- Waller, J., Parker, B., Hodges, K., & Walker, J. (2014). Nondestructive evaluation of additive manufacturing.
- Wang, L., Chen, X., Henkel, D., & Jin, R. (2021). Pyramid ensemble convolutional neural network for virtual computed tomography image prediction in a selective laser melting process. *Journal of Manufacturing Science and Engineering*, 143(12), 121003.
- Wells, L. J., Camelio, J. A., Williams, C. B., & White, J. (2014). Cyber-physical security challenges in manufacturing systems. *Manufacturing Letters*, 2(2), 74–77.
- Wu, D., Ren, A., Zhang, W., Fan, F., Liu, P., Fu, X., & Terpenney, J. J. (2018). Cybersecurity for digital manufacturing. *Journal of Manufacturing Systems*, 48, 3–12.
- Wu, M., Song, Z., & Moon, Y. B. (2019). Detecting cyber-physical attacks in CyberManufacturing systems with machine learning methods. *Journal of Intelligent Manufacturing*, 30(3), 1111–1123.
- Xiao, K., Forte, D., & Tehranipoor, M. M. (2015). Efficient and secure split manufacturing via obfuscated built-in self-authentication. 2015 IEEE International symposium on hardware oriented security and trust (HOST).
- Xu, X., & Yoneda, M. (2019). Multitask air-quality prediction based on LSTM-autoencoder model. *IEEE Transactions on Cybernetics*.
- Yampolskiy, M., Andel, T. R., McDonald, J. T., Glisson, W. B., & Yasinsac, A. (2014). Intellectual property protection in additive layer manufacturing: Requirements for secure outsourcing. In *Proceedings of the 4th Program Protection and Reverse Engineering Workshop*.
- Ye, Z., Liu, C., Tian, W., & Kan, C. (2021). In-situ point cloud fusion for layer-wise monitoring of additive manufacturing. *Journal of Manufacturing Systems*, 61, 210–222.
- Yılmaz, E. N., & Gönen, S. (2018). Attack detection/prevention system against cyber attack in industrial control systems. *Computers & Security*, 77, 94–105.
- Zeltmann, S. E., Gupta, N., Tsoutsos, N. G., Maniatakos, M., Rajendran, J., & Karri, R. (2016). Manufacturing and security challenges in 3D printing. *JOM Journal of the Minerals Metals and Materials Society*, 68(7), 1872–1881.