Explainable deep neural network for in-plain defect detection during additive manufacturing

Deepak Kumar
Aerospace Engineering Department, Embry-Riddle Aeronautical University, Daytona Beach, Florida, USA
Yongxin Liu
Mathematics Department, Embry-Riddle Aeronautical University, Daytona Beach, Florida, USA
Houbing Song
Department of Computer Science, University of Maryland, Baltomore, Maryland, USA, and
Sirish Namilae
Aerospace Engineering Department, Embry-Riddle Aeronautical University, Daytona Beach, Florida, USA

Abstract

Purpose – The purpose of this study is to develop a deep learning framework for additive manufacturing (AM), that can detect different defect types without being trained on specific defect data sets and can be applied for real-time process control.

Design/methodology/approach – This study develops an explainable artificial intelligence (AI) framework, a zero-bias deep neural network (DNN) model for real-time defect detection during the AM process. In this method, the last dense layer of the DNN is replaced by two consecutive parts, a regular dense layer denoted (L1) for dimensional reduction, and a similarity matching layer (L2) for equal weight and non-biased cosine similarity matching. Grayscale images of 3D printed samples acquired during printing were used as the input to the zero-bias DNN.

Findings – This study demonstrates that the approach is capable of successfully detecting multiple types of defects such as cracks, stringing and warping with high accuracy without any prior training on defective data sets, with an accuracy of 99.5%.

Practical implications – Once the model is set up, the computational time for anomaly detection is lower than the speed of image acquisition indicating the potential for real-time process control. It can also be used to minimize manual processing in AI-enabled AM.

Originality/value – To the best of the authors' knowledge, this is the first study to use zero-bias DNN, an explainable AI approach for defect detection in AM.

Keywords Additive manufacturing, Explainable AI, Deep neural network, Anomaly detection, Defect detection

Paper type Research paper

1. Introduction

The exponential increase in additive manufacturing (AM) over the past decade is apparent in a wide variety of applications such as automobile parts (Leal *et al.*, 2017), fine arts (Colorado *et al.*, 2021), aerospace components (Angrish, 2014), artificial medical organs (Melchels *et al.*, 2012). AM has several benefits over conventional manufacturing such as lowering material waste (Mani *et al.*, 2014), streamlining the manufacturing process (Despeisse and Ford, 2015), increasing productivity using cyber manufacturing (Rawat *et al.*, 2017) and reaching higher geometrical complexity (Vaezi *et al.*, 2013).

Extrusion-based AM processes including fused deposition modeling (FDM) and direct ink writing, have been heavily used in meso- and micro-scale fabrication of novel functional materials (Lewis, 2006; Turner *et al.*, 2014; Turner and Gold, 2015). For example, continuous carbon fiber-reinforced composites for high-strength structural components have been fabricated using FDM (Liu *et al.*, 2021a). Significant difficulties remain in understanding and controlling defect development in extrusion-based AM techniques, considering recent improvements in sophisticated process control and application to a wide range of material options. The most frequent flaws in these methods include bubbles and bulges, overfilling and underfilling, scarring, stringing, warping and geometric flaws resulting from an improper head position (Oleff *et al.*, 2021). Most of these flaws can be observed by *in-situ* surface or profile imaging of the filament (bulges, stringing) or the entire layer (overfill, underfill, scars, warping, geometric defects). For real-world applications and next-generation cyber-physical manufacturing systems, *in-situ* characterization approaches are crucial. Several studies have attempted *in-situ* process monitoring

Funding: This work was funded by National Science Foundation (NSF) Advanced Manufacturing Grant (2001038). We thank Martina Jani for assistance with 3D printing setup.

Data availability statement: The processed data required to reproduce these findings are included in the paper. The raw data required to reproduce these findings cannot be shared at this time due to technical or time limitations.

Received 6 May 2023
Revised 27 June 2023
Accepted 29 August 2023

The current issue and full text archive of this journal is available on Emerald Insight at: **https://www.emerald.com/insight/1355-2546.htm**

Rapid Prototyping Journal
30/1 (2024) 49–59
© Emerald Publishing Limited [ISSN 1355-2546]
[DOI 10.1108/RPJ-05-2023-0157]

to enable defect mitigation. For instance, the application of a charged-coupled device camera, high-speed camera and infrared camera for various AM processes have been examined (Chua *et al.*, 2017). Researchers were able to identify the emergence of geometrical defects (Zhang *et al.*, 2019), changes in the conditions of solidification (Tapia and Elwany, 2014) and unanticipated residual stress (Spencer *et al.*, 2021) throughout the printing process using the data from *in-situ* methods.

In-situ methods produce data that can be used for AI training. AI and data-intensive methods provide valuable tools to reveal the complicated correlation between defect formation, material properties and process control in current and future AM processes. Compared with traditional pattern recognition methods, deep learning provides a generic and unified data-processing paradigm. Developers do not need to define domain-specific features because deep neural networks (DNNs) can automatically learn the required feature extractors or convolutional filters during training, DNNs are capable of dealing with highly complicated and heterogeneous input data that integrate information from different sources to make accurate predictions or classifications. Consequently, a few studies have used DNNs and other machine-learning approaches for defect identification during AM (Caggiano *et al.*, 2019; Delli and Chang, 2018; Grasso *et al.*, 2018; Paraskevoudis *et al.*, 2020; Scime and Beuth, 2019; Ye *et al.*, 2018; Zhang *et al.*, 2018). For selective laser melting (SLM), researchers have used supervised learning methods, such as support vector machines (SVM), to detect unstable melting and keyhole defects (Grasso *et al.*, 2018; Scime and Beuth, 2019; Zhang *et al.*, 2018)(Grasso *et al.*, 2018; Scime and Beuth, 2019; Zhang *et al.*, 2018). SVM and k-nearest neighbor methods have been used with optical images obtained during the FDM process to detect defective parts at different checkpoints (Delli and Chang, 2018). Researchers have also used convolutional neural networks and DNN methods with 2D images to classify between five different melting states using plume and spatter images (Ye *et al.*, 2018) and detect process-related nonconformities (Caggiano *et al.*, 2019) for SLM. DNN has also been used to identify stringing in a filament in the fused filament process using the LabelImg tool (Paraskevoudis *et al.*, 2020). Liu *et al.*(2019) proposed an image analysis-based closed-loop feedback method for defect detection and mitigation using textural analysis-based image diagnosis. A machine vision method based on optical imaging is also used for defect detection by comparing the optical imaging and cad design for each layer (Fang *et al.*, 1998). A similar machine vision approach based on signature analysis differences between system-generated images and the original image of each layer is used to detect surface defects for ceramics sensors and actuators (Fang *et al.*, 2003). Cheng and Jafari(2008) proposed an image intensity-based machine vision method to defect surface pattern anomalies using a 3D image processing parametric model and Gaussian function in layered manufacturing.

Despite having high accuracy and accurate prediction, most DNNs models are often trained on specific defects and or stages to classify the input as normal or abnormal. Furthermore, conventional DNNs use dedicated abnormality detection models to identify abnormal inputs. These classifiers lack the capability of detecting inputs from unknown classes. They need a large number of abnormal datasets for each defect category. For example, to detect a defect of a particular type, e.g. crack, the models need to be trained with the data pertaining to the specific defect. It is extremely data-intensive and time-consuming to train a DNN model on all possible defect types in AM.

To address these problems in anomaly detection, we propose to use a unique methodology in deep learning, the zero-bias neural network. In this method, the last dense layer of the DNN is replaced by two consecutive parts, a regular dense layer denoted (L1) for dimensional reduction, and a similarity matching layer (L2) for equal weight and non-biased cosine similarity matching (Liu *et al.*, 2021b). This approach only needs to be trained on normal data and uses feature vectors in latent space to detect possible anomalies in the sample input. An abnormal sample is identified based on its feature vector and Mahalanobis distance from the centroid of the distributions feature vectors of existing classes from trained data. This approach works for all defect types and additionally provides a metric of the abnormality. Advanced AI-enabled AM is only possible with explainable AI that delivers accompanying evidence or reasons for the outcomes and predictions (Phillips *et al.*, 2020). The objective of this study is to formulate the zero-bias DNN approach for an AM problem and use it to demonstrate generic processing defect detection.

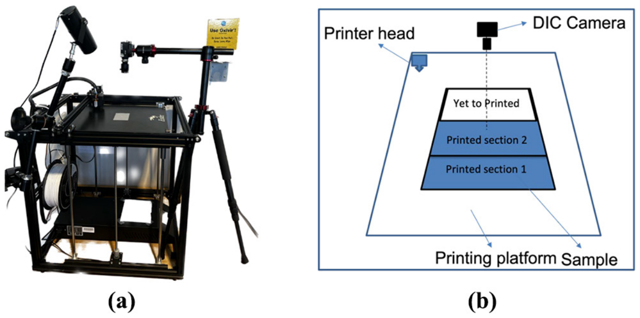
2. Methodology

2.1 Experimental

The Creality Ender 5 Plus 3D printer and a greyscale camera setup were used to generate data sets with and without printing defects. This printer has an over-the-top open printing bed which made the camera setup easy for image capturing. A marble polylactic acid (PLA) filament with a speckle pattern was used as a printing material for all samples. All tests were performed at the recommended temperature of 200°C–230°C for print and 60°C–80°C for the platform. A digital grey-scaled camera is positioned perpendicular to the sample and printing platform, as seen in Figure 1(a). A laser sensor and outside lights were used to maintain the sample heights, locations and quality same for all the tests. The printer's g-code was modified so the extruder could be moved away from the camera frame while the picture was taken. The extruder was programmed to travel away and stop for 5 s after printing a section of any layer. During this time, the camera was configured to capture two images per second. This approach allowed us to analyze the printing process more frequently and generate data sets with normal data and with different types of defects.

For printing, we took into consideration three distinct temperatures of 185°C, 200°C and 215°C and three different

Figure 1 (a) Printer and camera setup, (b) sample layout



(a) (b)

Source: Figure by authors

layer thicknesses (H) of 0.15, 2.0 and 0.25 mm. For the objective of collecting the data set for each setup, each test is carried out three times. The execution of a total of 27 tests resulted in the production of the training data set for this research. Three layers and three sections within each layer, measuring 96 × 60 × H, are printed using Marble PLA for each test, as seen in Figure 1(b). The updated g-code enables the extruder to return to its original position after printing the first segment, at which point the digital camera will automatically snap 10 photographs of each section using the VIC 3D software tool.

At this stage, all the images are saved in a single folder which is later processed and classified among classes and abnormalities. Additional details of image capture are provided in Jani *et al.* (2022).

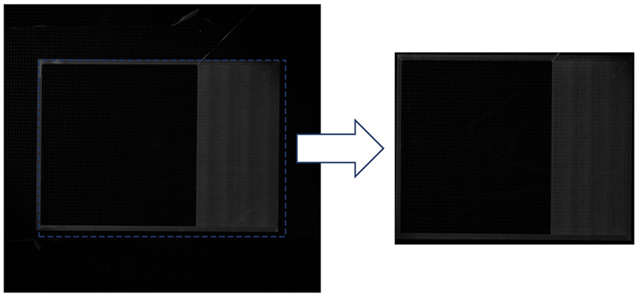
2.2 Data processing

For effective training and evaluation, DNNs require a significant volume of high-quality image data. To accommodate the specific requirements of the model, this data must be correctly pre-processed and structured. This involves normalizing the color and brightness levels, as well as resizing, cropping and other image modifications. Prior to using the data in the model, it should also be categorized and divided into training, validation and test sets. Three main phases were engaged in the data processing for this study: sample cropping, distribution of the sample among classes and construction of an image database for the model input. A rectangular frame of the sample size was used for all data sets to crop the sample out of the entire picture using the im-crop tool in MATLAB as illustrated in Figure 2. To create clear images focusing on the area of interest for each class, we eliminated the surrounding print-bed and other common parts that the camera captured in all images.

For image class distribution we used each section of a sample as a different class. The sample layout in Figure 1(b) shows three sections in each layer. Further, each sample has three layers, therefore the total normal data set was divided into nine different classes. All the images with an abnormality are combined from all sections and saved as class 10.

The normal and abnormal sample data sets were imported as a 4D array with height, breadth, number of channels and image as its columns, respectively. The folder name was used as the class label for various classes, and the data is sorted as normal and abnormal data. The normal data set is stored in 9 different classes as shown in Figure 3, while the abnormal data set is saved as Class 10. Only normal data is used to train and validate the DNN model, which is then used to derive the boundaries and fingerprints of each known class. A total of 70% of normal data is used to train and the remaining 30% is used as test data.

Figure 2 Image captured (left), cropped image used in the analysis (right)



Source: Figure by authors

In this study, we considered multiple types of defects such as cracks, warping, stringing, filament missing. Warping is caused by uneven cooling of the printing sample and bad attachment between the printing bed and sample edge. Due to the warping sample edge curl away from the printing bed as can be seen in Figure 4(a). Cracks are another most common defect type in 3D printing, caused by low glass transition temperatures shown in Figure 4(b). The abnormal samples exhibit visible cracks, warping and stringing. The different types of abnormalities are shown in Figure 4.

2.3 Zero-bias deep neural network model

DNN models can learn complex patterns from large numbers of structured image data sets and can make accurate predictions based on training variables. These models are typically trained using a large data set of labeled images. The first layer of neurons extracts the features from the image and passes them to the next layer. The higher layers of neurons then identify the patterns in the extracted features and gradually learn to recognize the presence of a defect in the image shown in Figure 4. However, these conventional techniques require extensive model training using both normal samples and defective samples. A further drawback of this approach is that one can only identify the defect on which the DNN model was trained.

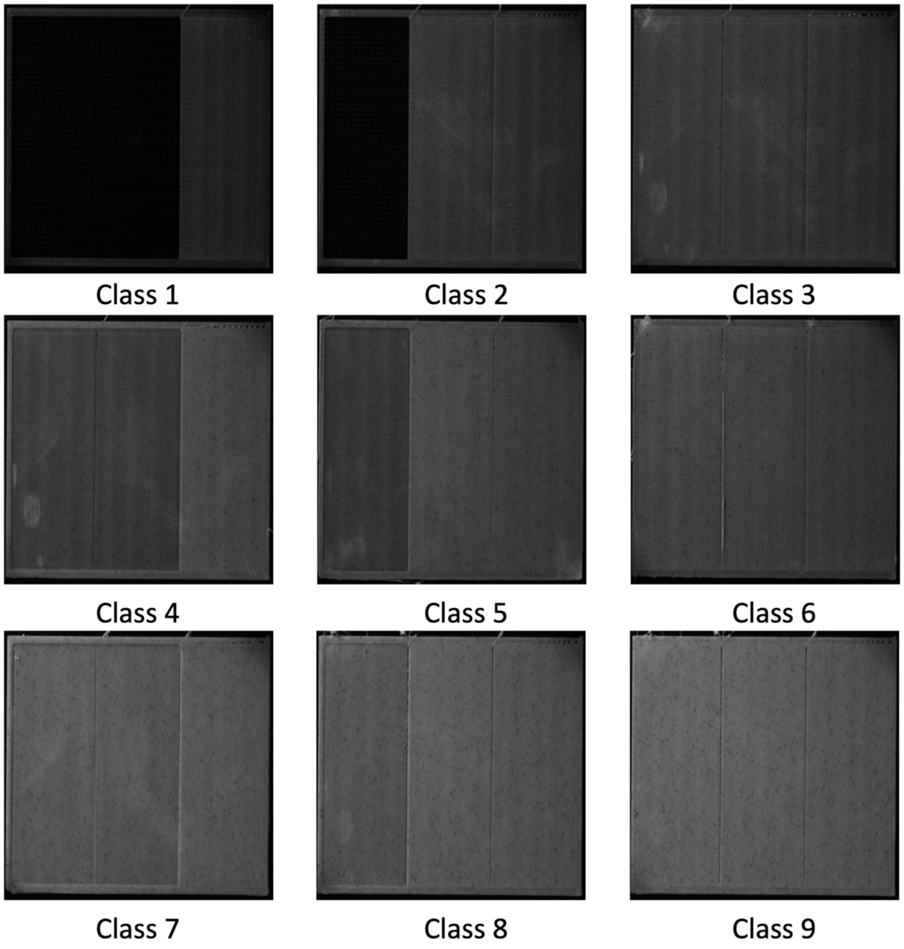
In this study, we used a zero-bias DNN to avoid these drawbacks in training and limitations in the detection of untrained defects. The zero-bias model and its relationship with traditional neural networks are shown schematically in Figure 5(a) and 5(b). A traditional neural network processes its input data through several convolutional layers to extract latent information before using several dense layers (completely connected layers) to eliminate redundant information and produce the final output, as shown in Figure 5(a). In the zero-bias schematically shown in Figure 5(b), the final dense layer is replaced by two subsequent layers, one regular (L1 for dimensional reduction) and another dense layer L2 without bias:

$$\begin{aligned} Last\ dense\ layer(L) &= regular\ denst\ layer(L1) \\ &\quad + Similarity\ Matching\ layer(L2) \end{aligned}$$

In previous work (Liu *et al.*, 2022), it was proved that this transformation could be trained with mathematically equivalent behavior as the original form, and therefore would maintain a similar performance as the untransformed DNN. The newly introduced cosine similarity-matching layer L2 significantly simplifies the decision latent space. As we have removed the bias neurons in the final decision process of the network, for each input, the neural network outputs a vector of cosine similarity scores against templates of all classes in the latent space. This can be used in abnormality detection.

Abnormality detection with zero-bias DNN model involves several steps. First, we trained the zero-bias model using only structured normal data and cross-validated the accuracy using a minibatch size of 32 images. An accuracy threshold was used to end the training process once the model achieved 99% accuracy. Multiple tools, such as cross-validation accuracy, loss

Figure 3 Representative images for the nine classes of normal data



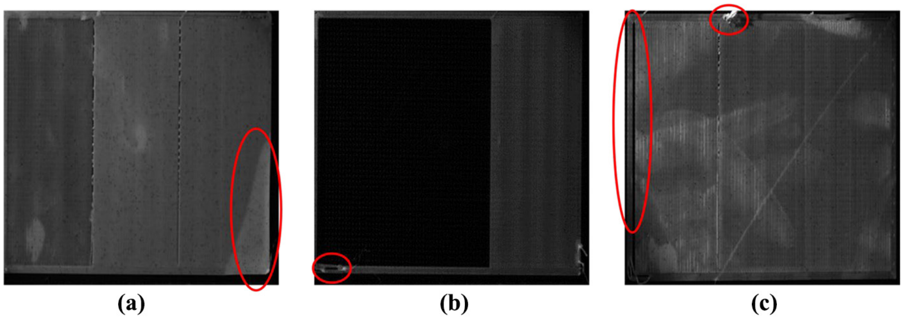
Class 1 Class 2 Class 3

Class 4 Class 5 Class 6

Class 7 Class 8 Class 9

Source: Figure by authors

Figure 4 Representative images of anomalies or defects



(a) (b) (c)

Notes: (a) Warping; (b) cracking; (c) stringing
Source: Figure by authors

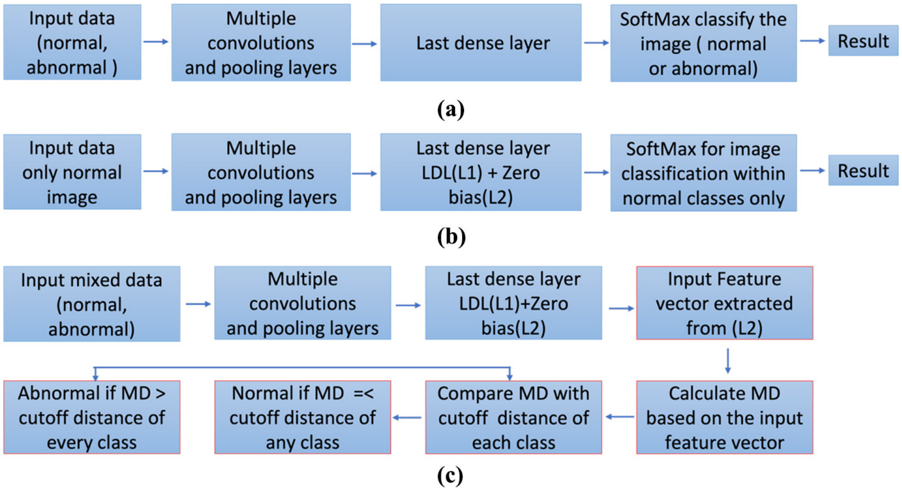
function and confusion matrix were used to validate the model's performance against the test data.

In the next step, we used layer L2 of the zero-bias DNN model to extract the feature vector of each image input before matching it to the fingerprint. For an ordinary DNN model, we know that for known input X_i:

$$Y_1[X_i] = W_1[X_i] + b \tag{1}$$

Where Y_1 is a compressed feature vector before fingerprint matching, W_1 is a matrix to store fingerprints and b is a bias matrix. For the zero-bias model, we divided the last dense layer into two subsequent layers so equation (1) will become:

Figure 5 (a) Generic DNN model framework, (b) schematic of the zero-bias DNN mod, (c) schematic of abnormality detection process using the zero-bias model


Input data (normal, abnormal)
Multiple convolutions and pooling layers
Last dense layer
SoftMax classify the image (normal or abnormal)
Result
(a)
Input data only normal image
Multiple convolutions and pooling layers
Last dense layer LDL(L1) + Zero bias(L2)
SoftMax for image classification within normal classes only
Result
(b)
Input mixed data (normal, abnormal)
Multiple convolutions and pooling layers
Last dense layer LDL(L1)+Zero bias(L2)
Input Feature vector extracted from (L2)
Calculate MD based on the input feature vector
Compare MD with cutoff distance of each class
Normal if MD =< cutoff distance of any class
Abnormal if MD > cutoff distance of every class
(c)

Source: Figure by authors

$$Y_2[X_i] = I(W_2) \,.\, Y_1[X_i] \tag{2}$$

Suppose that $Y_1[X_i]$ is a column vector, and then $I(\cdot)$denotes a function that converts each row vector in W_2 into its unit or direction vector. Therefore, all row vectors after $I(W_2)$ have a magnitude of 1.

Feature vectors of the known dataset are used to estimate the boundaries and fingerprints of each class using equation (2). Once we have the compressed feature vector of all known data sets, the centroid for each class was calculated by using equation (3). For each known data set, a covariance matrix was calculated using equation (4):

$$\mathrm{C}[X_i] = \mathrm{Mean}[Y_2[X_i]] \tag{3}$$

$$\mathrm{Cov}[X_i] = [Y_2[X_i], Y_2[X_i]] \tag{4}$$

Mahalanobis distance (MD) is a measure of the multivariate separation of a point from a distribution. It is commonly used to determine the degree of similarity based on the covariance of the data points (De Maesschalck *et al.*, 2000):

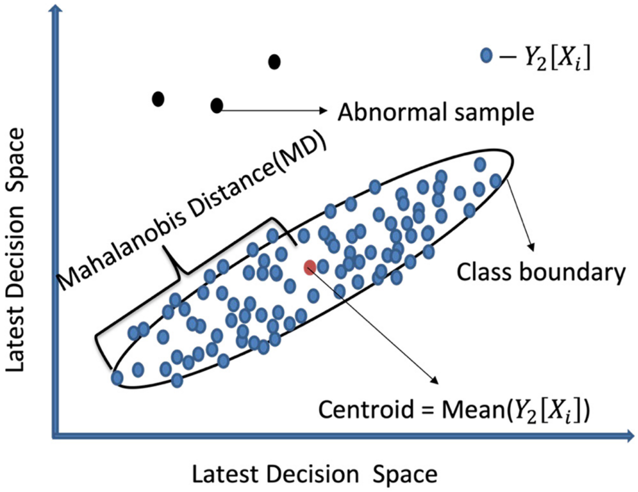
$$\mathrm{MD} = \surd((Y_2[X_i] - C)^T.Cov^{-1}.(Y_2[X_i] - C)) \tag{5}$$

A cutoff measure is formulated based on the MD between the centroid of a given class and the feature vector of the farthest normal input sample using equation (5), (6):

$$\mathrm{Cutoff\ distance(COD)} =$$
$$\mathrm{Max}\left(\surd\left((Y_2[X_i] - C)^T \,.\, Cov^{-1} \,.\, (Y_2[X_i] - C)\right)\right) \tag{6}$$

As shown in Figure 6, an abnormal sample is identified using a cutoff distance between the centroid of any class and the compressed feature vector of an input data set. Any input that is mapped outside the cutoff distance of all classes is considered abnormal:

Figure 6 Abnormality detection based on Mahalanobis distance



Source: Figure by authors

$$\mathrm{Anomaly} = \begin{cases} 0 \text{ if } \mathrm{Max}\left(\surd\left((Y_2[X_i] - C)^T.Cov^{-1}.(Y_2[X_i] - C)\right)\right) \le \mathrm{COD} \\ 1 \text{ if } \mathrm{Max}\left(\surd\left((Y_2[X_i] - C)^T.Cov^{-1}.(Y_2[X_i] - C)\right)\right) \ge \mathrm{COD} \end{cases}$$

It is uncommon for an abnormal image to have a feature vector identical to the known class data. Once we had decided on the sample category, we predicted the abnormality class using the trained DNN model illustrated in Figure 5(c). The trained model was able to assign each image to its class because all abnormal examples were drawn from the known classes. we used this prediction to label the abnormality class.

If we consider an input sample that has an abnormality, such as cracks, it is mixed with normal datasets. To detect an abnormal sample, we first processed the data through a trained DNN model [Figure 5(b)] and extracted the feature vector according to the process shown in Figure 5(c) using blue boxes.

Once we have the feature vector of the individual data input, we calculate the MD and compare it with the cut-off distance of each class. The feature vector of the abnormal sample will have a greater MD from the centroid of the classes than the cutoff distance of all known classes. In Figure 5(c), the additional procedure for abnormality identification is shown using red boxes.

3. Results

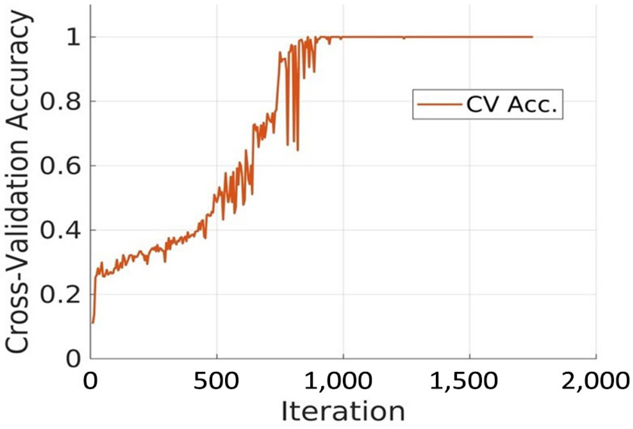
Abnormality detection with a zero-bias DNN model mainly consists of two steps as described above. The first step is the model training to its optimum accuracy using only normal datasets and extracting the feature vector from the trained model's last dense layer (L2). For training, we used 1,870 normal image datasets with the model characteristics shown in Table 1.

We used a K-fold cross-validation approach with an accuracy threshold value for accurate measurements to achieve the highest accuracy. The accuracy of each training iteration is plotted using a mini-batch of 32 samples. The small batch size at the beginning of the training caused minor fluctuations in accuracy. Figure 7 shows that the model achieves a cross-validation accuracy of 99.99%. We used the loss function to adjust the weights and parameters of the model to reduce the difference between the model output and the experimental data. Figure 8 shows how the loss function decreased as the accuracy increased. Once the model exceeded the accuracy threshold, a confusion matrix was used for visual validation. As shown in Figure 9, the model accurately classified 100% of the normal data input into corresponding classes.

Table 1 Zero-bias DNN model training parameters

Model characteristics	Value
Input size	600 × 600 × 1
Layers	13
Number of epochs	25
Minibatch size	32
Momentums – SGD	0.9
Initial learning rate	0.01
Decay	0.01

Figure 7 Cross-validation accuracy vs iterations



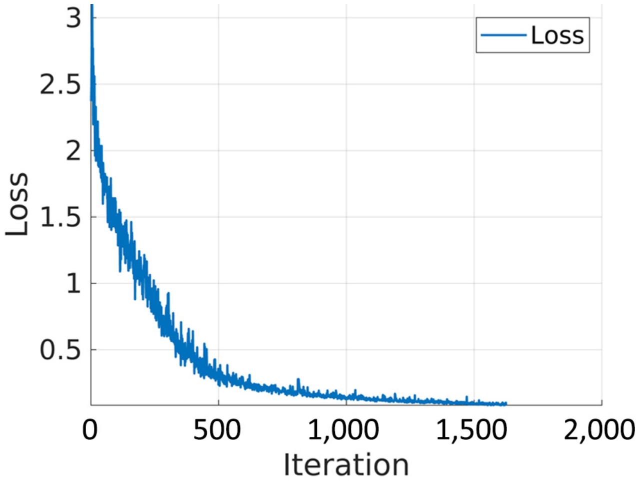
Source: Figure by authors

The trained zero-bias DNN model is then converted into an abnormality detector schematically described in Figure 5(c). In this second step, we extracted the feature vectors of each known data set and the boundary of each known class.

The cutoff distance for each class is calculated based on the MD between the centroid of the class and its furthermost feature vector. The cutoff metric for the nine classes considered here is tabulated in Table 2. All input data sets were then compared with respect to the cutoff distance, and abnormality detection was performed based on the value of the MD of the input.

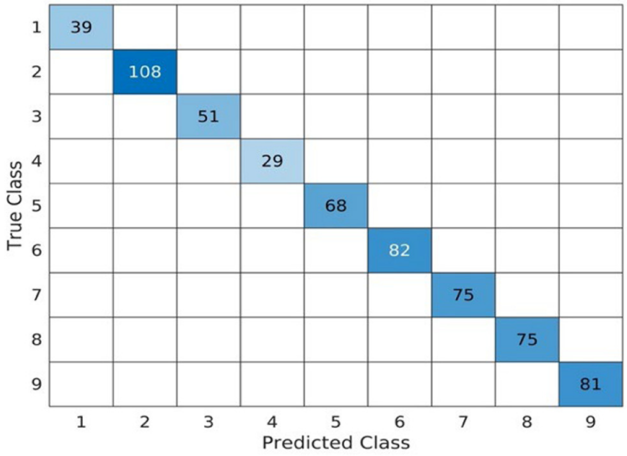
As shown in Table 2, for all normal sample inputs, the MD between the feature vector and centroid of the corresponding class was less than the cutoff distance. Consider the normal image shown in Figure 10(a). The corresponding MD value for class 1 is 3.4e-6 (third-row second column), and this is less than

Figure 8 Loss function vs iterations



Source: Figure by authors

Figure 9 Confusion matrix for the test data for the nine classes



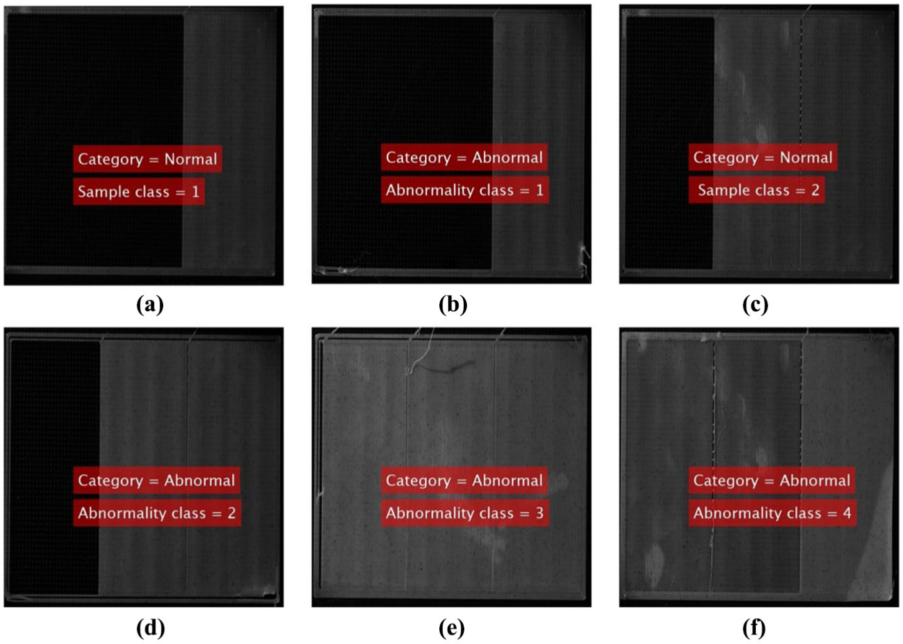
Source: Figure by authors

Table 2 Comparison of cutoff metrics for different classes and Mahalanobis distance for selected normal and abnormal samples

Class	1	2	3	4	5	6	7	8	9
Cutoff metric	10.08	13.90	13.02	18.83	26.59	25.31	24.82	23.14	22.37
Class 1 (*e06) Figure 10(a)	3.4e-6	293.64	1.44	101.71	1.40	63.77	31.36	103	31.3
Class1 cracks(*e06) fig-10(b)	0.0055	1.60	5.62	2.06	0.29	1.77	2.41	9.59	1.38
Class 2 (*e06) Figure 10(c)	328.28	11.6e-6	9.11	4.00	0.06	2.41	4.21	3.13	1.53
Class 2 filament Missing(*e06) Figure 10(d)	511.31	0.01	7.08	3.16	0.02	2.01	5.11	3.25	2.23
Class3 sample(*e06)	544.49	3.45	5.6e-6	2.74	0.14	1.08	4.41	3.05	1.71
Class3 stringing(*e06) Figure 10(e)	599.65	3.24	0.26	1.65	0.47	0.49	4.63	5.05	2.94
Class4 sample(*e06)	273.23	3.45	5.19	6.3e-6	0.06	2.12	1.31	5.68	4.37
Class4 warping(*e06) Figure 10(f)	120.37	3.18	4.36	0.14	0.21	3.26	1.13	5.29	2.39

Source: Table by authors

Figure 10 (a) Normal sample from Class 1, (b) Abnormal sample from Class 1 with a crack, (c) Normal sample from Class 2, (d) Abnormal sample from Class 2 with a missing filament, (e) Abnormal sample from Class 3 with stringing, (f) Abnormal sample from Class 4 with warping



Source: Figure by authors

the corresponding cutoff metric for class 1. The MD values for the same image corresponding to the other classes are listed in the third row, and these values are higher than the cutoff metrics for the remaining eight classes. Therefore, the image in Figure 10(a) is classified as a Class 1 normal image.

Consider a defective sample with a crack as shown in Figure 10(b). For this sample, the MD was greater than the cutoff distance for all classes. However, the MD is lowest for Class 1, therefore this image is an abnormal (defective) Class 1 image. Similarly, the warping defect shown in Figure 10(f) has a MD value greater than the cutoff for all classes, but the value corresponding to Class 4 is the lowest, therefore this image is classified as an abnormality in Class 4.

Note that the model is trained using a normal data set only. Despite this, the model was able to detect multiple types of defects such as cracks [Figure 10(b)], missing filaments [Figure 10(d)], stringing [Figure 10(e)] and warping [Figure 10(f)]. Furthermore, the model identifies the parent class for each of these defects.

We considered weights in the zero-bias similarity layer as class fingerprints. This allowed us to graphically prove that the feature vector of the known data set is closer to its class fingerprints and that input sample with abnormality have their feature vector away from all known class fingerprints. The Voronoi diagram allows a graphical representation of the decision boundary between different classes of data points in the latent decision space. Each region represents a class and points in the same region belonged to the same class. In this study, we used two different types of data batches to visualize the normal and abnormal data using Voronoi diagrams. The first batch contained only the normal test data from the model training. The extracted feature vectors were further processed using a nonlinear dimensional reduction method called *t*-distributed stochastic neighbor embedding

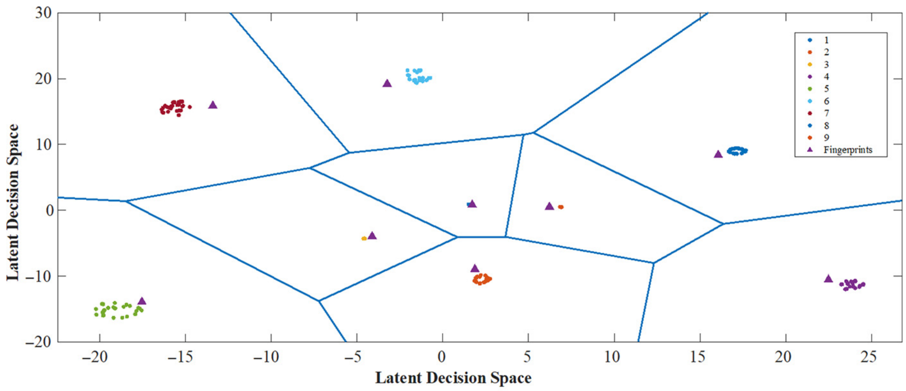
(T-sne)(Van Der Maaten and Hinton, 2008). T-Sne remapped high-dimensional extracted data into a 2D space vector while maintaining the same distribution for the data points. For remapping the high dimensional data, we consider Euclidean distance with an exact algorithm. For the normal data set, the Voronoi graphical representation visually shows the model performance by distributing all the known data samples into their respective regions and boundaries. As shown in Figure 11, all the extracted feature vectors were distributed closer to class fingerprints in the latent decision space.

The second batch combines normal and abnormal pictures from all the classes. As shown in Figure 12, the abnormal sample feature vectors were distributed farther away from the fingerprints than the normal data.

The True Positive Rate (TPR), True Negative Rate (TNR), False Positive Rate (FPR) and False Negative Rate (FNR) are important evaluation metrics for assessing the performance of a DNN model. TPR measures the percentage of abnormal samples correctly identified as abnormal, whereas TNR measures the percentage of the normal sample correctly identified as normal. FPR and FNR are measures of incorrectly measured samples. To calculate the values of all performance functions, we used two sets of data batches containing normal and abnormal samples individually. Initially, we passed all 108 abnormal images through our zero-bias DNN abnormality detection module. We recorded the accurate prediction (abnormal) as TP and false prediction (normal) as FP. Next, we imported 560 normal sample images and repeated this procedure. The correct prediction (normal) was registered as TN, and the incorrect prediction (abnormal) was recognized as FN. A cutoff threshold was used to characterize the delta between cutoff distance and MD. The model performance rates based on TPR, TNR, FPR and FNR as a function of the cutoff threshold is plotted in Figure 13. Table 3 lists the accuracy based on these measures. We find that the model accuracy based on these measures is more than 99%.

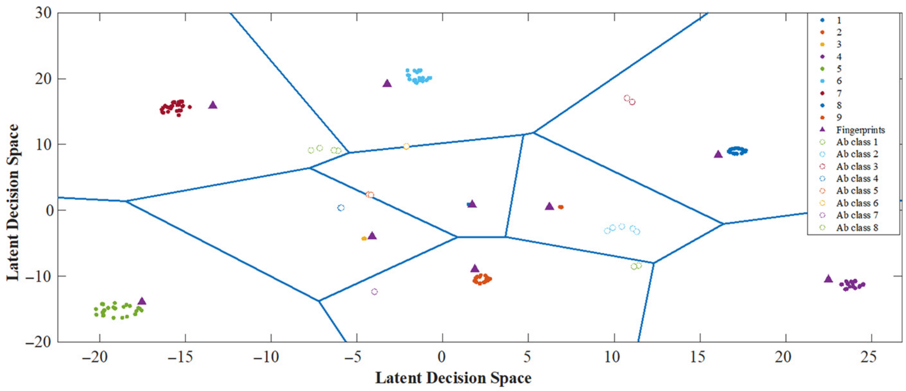
We compared the model performance on a standalone CPU which has 2.40 GHz dual-core processor, 64 GB RAM and an Intel i9 processor with a single Nvidia RTX3090 GPU computer. The GPU-based machine was able to train in 48 min and performed all tasks in less time than the CPU-based machine. The same input data were used three times for each sort of performance test, and the aggregated data were utilized to plot the results with the error bar. Figure 14 shows the performance of both machines for test data prediction, the trained model reload time for abnormality detection and the time taken for abnormality detection of a single input image. In our experiment, we captured two images per second and

Figure 11 Voronoi for the normal data set



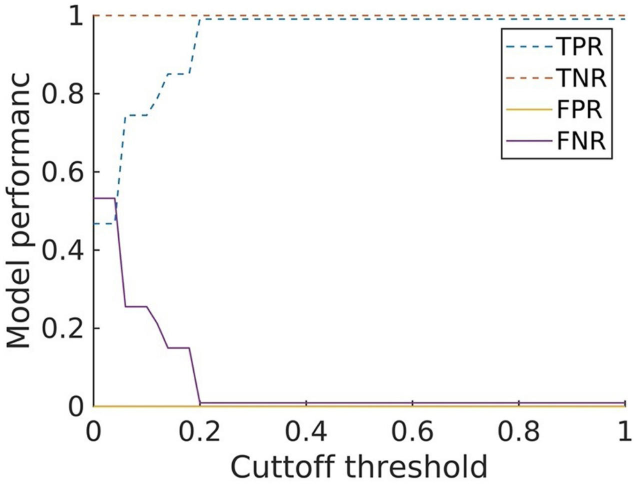
Source: Figure by authors

Figure 12 Voronoi for the mixed data set



Source: Figure by authors

Figure 13 Model performance characteristics



Source: Figure by authors

Table 3 Model performance characteristic

Characteristic	Formula	Efficiency (%)
True Positive Rate (TPR)	TP/(TP+FN)	99.08
True Negative Rate (TNR)	TN/(TN+FP)	100
False Positive Rate (FPR)	FP/(TN+FP)	0
False Negative Rate (FNR)	FN/(TP+FN)	0.92
Accuracy	TP+TN/(TP+TN+FP+FN)	99.54

Source: Table by authors

GPU-based machine was able to detect abnormal samples within 0.62 s, whereas CPU-based machine took 2.83 s. The computational performance is expected to significantly improve using high-performance parallel computing. These timing data suggest the potential of using this approach for real-time defect detection and process control.

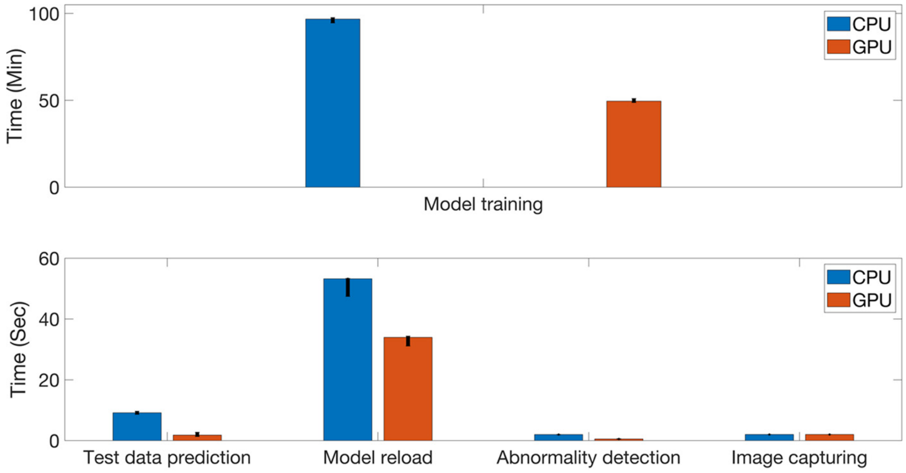
4. Discussion

In this study, we used a zero-bias DNN model trained on normal data to detect different types of printing defects including cracks, warping and stringing. The approach is based on replacing the last dense layer with two consecutive parts, a regular dense layer for dimensional reduction and a similarity-matching layer. This modification significantly simplifies the decision latent space. As we have removed the bias neurons in the final decision process of the network, for each input, the neural network outputs a vector of cosine similarity scores against templates of all classes in the latent space. The theoretical basis for anomaly detection with zero-bias neural networks is that inputs with unexpected defects or abnormal patterns will generate low similarity scores that are detectable by a global threshold.

This approach overcomes some of the major requirements of conventional anomaly detection approaches. First, while the decision processes of many DNN models are unclear, there is evidence that DNNs can unintentionally learn biases (Chen *et al.*, 2018; Chow *et al.*, 2020; Dhar and Shamir, 2022; Rahaman *et al.*, 2019). This can adversely affect defect detection and is particularly critical if the related data are to be used for process modification. One needs to understand the DNNs' decision processes to eliminate unreasonable biases. Current solutions (Alvi *et al.*, 2019; Tao *et al.*, 2016) either improve the quality of the training data set or use data augmentation methods, which encourage DNNs to learn robust features rather than address the root problem. The second problem is that conventional neural network classifiers lack the capability of detecting inputs from unknown classes. In manufacturing applications, this translates to the inability to detect new types of defects other than the specific defect the model is trained on. The approach presented here provides makes the decision process for defect detection transparent. Additionally, the approach eliminates the need for training on individual defects.

The transparency in the DNN model's decision-making is the basis for explainability and trustworthiness. Compared with some existing work such as GradCAM (Ahmed *et al.*, 2022)and LIME that reveal the pixels or regions that contribute positively

Figure 14 Computational timing performance for training and testing using CPU and GPU computers



Source: Figure by authors

or negatively to the classification score of each class, zero-bias neural network reveals why the decision is made by providing only the most relevant and vital information. Therefore, we believe the zero-bias neural network can be categorized as explainable AI but with a newer and more compact perspective.

The application of zero-bias DNN to AM shows promise for several further extensions. The proof of concept in this study was performed using FDM polymer printing. One limitation of interrupting the printing process is that it leads to a different consolidation process compared to an uninterrupted process, and has some drawbacks related to print quality, print alignment and calibration. We minimize these effects using small interruption time, accurate interruption points and appropriate pause-resume techniques supported by the 3D printer. Interrupting the printing process for in-situ characterization provides us with a normal data set at more stages of printing than without interrupting. This allowed us to analyze the printing process more frequently and provide early visible information on defects. Despite limitations, we used the interrupted characterization approach because it allowed us to analyze the printing process more frequently and provide early visible information on defects, and data sets for testing a new computational method.

Defects are ubiquitous in the AM of metallic and composite materials because of the higher thermal gradients and dissimilar properties of constituent materials. Extension of this approach to these materials is currently in progress and will provide an effective route for process improvement. Additionally, this approach lends itself to data fusion with inputs from multiple image data streams, such as grayscale images, thermal maps and deformation maps. Different data streams provide different perspectives with respect to defect identification and integrating them can significantly improve overall performance. The model accuracy can be improved and extended to complex printed shapes with enhancements such as image segmentation and localization of defects.

5. Conclusions

In this study, we implemented an explainable DNN framework for detecting processing defects during FDM AM. The last dense layer of the DNN is split into a regular dense layer (L1) for dimensional reduction, and a similarity matching layer (L2). This modification enables the output of a vector of cosine similarity scores against templates of all classes in the latent space which is used for defect detection. The data for the DNN modeling was acquired by in-situ imaging of an FDM printing process. The model was trained using a data set of normal images. The zero-bias DNN model was able to detect multiple defects such as cracks, warping and stringing without individual training on any defective sample images. The true positive, true negative, false positive and false negative rates for defect detection are 99.08%, 100%, 0% and 0.92%, respectively, with an abnormality detection accuracy of 99.54%. The computational performance of the model on GPUs suggests that the time required for defect detection is faster than the image acquisition rate, therefore the approach can potentially be used for process correction.

References

Ahmed, I., Jeon, G. and Piccialli, F. (2022), "From artificial intelligence to explainable artificial intelligence in industry 4.0: a survey on what, how, and where", *IEEE Trans Industr Inform*, Vol. 18, doi: 10.1109/TII.2022.3146552

Alvi, M., Zisserman, A. and Nellåker, C. (2019), "Turning a blind eye: explicit removal of biases and variation from deep neural network embeddings", *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, doi: 10.1007/978-3-030-11009-3_34

Angrish, A. (2014), "A critical analysis of additive manufacturing technologies for aerospace applications", *IEEE Aerospace Conference Proceedings*, doi: 10.1109/AERO.2014.6836456

Caggiano, A., Zhang, J., Alfieri, V., Caiazzo, F., Gao, R. and Teti, R. (2019), "Machine learning-based image processing for on-line defect recognition in additive manufacturing", *CIRP Annals*, Vol. 68, doi: 10.1016/j.cirp.2019.03.021.

Chen, Z., Yeo, C.K., Lee, B.S. and Lau, C.T. (2018), "Autoencoder-based network anomaly detection", *Wireless Telecommunications Symposium*, doi: 10.1109/WTS.2018.8363930

Cheng, Y. and Jafari, M.A. (2008), "Vision-based online process control in manufacturing applications", *IEEE Transactions on Automation Science and Engineering 5*, doi: 10.1109/TASE.2007.912058.

Chow, J.K., Su, Z., Wu, J., Tan, P.S., Mao, X. and Wang, Y. H. (2020), "Anomaly detection of defects on concrete structures with the convolutional autoencoder", *Advanced Engineering Informatics*, Vol. 45, p. 101105, doi: 10.1016/j.aei.2020.101105.

Chua, Z.Y., Ahn, I.H. and Moon, S.K. (2017), "Process monitoring and inspection systems in metal additive manufacturing: status and applications", *International Journal of Precision Engineering and Manufacturing-Green Technology*, Vol. 4 No. 2, doi: 10.1007/s40684-017-0029-7.

Colorado, H.A., Mendoza, D.E. and Valencia, F.L. (2021), "A combined strategy of additive manufacturing to support multidisciplinary education in arts", *Biology, and Engineering. J Sci Educ Technol*, Vol. 30, doi: 10.1007/s10956-020-09873-1.

De Maesschalck, R., Jouan-Rimbaud, D. and Massart, D.L. (2000), "The Mahalanobis distance", *Chemometrics and Intelligent Laboratory Systems*, Vol. 50 No. 1, doi: 10.1016/S0169-7439(99)00047-7.

Delli, U. and Chang, S. (2018), "Automated process monitoring in 3D printing using supervised machine learning", doi: 10.1016/j.promfg.2018.07.111

Despeisse, M. and Ford, S. (2015), "The role of additive manufacturing in improving resource efficiency and sustainability", *IFIP Advances in Information and Communication Technology*, doi: 10.1007/978-3-319-22759-7_15.

Dhar, S. and Shamir, L. (2022), "Systematic biases when using deep neural networks for annotating large catalogs of astronomical images", *Astronomy and Computing*, Vol. 38, p. 100545.

Fang, T., Jafari, M.A., Danforth, S.C. and Safari, A. (2003), "Signature analysis and defect detection in layered manufacturing of ceramic sensors and actuators", *Machine Vision and Applications*, Vol. 15 No. 2, doi: 10.1007/s00138-002-0074-1.

Fang, T., Bakhadyrov, I., Jafari, M.A. and Alpan, G. (1998), "Online detection of defects in layered manufacturing", *Proceedings – IEEE International Conference on Robotics and Automation*, doi: 10.1109/ROBOT.1998.676386

Grasso, M., Demir, A.G., Previtali, B. and Colosimo, B.M. (2018), "In situ monitoring of selective laser melting of zinc powder via infrared imaging of the process plume", *Robot ComputIntegrManuf*, Vol. 49, doi: 10.1016/j.rcim.2017.07.001.

Jani, M., Chava, S., Brown, J. and Namilae, S. (2022), "In-Situ monitoring of additive manufacturing using digital image correlation", *AIAA Science and Technology Forum and Exposition, AIAA SciTech Forum*, doi: 10.2514/6.2022-0076

Leal, R., Barreiros, F.M., Alves, L., Romeiro, F., Vasco, J.C., Santos, M. and Marto, C. (2017), "Additive manufacturing tooling for the automotive industry", *The International Journal of Advanced Manufacturing Technology*, Vol. 92 Nos 5/8, doi: 10.1007/s00170-017-0239-8.

Lewis, J.A. (2006), "Direct ink writing of 3D functional materials", *Advanced Functional Materials*, Vol. 16 No. 17, doi: 10.1002/adfm.200600434.

Liu, G., Xiong, Y. and Zhou, L. (2021a), "Additive manufacturing of continuous fiber reinforced polymer composites: design opportunities and novel applications", *Composites Communications*, Vol. 27, p. 100907, doi: 10.1016/j.coco.2021.100907.

Liu, Y., Wang, J., Li, J., Niu, S. and Song, H. (2021b), "Class-Incremental learning for wireless device identification in IoT", *IEEE Internet of Things Journal*, Vol. 8 No. 23, doi: 10.1109/JIOT.2021.3078407.

Liu, C., Law, A.C.C., Roberson, D. and Kong, Z. (2019), "Image analysis-based closed loop quality control for additive manufacturing with fused filament fabrication", *Journal of Manufacturing Systems*, Vol. 51, doi: 10.1016/j.jmsy.2019.04.002.

Liu, Y., Wang, J., Li, J., Niu, S., Wu, L. and Song, H. (2022), "Zero-Bias deep-learning-enabled quickest abnormal event detection in IoT", *IEEE Internet of Things Journal*, Vol. 9 No. 13, doi: 10.1109/JIOT.2021.3126819.

Mani, M., Lyons, K.W. and Gupta, S.K. (2014), "Sustainability characterization for additive manufacturing", *J Res Natl Inst Stand Technol*, Vol. 119, doi: 10.6028/jres.119.016.

Melchels, F.P.W., Domingos, M.A.N., Klein, T.J., Malda, J., Bartolo, P.J. and Hutmacher, D.W. (2012), "Additive manufacturing of tissues and organs", *Progress in Polymer Science*, Vol. 37 No. 8, doi: 10.1016/j.progpolymsci.2011.11.007.

Oleff, A., Küster, B., Stonis, M. and Overmeyer, L. (2021), "Process monitoring for material extrusion additive manufacturing: a state-of-the-art review", *Progress in Additive Manufacturing*, Vol. 6 No. 4, doi: 10.1007/s40964-021-00192-4.

Paraskevoudis, K., Karayannis, P. and Koumoulos, E.P. (2020), "Real-time 3d printing remote defect detection (stringing) with computer vision and artificial intelligence", *Processes*, Vol. 8 No. 11, doi: 10.3390/pr8111464.

Phillips, P.J., Hahn, C.A., Fontana, P.C., Broniatowski, D.A., Przybocki, M.A., Hahn, C.A. and Fontana, P.C. (2020), "Four principles of explainable artificial intelligence: draft NISTIR 8312", National Institute of Standards and Technology Interagency or Internal Report, doi: 10.6028/NIST.IR.8312-draft.

Rahaman, N., Baratin, A., Arpit, D., Draxlcr, F., Lin, M., Hamprecht, F.A., Bengio, Y. and Courville, A. (2019), "On the spectral bias of neural networks", *36th International Conference on Machine Learning*, ICML.

Rawat, D.B., Brecher, C., Song, H. and Jeschke, S. (2017), *Industrial Internet of Things: Cybermanufacturing Systems*, Springer, Cham.

Scime, L. and Beuth, J. (2019), "Using machine learning to identify in-situ melt Pool signatures indicative of flaw formation in a laser powder bed fusion additive manufacturing process", *Additive Manufacturing*, Vol. 25, doi: 10.1016/j.addma.2018.11.010.

Spencer, R., Hassen, A.A., Baba, J., Lindahl, J., Love, L., Kunc, V., Babu, S. and Vaidya, U. (2021), "An innovative digital image correlation technique for in-situ process monitoring of composite structures in large scale additive manufacturing", *Composite Structures*, Vol. 276, doi: 10.1016/j.compstruct.2021.114545.

Tao, Y., Gao, X., Hsu, K., Sorooshian, S. and Ihler, A. (2016), "A deep neural network modeling framework to reduce bias in satellite precipitation products", *Journal of Hydrometeorology*, Vol. 17 No. 3, doi: 10.1175/JHM-D-15-0075.1.

Tapia, G. and Elwany, A. (2014), "A review on process monitoring and control in metal-based additive manufacturing", *Journal of Manufacturing Science and Engineering*, Vol. 136 No. 6, doi: 10.1115/1.4028540.

Turner, B.N. and Gold, S.A. (2015), "A review of melt extrusion additive manufacturing processes: II. materials, dimensional accuracy, and surface roughness", *Rapid Prototyping Journal*, Vol. 21 No. 3, doi: 10.1108/RPJ-02-2013-0017.

Turner, B.N., Strong, R. and Gold, S.A. (2014), "A review of melt extrusion additive manufacturing processes: I", *Process Design and Modeling. Rapid Prototyp J*, doi: 10.1108/RPJ-01-2013-0012.

Vaezi, M., Chianrabutra, S., Mellor, B. and Yang, S. (2013), "Multiple material additive manufacturing – part 1: a review", *Virtual Phys Prototyp*, Vol. 8.

Van Der Maaten, L. and Hinton, G. (2008), "Visualizing data using t-SNE", *Journal of Machine Learning Research*, Vol. 9.

Ye, D., HsiFuh, J.Y., Zhang, Y., Hong, G.S. and Zhu, K. (2018), "In situ monitoring of selective laser melting using plume and spatter signatures by deep belief networks", *ISA Transactions*, Vol. 81, doi: 10.1016/j.isatra.2018.07.021.

Zhang, Y., Fuh, J.Y.H., Ye, D. and Hong, G.S. (2019), "In-situ monitoring of laser-based PBF via off-axis vision and image processing approaches", *AdditManuf*, Vol. 25, doi: 10.1016/j.addma.2018.10.020.

Zhang, Y., Hong, G.S., Ye, D., Zhu, K. and Fuh, J.Y.H. (2018), "Extraction and evaluation of melt pool, plume and spatter information for powder-bed fusion AM process monitoring", *Materials & Design*, Vol. 156, doi: 10.1016/j.matdes.2018.07.002.

Corresponding author

Sirish Namilae can be contacted at: namilaes@erau.edu