Proceedings of the ASME 2022
International Design Engineering Technical Conferences and
Computers and Information in Engineering Conference
IDETC/CIE2022
August 14-17, 2022, St. Louis, Missouri

DETC2022-88313

ANOMALY DETECTION OF LASER POWDER BED FUSION MELT POOL IMAGES USING COMBINED UNSUPERVISED AND SUPERVISED LEARNING METHODS

Matthew M. Sato[1], Vivian Wen Hui Wong[1], Kincho H. Law[1]
Ho Yeung[2], Zhuo Yang[2], Brandon Lane[2], Paul Witherell[2]

[1]Engineering Informatics Group, Civil and Environmental Engineering, Stanford University, Stanford, CA
[2]Engineering Lab, National Institute of Standards and Technology, Gaithersburg, MD

ABSTRACT

Laser Powder Bed Fusion (LPBF) is one of the most promising forms of Additive Manufacturing (AM), allowing easily customized metal manufactured parts. Industry use is currently limited due to the often unknown and unreliable part quality, which is largely caused by the complex relationships between process parameters that include laser power, laser speed, scan strategy, and other machine settings. Melt pools can be monitored with a camera aligned co-axially with the laser to monitor part quality. However, the number of images acquired can be large, exceeding hundreds of thousands for a single part. This paper investigates how the K-Means algorithm, an unsupervised machine learning method, can be used to cluster images of melt pools based on their shape, including undesirable anomalous melt pools. Another unsupervised learning method in this paper is the U-Net autoencoder, which identifies anomalous melt pools by identifying images with a large reconstruction loss. The K-Means clustering or autoencoder provides labels that can be used for training a convolutional neural network image classifier. The image classifier can then be used to identify anomalous melt pools during the LPBF process. This paper provides a first step for real-time process control of the LPBF process by demonstrating how anomalous melt pools can be automatically identified in real-time.

Keywords: Smart Manufacturing, Anomaly Detection, Additive Manufacturing, Convolutional Neural Network (CNN), K-Means Clustering, Autoencoder

1. INTRODUCTION

Laser Powder Bed Fusion (LPBF) is one of the most promising forms of metal additive manufacturing (AM). AM has gained popularity in manufacturing with its ability to quickly fabricate complex, customizable shapes while minimizing waste. LPBF is a layer-wise process which spreads a layer of metal powder on top of a substrate then selectively melts the powder with a laser. These steps are repeated layer-by-layer to manufacture the desired geometry. The region of molten metal is known as the melt pool. The process parameters chosen by the user include laser speed, laser power, layer height, and laser scan strategy, among others. The process parameters have a complex relationship that influences the quality of the manufactured part and is not fully understood [1]. The unknown and variable quality of the manufactured parts prevents widespread use of LPBF in industry [2]. Much work has been completed in defect detection, yet there are still large gaps in the understanding of LPBF and quality control of parts [3]. This paper works toward in-situ prediction of part quality, which helps reject bad parts or alleviate costly post-build part inspection.

Two of the most researched LPBF defects are lack of fusion porosity and keyhole porosity. Lack of fusion porosity is caused by the poor overlap of melt pools which leads to regions of unmelted powder. Keyhole porosity is a result of vaporization of the powder by the laser which creates a depression in the melt pool. The depression closes in on itself, entraining the vapor [4]. Porosity is important to the quality of LPBF parts as it can lead to poor mechanical behaviors of the parts [5]. Other LPBF defects include balling, cracking, geometric and surface defects, and microstructural inhomogeneities and are summarized in [2].

Assessing the quality of the LPBF process in real-time avoids the trial and error typically needed to find optimal process parameters. This paper's focus is to extract meaningful information from a melt pool monitoring system, which can indirectly provide information about the quality of the part. To monitor melt pools in real-time, a camera captures images of the melt pool throughout the LPBF process with examples shown in Figure 1. These melt pools can provide insight into the quality of the part through the behaviors of the process. For example, melt pools that are small, irregular shaped, or darker may indicate eventual defects in the manufactured part, although these descriptors are not exhaustive and may not always indicate anomalies. Of particular interest in this paper is the "torus" shaped melt pool in Figure 1 distinguished by the dark spot in the center of the melt pool. The torus-shaped melt pool is problematic because the dark area may indicate partially melted

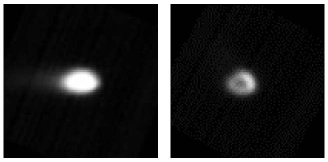

Figure 1: A TYPICAL MELT POOL (LEFT) AND A TORUS-SHAPED MELT POOL (RIGHT) FROM [25].

or unmelted metal, which could lead to porosities or other defects. For the remainder of this paper, any reference to anomalous melt pools refers to the torus-shaped melt pool unless otherwise stated. While this paper focuses mainly on this torus-shaped melt pool, the ideas are extendable to other types of anomalies and several examples are provided in the discussion.

The contributions of this paper include a two-step method for detecting anomalous melt pools with machine learning using originally unlabeled data. First, labels are automatically created for a large unlabeled dataset using unsupervised learning. Next, a supervised learning algorithm uses the newly created labels to classify the images as anomalous or non-anomalous. The image classifier can quickly identify anomalous melt pools and change process parameters in real-time to promote a higher quality part. The implementation of the real-time process control with an LPBF machine is beyond the scope of this paper. Ultimately, the paper provides an end-to-end process for melt pool anomaly detection which eliminates the need for costly human labeling.

The remaining sections of the paper include a literature review in Section 2, a description of the methodologies in Section 3, and a description of the experiment in Section 4. This paper is concluded with a discussion in Section 5 and final remarks in Section 6.

2. RELATED WORKS

Traditional methods to determine part quality include X-ray computed tomography (XCT) [6] and ultrasonic testing [7], which have been used to nondestructively evaluate the LPBF parts *after* manufacturing. XCT and ultrasonic testing can be effective methods for measuring the porosity of a part [4,7], but material and time are wasted since the part quality is known only after manufacturing. More recently, in-situ ultrasonic and X-ray imaging has been used in LPBF [8,9], but are in early stages of development and has mostly been limited to studies of fundamental melt pool dynamics rather than process monitoring.

The methods mentioned so far are primarily for offline process optimization, which determines the optimized process parameters from the experimental results before manufacturing. Even with optimized parameters, varying environmental conditions or machine uncertainty [10] could still lead to an undesirable part and demonstrates the necessity of a real-time monitoring system. To monitor the LPBF process in real-time, various melt pool monitoring strategies have been developed [2]. While melt pool monitoring cannot observe part defects as directly as XCT or ultrasonic testing, it may be incorporated into real-time process control.

Many melt pool monitoring methods rely on high-speed imaging or thermography. These imaging methods are often categorized as ‘staring’, in which a camera is in a fixed position with respect to the build area [11], or ‘co-axial’, where the camera is optically aligned with the processing laser such that the melt pool continuously remains in the camera’s field of view [12]. Staring configuration cameras have a fixed pixel resolution, which poses a tradeoff between the field of view and the resolution on the build plane. However, co-axial imaging can provide relatively higher pixel resolution that spans the entire build area, providing higher fidelity melt pool observation adaptable to real-time monitoring and control.

Machine learning is used to evaluate the large amount of data generated by melt pool monitoring and has become popular in defect detection for AM. Unsupervised learning methods have been used to cluster melt pool types together by first running an unsupervised learning algorithm to extract melt pool features and then running a different unsupervised learning algorithm to cluster the images [13,14]. Others have used unsupervised learning for anomaly detection on a layer-wise basis by evaluating an entire layer at a time [15]. Convolutional neural networks have also proved to be effective in anomaly detection for process monitoring systems [16,17].

This paper presents a new combination of unsupervised learning methods followed by a supervised convolutional neural network. This combination of tools allows for anomaly detection based on inference from an originally unlabeled dataset.

3. METHODOLOGY

This paper uses a combination of unsupervised and supervised learning techniques to detect anomalies as shown in the workflow in Figure 2. Unsupervised learning algorithms (K-Means or autoencoder) are first used to create labels for the large dataset of images, a task too large to be completed manually by a human. A convolutional neural network (supervised learning) trains with these labels to detect anomalies in new datasets. While the quality of the part and formation of defects is not directly observed, this paper shows how images from a co-axial camera may be processed in real-time to identify potentially anomalous melt pool behavior. In turn, these anomalous melt

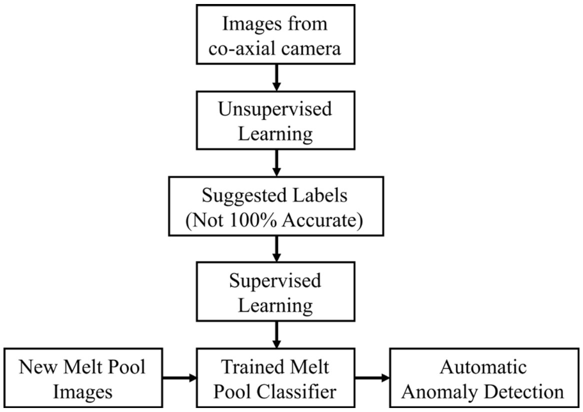
Images from co-axial camera
Unsupervised Learning
Suggested Labels (Not 100% Accurate)
Supervised Learning
New Melt Pool Images
Trained Melt Pool Classifier
Automatic Anomaly Detection

Figure 2: WORKFLOW FOR ANOMALY DETECTION.

pool observations may be correlated to part quality or defects, and ultimately used in real-time part quality prediction.

3.1 Unsupervised Learning Algorithms to Create Labels

Two different unsupervised machine learning methods, K-Means clustering and autoencoder, are used to identify anomalies and create labels for the dataset directly from melt pool images. The following subsections details how the two methods can be used independently or in combination to create labels for the originally unlabeled dataset.

3.1.1 K-Means Clustering

The K-Means algorithm is a relatively simple and computationally inexpensive method used to group unlabeled data into $\boldsymbol{k}$ distinct clusters. The K-Means algorithm first initializes $\boldsymbol{k}$ random cluster centroids and each image is assigned to the cluster of the nearest centroid. The algorithm seeks to minimize the distortion function:

$$\boldsymbol{J}(\boldsymbol{c}, \boldsymbol{\mu}) = \sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{n}} ||\boldsymbol{x}^{(\boldsymbol{i})} - \boldsymbol{\mu}_{\boldsymbol{c}^{(i)}}||^{\mathbf{2}} \qquad (\mathbf{1})$$

where for a total of $\boldsymbol{n}$ images, $\boldsymbol{x}^{(\boldsymbol{i})}$ is the $\boldsymbol{i}$'th image and $\boldsymbol{\mu}_{\boldsymbol{c}^{(i)}}$ is the centroid closest to the $\boldsymbol{i}$'th image. To minimize the distortion function, each centroid location is updated to the centroid of all images in its cluster. The images are then reassigned to their nearest cluster centroid. These steps are repeated until convergence.

The distortion function (Eq. 1) is non-convex and depending on the initial cluster centroids, different local optima may be obtained from the K-Means algorithm. To find the global optimum, the K-Means algorithm is run multiple times with different initial cluster centroids, and the solution with the lowest distortion is selected.

3.1.2 Convolutional Neural Networks

A Convolutional Neural Network (CNN) is a deep learning algorithm commonly used for image applications. CNNs are neural networks that typically use some combination of convolutional layers, normalization layers, and activation layers as shown in Figure 3. These combinations of layers have shown to be very successful in image classification for a variety of applications and are researched widely in medical imaging [18]. This subsection describes CNNs which is the basis for the autoencoder in Section 3.1.3 and image classifier in Section 3.2.

Convolutional layers are the distinguishing part of a CNN and convolve an input with a kernel to produce the output. Different kernels are used for purposes such as edge detection, image sharpening, etc.

The normalization layer normalizes the previous layers for use in future layers (such as activation layers). A typical normalization layer uses batch normalization, which normalizes the input by the mean of 0 and standard deviation of 1 within the same batch. Batch normalization is known to improve the performance of the CNN and tends to reduce the number of iterations needed for training [19].

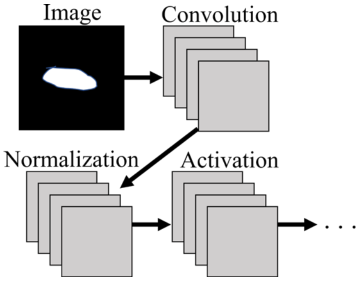

Figure 3: A TYPICAL CNN ARCHITECTURE.

The activation layer applies a nonlinear function such as the rectified linear unit (ReLU) or sigmoid activation function to the input. The activation follows the normalization layer.

The pooling layer is used to downsample the data. Convolutional layers may be susceptible to small changes in the location of the feature, and pooling layers help minimize this effect by reducing the feature size. Some typical forms of pooling are max pooling, which takes the maximum value of each patch as the output value, and average pooling, which takes the average value of each patch as the output value.

The CNN attempts to minimize the loss with each iteration of training, which is measured by a loss function. The loss function measures the difference between the CNN predictions and the ground truth. Backpropagation reduces the loss during training by using the gradient of the loss function to update the model parameters.

3.1.3 Autoencoder

Autoencoders are an unsupervised learning algorithm which can be used for anomaly detection. Autoencoders have been used for anomaly detection in a variety of applications including post-manufacturing defect detection of AM parts [20]. This paper uses an autoencoder as an alternative to K-Means for generating labels. The autoencoder is a CNN that first encodes an image by compressing the image through a series of convolutional layers and pooling layers. The autoencoder then decodes the encoded image with another series of convolutional layers to reconstruct the original image. The difference between the original image and the reconstructed image is referred to as the reconstruction loss. During training the autoencoder updates the CNN parameters through backpropagation to minimize reconstruction loss. The autoencoder is trained on non-anomalous images so that the autoencoder minimizes the loss when reconstructing normal images. Anomalous images are then identified by their high reconstruction loss compared to non-anomalous images. Typically, images with reconstruction loss above an established threshold are identified and labeled as anomalous. Some autoencoders use the U-Net architecture, in which the encoding and decoding sides of the autoencoder are symmetric as shown in Figure 4.

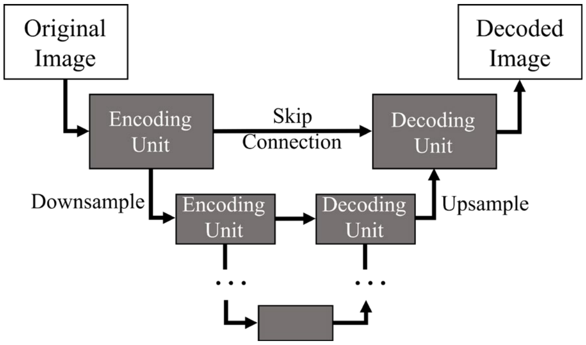

Figure 4: A VANILLA U-NET AUTOENCODER.

3.2 Supervised Learning Algorithm: CNN Classifier

A supervised learning CNN trained on the labels developed by K-Means or autoencoder is used to classify anomalies in new datasets. The CNN used for image classification is a deep neural network based on the MobileNetV2 architecture [21] that is pretrained on the ImageNet dataset [22], a set of over 1 million generic images. The MobileNetV2 architecture contains combinations of depthwise separable convolutional layers, batch normalization layers, linear bottlenecks, inverted residuals, and ReLU6 activation. Depthwise separable convolutional layers improve the efficiency of the neural network and separate a typical convolution into two separate layers: a single depthwise convolution filter per channel and a 1 × 1 pointwise convolution. The MobileNetV2 contains 53 convolutional layers. Linear bottlenecks use a linear function to reduce the size of the data. The inverted residuals provide a skip connection between bottlenecks. A final layer with sigmoid activation is added to the MobileNetV2 CNN. The sigmoid activation is used for the output (prediction) layer since the labels are binary.

Transfer learning is used to compensate for the limited number of training images and to reduce the training expense. Transfer learning is a technique in which information learned in one task is used to improve the generalization in another task and has been used in other settings to achieve higher accuracy where there is limited data [23,24]. Early layers in a neural network learn general features, such as edges, while later layers recognize more specific features of an image. Because many of the layers recognize features that occur in many images, a neural network pretrained on other generic images (ImageNet) leverages the common features between the melt pools and other generic images and do not need to be updated. Even though the ImageNet dataset contain color and no melt pool images, the pretrained CNN on ImageNet is still effective at extracting features from the melt pool images. Holding the pretrained neural network weights fixed and training a single new output layer with the melt pool images creates a task specific CNN which classifies anomalies in melt pool images.

4. Case Study

4.1 Dataset

The dataset used in this paper is publicly available as part of NIST's Additive Manufacturing Metrology Testbed [25]. The data is collected from the experiment described in [26], which consists of twelve LPBF parts with dimensions of 10 mm width, 10 mm depth, and 5 mm height. The metal powder used for the experiment was nickel alloy 625. Each part used a unique set of laser power, laser velocity, and scanning strategy. Each part had 250 layers with a constant layer height of 20 μm and a D4σ laser spot diameter of 85 μm. Some basic descriptions of the scan strategies [27,28] used are:

- Energy density: a commonly used term to describe the relationship between laser speed and power. Constant energy density means the ratio of power to velocity remains constant.
- Islands: the layer is separated into different regions, and each region is completed one at a time. This dataset uses four regions.
  - Island stripe: Each region is completed using a striping scanning pattern.
  - Island concentric: Each region is completed using a concentric scanning pattern.
- Continuous: the laser makes a smooth connection to the next line without stopping.
- Interleaved power: Several passes are first performed at a large spacing. The laser then scans in between the existing passes to fill in the layer.

Details of laser power/speed and scanning strategy for selected parts used in the dataset are provided in Table 1.

A co-axial melt pool monitoring camera was installed in the LPBF machine to capture images of the melt pools [26]. The camera was aligned with the laser so that each melt pool appeared in the same location in each image. The image is 8-bit grayscale. An integration time of 20 μs was chosen, where a grayscale of 80 approximately matches the melting point based on melt pool width comparison from previous experiments. The camera collects images from part 1 for layers 1, 13, 25, etc.; from part 2 for layers 2, 14, 26, etc.; and a similar pattern for parts 3-12. The camera captures 8-bit images at a rate of 2000 Hz. The dataset contains approximately 2500 to 7000 images per layer depending on the scan strategy. The dataset is unlabeled and contains blank images which occur from image capture when the laser power is off.

Since the laser scans in many different directions, the melt pool images are normalized through preprocessing. The images are rotated so that the melt pools correspond to left to right laser movement. The rotation is determined by assuming an elliptical melt pool shape and rotating the image so the major axis of the

Table 1: PROCESS PARAMETERS OF THE DATASET [26].

Part #	Laser Power/Speed	Scanning Strategy
10	Constant energy density	alternative stripe / island concentric layers, continuous
11	Constant energy density	Island stripe, continuous, interleaved power
12	Constant energy density	Island concentric, interleaved power continuous

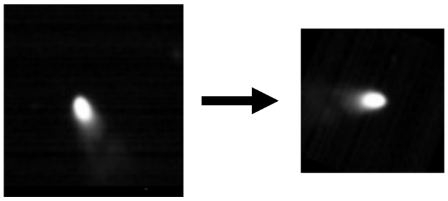

Figure 5: PREPROCESSING OF THE MELT POOL IMAGES.

ellipse is horizontal. Then, the images are cropped to 100 pixels × 100 pixels with the centroid of the melt pool centered in the image. Normalization is needed for the unsupervised learning methods so that the images are identified based on the melt pool shape and not the relative position or orientation of the melt pool. Figure 5 shows an example of preprocessed melt pool images.

4.2 K-Means Implementation and Results

To illustrate the K-Means algorithm, clustering was performed as described in Section 3.1.1 with several layers of the dataset that were known to contain anomalous-shaped melt pools. Performing the K-Means algorithm with 10 different random cluster centroid initializations and 300 iterations for each initialization performed well. Additional initializations or iterations increased computational cost with no improvement in performance. The only hyperparameter investigated is the number of clusters which largely influences the quality of the melt pool clustering.

To assess the clustering quality, the precision metric was used within the cluster containing torus-shaped melt pools. Precision is defined as the ratio of the number of human-identified torus-shaped melt pools in the cluster to the total number of melt pools in the cluster. Since this metric must be performed manually for unsupervised learning, 100 images were randomly sampled from the cluster to calculate the precision metric. The number of true positives and false positives were determined through human observation and precision was calculated. Clusters that were identified as anomalous but contained mostly non-torus melt pools were also sampled for torus melt pools (false negatives). While an estimate, these false negatives give some indication of the number of torus-shaped melt pools incorrectly grouped into a separate cluster.

The results of K-Means with different numbers of clusters are shown in Figure 6. For a small number of clusters, the precision is low, which means different types of anomalies are grouped in a single cluster. Since the anomalous melt pools are a small percentage of the images in the dataset, a small number of clusters does not sort different types of anomalies into unique clusters. For example, Figure 7 shows torus and non-torus melt pools in a single cluster when 5 clusters are used. These other melt pools may be considered anomalous (due to their small area or asymmetry) but may be the result of different process parameters. Instead, it is desirable to group each class of anomaly into a distinct cluster.

As the number of clusters increases, the precision increases as the torus melt pools are sorted into a distinct cluster. However, too many clusters result in multiple clusters containing mostly torus melt pools, which can be seen by the rise in the number of false negatives as the number of clusters increases from 10 to 20 in Figure 6. A balance must be maintained between precision and false negatives to obtain acceptable clusters. In this study, 12 clusters were chosen as the best option due to its high precision and low number of false negatives.

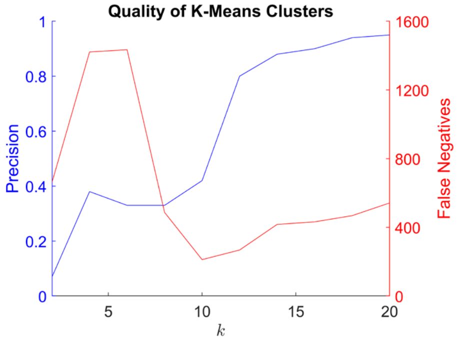

Figure 6: PRECISION STATISTIC FOR K-MEANS.

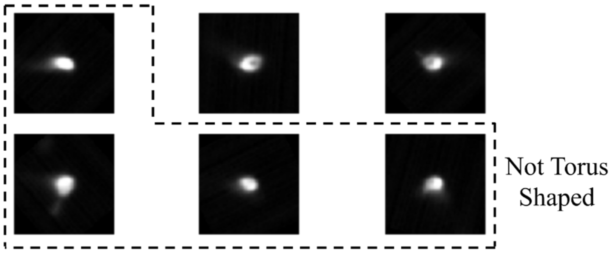

Figure 7: MELT POOLS REPRESENTATIVE OF A SINGLE CLUSTER WITH $\boldsymbol{k} = \mathbf{5}$.

K-Means with 12 clusters results in a cluster containing primarily torus anomalies with a low number of false negatives. The other anomalous melt pools from Figure 7 are separated into distinct clusters. The increase in clusters completes the goal: anomalous and non-anomalous melt pools are separated, and different classes of anomalies are grouped in separate clusters. The melt pools from the torus shaped cluster are labeled as anomalous and these labels are saved to train a CNN image classifier. Note that only binary labels are considered: the torus-shaped melt pools are labeled anomalous and all other melt pools are labeled non-anomalous. It is possible to separately label all different clusters but is not explored in this paper.

A further increase in clusters would provide greater precision but multiple clusters would contain the same class of anomaly. By inspecting the cluster outputs for $\boldsymbol{k} = \mathbf{12}$, each cluster contains a single unique shape and so this dataset contains approximately 12 different classes of melt pool shapes. Finding the appropriate number of clusters is the same as finding the unknown number of melt pool classes.

In summary, finding the optimal number of clusters requires a trial and error approach. The K-Means method can also be used to select and label a class of melt pool shapes other than torus-shaped straightforwardly. While the K-Means algorithm can provide good results once the optimal number of clusters is found, the process of finding the optimal number of clusters can be time-consuming and requires high levels of human interaction. A second method of labeling anomalous melt pools that does not require such a high level of human supervision is discussed next.

4.3 Autoencoder Implementation and Results

The autoencoder used in this paper uses the architecture in Figure 4 with five encoding units and five decoding units. Each encoding and decoding unit contains the following sequence of layers: 2D convolution, batch normalization, ReLU activation, 2D convolution, batch normalization, and ReLU activation. Downsampling is performed between encoding units with a max pool layer of kernel size 2, and upsampling is performed between decoding units using bilinear interpolation with a scale factor of 2. Mean squared error is used as the loss function:

$$\text{Loss} = \frac{\mathbf{1}}{\boldsymbol{N}}\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{N}}(\boldsymbol{y_i} - \boldsymbol{\hat{y}_i})^{\mathbf{2}} \qquad (\mathbf{2})$$

where $\boldsymbol{y_i}$ is the $\boldsymbol{i}$'th image and $\boldsymbol{\hat{y}_i}$ is the reconstruction of the $\boldsymbol{i}$'th image.

The autoencoder is trained on melt pool images taken during layers 200-202 and 205-211 of the part build, a total of 41 857 images. The images from these layers exclude the scan strategies used in parts 11 and 12 described in Table 1 since these scan strategies are the most experimental and are more likely to contain anomalies. Training the autoencoder on primarily non-anomalous melt pools allows for larger reconstruction loss of anomalous shaped images during testing and better results. The autoencoder was trained for 30 epochs with a learning rate of 0.001, batch size of 128, and the Adam optimizer [29]. The training time on a desktop machine with a NVIDIA GeForce MX250 4 GB GPU was approximately 9 hours. The loss of these training images after 30 epochs of training is shown in Figure 8. The loss from blank images is omitted from Figure 8 because they produce a much larger loss than other images. These blank images are easily filtered out based on the pixel values of the image or by the large losses being identified as outliers.

Now, a threshold can be established to identify anomalous melt pools. A typical threshold may use the interquartile rule:

$$\text{Threshold} = \boldsymbol{Q_{75}} + \mathbf{1.5} \times \text{IQR} \qquad (\mathbf{3})$$

where $\boldsymbol{Q_{75}}$ is the 75th percentile and IQR is the interquartile range ($\boldsymbol{Q_{75}} - \boldsymbol{Q_{25}}$). However, as seen in Figure 8, this threshold is too low, and too many normal melt pools would be identified within this threshold. Instead, thresholds equal to the 98th and 99th percentile were investigated. With the 98th percentile threshold, many classes of anomalies are identified. Torus-shaped images consist of only 11 % of the total images identified as anomalous as shown in Table 2. With the 99th percentile threshold, the autoencoder identifies three classes of anomalies: torus, large, and small melt pools shown in Figure 9. As with the blank images, the large melt pools can be filtered out based on the large values of the pixels in the image compared to the other melt pools. However, the torus melt pools and small melt pools have no features that are as easily distinguishable. So, the autoencoder is limited in that it selects melt pools that are either small or torus-shaped. As indicated in Table 2, the 99th percentile threshold identifies about half torus and half small melt pools, with much better results than the 98th percentile threshold.

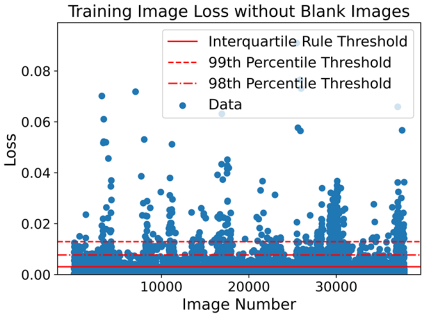

Figure 8: AUTOENCODER LOSS ON TRAINING IMAGES.

Table 2: PRECISION STATISTICS FOR AUTOENCODER.

Threshold	Precision
98 %	0.110
99 %	0.532
99 % + K-Means	0.915

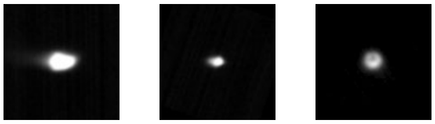

Figure 9: ANOMALIES IDENTIFIED BY AUTOENCODER.

The autoencoder required no human supervision in comparison to the K-Means method. The much lower precision for the autoencoder compared to the K-Means labels indicates a tradeoff between the level of human supervision needed and the quality of the labeling. To eliminate human supervision and have high precision labels, the two unsupervised learning methods can be combined. Because the 99th percentile threshold narrows the anomalies down to two classes of melt pools, the autoencoder can be combined with the K-Means method to eliminate the need to guess the number of clusters. The K-Means algorithm clustered the images that the 99th percentile autoencoder detected as anomalies into 2 clusters and Table 2 shows the precision was 0.915, larger than the K-Means or autoencoder alone. Because the autoencoder reliably selects only two classes of anomalies, the K-Means method does not require trial and error to tune the number of clusters. The labels created by both the 99th percentile autoencoder and 99th percentile autoencoder plus K-Means were saved for use in the image classifier in the next section.

The autoencoder approach is best suited for binary labeling, either anomalous or non-anomalous. On the other hand, the K-Means method can provide as many labels as the number of clusters selected. Different anomalies may be caused by different process parameters, so such distinct labeling may be beneficial as discussed in detail in a later section. The autoencoder combined with K-Means could also provide multiple labels, but not as many as the K-Means method alone.

Table 3: STATISTICS FOR THE CNN IMAGE CLASSIFIER ON THE TRAINING DATASET.

Epochs	Learning Rate	Training Time	Test Accuracy			Test Loss		
			K-Means	A.E.	A.E. + K-Means	K-Means	A.E.	A.E. + K-Means
10	0.001	15 min.	96 %	98 %	98 %	0.10	0.06	0.09
20	0.00001	26 min.	97 %	99 %	99 %	0.08	0.02	0.04
30	0.00001	23 min.	98 %	99 %	99 %	0.06	0.03	0.05

4.4 Image Classifier Implementation and Results on the Test Dataset

The CNN uses the MobileNetV2 architecture pretrained on ImageNet. To utilize transfer learning, only the weights of the output layer are updated, while the weights from MobileNetV2 pretrained on ImageNet are kept fixed. The output layer contains 1281 trainable parameters and training is completed for 10 epochs with a learning rate of 0.001. Cross entropy is used for the loss function, which is a typical loss function for binary labels:

$$\text{Loss} = -\frac{\mathbf{1}}{\boldsymbol{N}}\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{N}} \boldsymbol{y_i} \log \boldsymbol{\hat{y}_i} + (\mathbf{1} - \boldsymbol{y_i}) \log(\mathbf{1} - \boldsymbol{\hat{y}_i}) \qquad (\mathbf{4})$$

where $\boldsymbol{y_i}$ is the true label for the $\boldsymbol{i}$'th training example and $\boldsymbol{\hat{y}_i}$ is the predicted label for the $\boldsymbol{i}$'th training example. The algorithm used a minibatch size of 32 and the Adam optimizer [29].

After the initial training of the output layer, fine-tuning is completed by training some of the MobileNetV2 layers for an additional 20 epochs with a learning rate of 0.00001. This learning rate was lower than for the first 10 epochs to prevent overfitting and preserve the information from the ImageNet pretraining. In this paper, the last 100 layers of MobileNetV2 are also updated, which have 1 862 721 trainable parameters.

The CNN uses the labels developed from Sections 4.2 and 4.3. Non-anomalous melt pools were randomly selected to match the number of torus-shaped melt pools to produce a balanced dataset. The K-Means and autoencoder produce training, validation, and test sizes of about 18 000, 1800, and 1800, respectively. The autoencoder plus K-Means dataset contains approximately 10 000 training images, and 2000 validation and test images each.

Accuracy, or the percent of predicted labels that match the labels produced by the unsupervised learning methods, was used as the primary metric to evaluate the learning algorithm. The accuracy and loss for the neural networks trained on the three different datasets are shown in Table 3. The classifier reaches good accuracy after the first 10 epochs (96 % to 97 % accuracy) on all three datasets and even better accuracy after an additional 10 epochs (97 % to 99 % accuracy). After 30 epochs the accuracy reaches 98 % for the K-Means dataset and the model trained for 30 epochs is chosen for analysis. The autoencoder and autoencoder plus K-Means models see an increase in test loss after 30 epochs of training and no increase in accuracy, so the model after 20 epochs of training is selected for these training datasets.

Table 4: EVALUATION METRICS OF IMAGE CLASSIFIERS ON THE SECOND DATASET.

Training Labels Model	Recall	Precision	F_1
K-Means	0.984	0.651	0.783
Autoencoder	0.731	0.868	0.794
A.E. + K-Means	0.862	0.751	0.803

4.5 Image Classifier Results on an Unseen Dataset

All three training datasets produce an image classifier which provide high accuracy on the test dataset. However, these datasets were labeled via unsupervised learning and the test accuracy may not represent the true accuracy since the labels were not perfect. To better understand the success of the image classifiers, the three models were used to identify anomalous melt pools in a new dataset created from a separate build. The geometry and process parameters of the parts for the new dataset are similar to the original dataset and provide new melt pool shapes to test the CNN.

The image classifiers were tested on 187 005 images from seven layers selected from part 10 of the new dataset. The images were manually labeled with assistance from the image classifiers. Instead of labeling the entire set of images, only images taken nearby anomalies identified by the image classifiers were inspected, thereby, narrowing the number of images to just hundreds. Then, using previous experience, a human identified and labeled any anomalous melt pools. This example of labeling the images shows the utility of the image classifier in an offline sense. Instead of a human quality control agent needing to parse through hundreds of thousands of images to identify any anomalous melt pools, the image classifier can narrow the number of images down to only hundreds. A total of 305 torus melt pools were found in the manual labeling process.

The results of the image classifier compared to the human labels are shown in Table 4. The recall metric is the ratio of torus anomalies that are correctly identified to the total number of torus anomalies. The precision metric is the fraction of identified melt pools that are truly torus-shaped. The F1 metric is the harmonic mean of recall and precision and combines the recall and precision metrics.

The results show that the classifier trained with K-Means labels has the highest recall and lowest precision. While this image classifier correctly selects many torus melt pools, it has the most false positives among the models trained. Interestingly, the image classifier trained on the autoencoder labels has the highest precision, even though the training labels for the autoencoder had the lowest precision of the three training sets. An intermediate between the two models is the classifier trained on the combined autoencoder and K-Means labels. This classifier can correctly identify approximately 86 % of torus melt pools while having a precision between the other two classifiers.

Lastly, on a desktop machine with a NVIDIA GeForce MX250 4 GB GPU, the time to process and evaluate a single image is about 0.01 seconds. For real-time evaluation, this evaluation rate is slower than the capture rate from the datasets used in this paper. However, a faster machine may be used to reduce the evaluation time or images can be taken at a lower frequency. Ultimately, the evaluation speed is quick enough to provide meaningful information in a real time setting.

5. DISCUSSION

5.1 Process Parameters Leading to Anomalous Melt Pools

Because the quantity of anomalies is low compared to the total number of images, the connection between the process parameters and melt pool shapes is needed to ensure the anomalous melt pools are not random or due to camera malfunction. Here, we look at Part 10 from the second dataset which is known to have torus-shaped melt pools. As described in Table 1, Part 10 alternates each layer between a striping scan strategy (even layers) and an island concentric scan strategy (odd layers). Figure 10 shows the location of torus melt pools identified by the classifier trained on the autoencoder plus K-Means labels in several layers of the build. It is possible local material properties contribute towards some of the defects, but the remainder of this section shows the defects from the perspective of process parameters.

First, consider the detected melt pools identified with the label "1" in Figure 10. This melt pool is the first melt pool captured by the camera in each layer and is consistently torus-shaped in each layer. A possible cause of this torus melt pool is the powder is not sufficiently heated at the beginning of the layer, and thus very little melting occurs. While typically there are no additional anomalies directly after this melt pool, layer 232 shows continued anomalies and the image classifiers would be able to alert the machine of the continued anomalies.

Next, consider the anomalous melt pools identified with the label "2" in Figure 10. These melt pools occur at the center of each concentric island in odd layers and are falsely detected by the image classifier. These melt pools correspond to the very large melt pools in Figure 11. These melt pools are caused by the concentration of heat at the center of the island. In this scanning strategy, the laser completes concentric circles from the outside to the inside. As the laser nears the center of the island, the circles are much smaller which means the laser is heating the same general area for a longer period, and the temperature of the metal increases, promoting the larger melt pool shape in Figure 11. While not torus-shaped, these melt pools are an anomaly, and the excessive heat and melt pool could lead to keyholing.

Lastly, consider the melt pools identified on the edges of the parts in Figure 10, which consists of most of the melt pools identified as torus-shaped. These melt pool images are taken during the "contouring" of each layer, or when the laser traces the outline of the part at the beginning of each layer. As is commonly done in practice, the contour process parameters are different than the infill parameters: the laser velocity and power are lower. For part 10, the laser energy density of the contouring is also lower than during the infill.

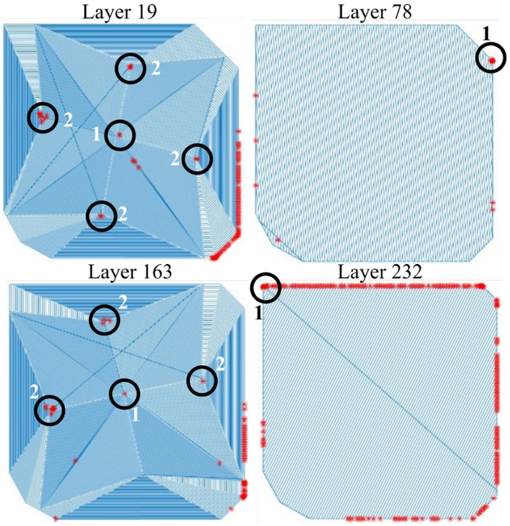

Figure 10: THE LOCATION OF DETECTED TORUS MELT POOLS IN PART 10 IN THE SECOND DATASET.

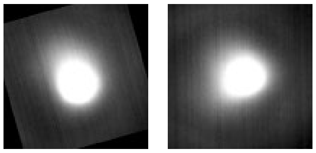

Figure 11: THE MELT POOL SHAPE ASSOCIATED WITH THE "2" LABEL IN Figure 10.

While there exist melt pools that are truly false positives such as those in Figure 11, we note that the precision cited in Table 4 is likely an underestimate of the true precision. The manual labeling of the images excluded some melt pool shapes which were anomalous but not torus-shaped. However, most of the false positives selected by the image classifiers are still anomalies, just not exactly torus-shaped. These false-positive images mostly occur within the same time range as the torus-shaped images. Consider layer 141 of part 10 in the second dataset and the classifier trained on K-Means labels, which has a precision of 0.485 for this layer. As shown in Figure 12, almost all the false positive images detected occur along the right edge

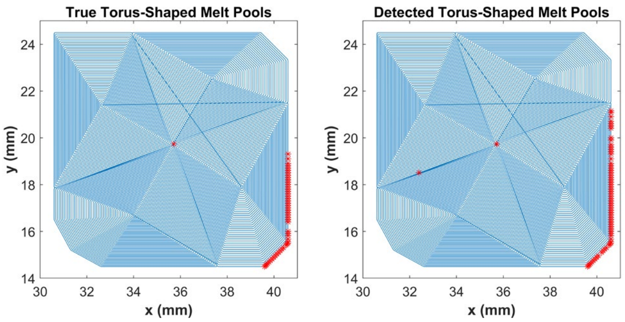

Figure 12: THE LOCATION OF TRUE AND DETECTED TORUS-SHAPED MELT POOLS IN LAYER 141 OF PART 10 IN THE SECOND DATASET.

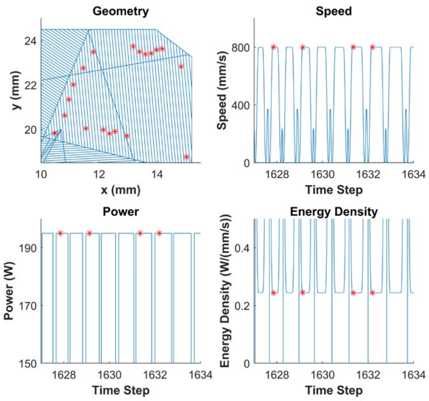

Figure 13: THE PROCESS PARAMETERS OF PART 7 LAYER 127 (* INDICATES THE ANOMALOUS MELT POOL).

of the part which is also where all the true torus melt pools occur. The false positive anomalous melt pools are likely caused by the same process parameters leading to torus melt pools, and even though they are not torus-shaped, the image classifier still identifies them. Whereas a human would see two distinct melt pool shapes, the CNN identifies features in the melt pool images that suggest both anomalies are created by the same process parameters.

This paper has focused on torus melt pools. However, the methods described in this paper are generalizable and useful for other anomalous melt pools shapes identified by the unsupervised learning methods. For example, consider the large, high-intensity melt pools that were filtered from the autoencoder. These images can be labeled and used to train an image classifier just as with the torus melt pools. Figure 13 shows the spatial location of these anomalous melt pools and the corresponding process parameters in layer 127 for part 7 in the original dataset. Notice that the anomalies occur immediately after an increase in speed and power. The cause is explained through the physical requirements of the LPBF machine. As the laser makes each striping pass, it must slow down, stop, and speed back up in the next direction. As demonstrated in Figure 13, these slowdowns are also accompanied by a decrease in power to keep the energy density constant. However, these turns also cause the heat to concentrate, like the situation of melt pools labeled “2” in Figure 10. This heat concentration causes larger melt pools and could lead to keyholing. An image classifier that looks for these melt pools could indicate to the machine that the power needs to be reduced further on turns to prevent the problem in the future.

6. CONCLUSION

This paper explores the use of machine learning methods for identifying anomalies from melt pool images. This paper shows how unsupervised learning methods, such as the K-Means algorithm or autoencoder, can be used to effectively label the images of the melt pools. The quantity of images produced during the LPBF process is too large to be reasonably labeled by humans and these methods provide varying levels of quality in the labeling. Using a combination of an autoencoder followed by K-Means provides the best labeling with very minimal human oversight. Trained using the labeled data from the unsupervised learning methods, the CNN image classifiers demonstrate success in identifying torus-shaped melt pools on independent datasets. The image classifiers can identify anomalies caused by varying physical processes and can do a better job than a human of identifying melt pool features that are caused by the same process parameters.

While much work remains to be done in automating the image labeling process (with minimal human efforts) and the training of effective anomaly detection (classification) models, the efforts in this paper contribute to real-time process monitoring and control that seeks to change process parameters in real-time to minimize low-quality LPBF parts. By identifying poor quality melt pools in real-time, the machine settings can be updated to produce melt pools of higher quality, which can produce manufactured parts of higher quality. Future research may investigate other methods to identify anomalies, such as semi-supervised learning. Semi-supervised learning uses a partially labeled dataset and could provide an economic alternative to the completely unlabeled dataset in this work. Also, a metric for evaluating melt pool quality could be established to reduce the human subjectivity, perhaps by comparing XCT images with the location of potentially anomalous melt pools. Other work can determine which machine updates should occur when low-quality melt pools are detected. In summary, this paper provides a framework for process control of LPBF additive manufacturing that can lead to high quality and more reliable LPBF parts.

ACKNOWLEDGEMENTS

The effort by the researchers at Stanford University has been partially supported by the Stanford Center at the Incheon Global Campus (SCIGC), which is sponsored in part under the National Program to Subsidize Attracting Foreign Educational Institution and Research Institutes published by the Ministry of Trade, Industry, and Energy of the Republic of Korea and managed by the Incheon Free Economic Zone Authority.

Certain commercial systems are identified in this article. Such identification does not imply recommendation or endorsement by NIST; nor does it imply that the products identified are necessarily the best available for the purpose. Further, any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of NIST or any other supporting U.S. Government or corporate organizations.

REFERENCES

[1] Wang, P., Yang, Y., and Moghaddam, N. S. “Process Modeling in Laser Powder Bed Fusion towards Defect Detection and Quality Control via Machine Learning: The State-of-the-Art and Research Challenges.” *Journal of Manufacturing Processes,* 73:961-984, 2022.

[2] Grasso, M., and Colosimo, B. M. “Process Defects and in Situ Monitoring Methods in Metal Powder Bed Fusion: A Review.” *Measurement Science and Technology,* Vol. 28, 2017.

[3] Wang, P., Yang, Y., and Moghaddam, N. S. "Process Modeling in Laser Powder Bed Fusion towards Defect Detection and Quality Control via Machine Learning: The State-of-the-Art and Research Challenges." *Journal of Manufacturing Processes*, 73: 961–984, 2022.

[4] du Plessis, A. "Effects of Process Parameters on Porosity in Laser Powder Bed Fusion Revealed by X-Ray Tomography." *Additive Manufacturing* Vol. 30, 2019.

[5] Ronneberg, T., Davies, C. M., and Hooper, P. A. "Revealing Relationships between Porosity, Microstructure and Mechanical Properties of Laser Powder Bed Fusion 316L Stainless Steel through Heat Treatment." *Materials & Design*, Vol. 189, 2020.

[6] du Plessis, A. "X-Ray Tomography for the Advancement of Laser Powder Bed Fusion Additive Manufacturing." *Journal of Microscopy*, 2020.

[7] Kim, C., Yin, H., Shmatok, A., et.al. "Ultrasonic Nondestructive Evaluation of Laser Powder Bed Fusion 316L Stainless Steel." *Additive Manufacturing*, 38, 2021.

[8] Martin, A. A., Calta, N. P., Hammons, J. A., et.al. "Ultrafast Dynamics of Laser-Metal Interactions in Additive Manufacturing Alloys Captured by in Situ X-Ray Imaging." *Materials Today Advances*, 1:100002, 2019.

[9] Honarvar, F., and Varvani-Farahani, A. "A Review of Ultrasonic Testing Applications in Additive Manufacturing: Defect Evaluation, Material Characterization, and Process Control." *Ultrasonics*, 108: 106227, 2020.

[10] Reutzel, E. W., and Nassar, A. R. "A Survey of Sensing and Control Systems for Machine and Process Monitoring of Directed-Energy, Metal-Based Additive Manufacturing." *Rapid Prototyping Journal*, 21:159–167, 2014.

[11] Shrestha, S., and Chou, K. "Single Track Scanning Experiment in Laser Powder Bed Fusion Process." *Procedia Manufacturing*, 26:857-864, 2018.

[12] Clijsters, S., Craeghs, T., Buls, S., et.al. "In Situ Quality Control of the Selective Laser Melting Process Using a High-Speed, Real-Time Melt Pool Monitoring System." *The International Journal of Advanced Manufacturing Technology* 75(5):1089-1101, 2014.

[13] Scime, L., and Beuth, J. "Using Machine Learning to Identify In-Situ Melt Pool Signatures Indicative of Flaw Formation in a Laser Powder Bed Fusion Additive Manufacturing Process." *Additive Manufacturing*, 25:151-165, 2019.

[14] Fathizadan, S., Ju, F., and Lu, Y. "Deep Representation Learning for Process Variation Management in Laser Powder Bed Fusion." *Additive Manufacturing,* 42, 2021.

[15] Mahmoudi, M., Ezzat, A. A., and Elwany, A. "Layerwise Anomaly Detection in Laser Powder-Bed Fusion Metal Additive Manufacturing." *Journal of Manufacturing Science and Engineering*, 141(3), 2019.

[16] Baumgartl, H., Tomas, J., Buettner, R., et.al. "A Deep Learning-Based Model for Defect Detection in Laser-Powder Bed Fusion Using in-Situ Thermographic Monitoring." *Progress in Additive Manufacturing*, 5(3):277-285, 2020.

[17] Westphal, E., and Seitz, H. "A Machine Learning Method for Defect Detection and Visualization in Selective Laser Sintering Based on Convolutional Neural Networks." *Additive Manufacturing*, 41:101965, 2021.

[18] Anwar, S. M., Majid, M., Qayyum, A., et.al. "Medical Image Analysis Using Convolutional Neural Networks: A Review." *Journal of Medical Systems*, 42(11):226, 2018.

[19] Ioffe, S., and Szegedy, C. "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift." *Proceedings of the 32nd International Conference on Machine Learning*. Lille, France, 2015.

[20] Wong, V., Ferguson, M., Law, K., et.al. "Segmentation of Additive Manufacturing Defects Using U-Net." *Journal of Computing and Information Science in Engineering* 22(3):1-44, 2021.

[21] Sandler, M., Howard, A. G., Zhu, M., et.al. "Inverted Residuals and Linear Bottlenecks: Mobile Networks for Classification, Detection and Segmentation." *IEEE Conference on Computer Vision and Pattern Recognition*. Salt Lake City, UT, 2018.

[22] Deng, J., Dong, W., Socher, R., et.al. "ImageNet: A Large-Scale Hierarchical Image Database." *IEEE Conference on Computer Vision and Pattern Recognition*. Miami, FL, 2009.

[23] Deepak, S., and Ameer, P. M. "Brain Tumor Classification Using Deep CNN Features via Transfer Learning." *Computers in Biology and Medicine*, 111:103345, 2019.

[24] Kolar, Z., Chen, H., and Luo, X. "Transfer Learning and Deep Convolutional Neural Networks for Safety Guardrail Detection in 2D Images." *Automation in Construction*, 89:58-70, 2018.

[25] Lane, B., and Yeung, H. "Process Monitoring Dataset from the Additive Manufacturing Metrology Testbed (AMMT): 3D Scan Strategies." National Institute of Standards and Technology, 2019. DOI:10.18434/M32044.

[26] Lane, B., and Yeung, H. "Process Monitoring Dataset from the Additive Manufacturing Metrology Testbed (AMMT): 'Three-Dimensional Scan Strategies.'" *Journal of Research of the National Institute of Standards and Technology,* 124:124033, 2019.

[27] Yeung, H., Lane, B., Donmez, M., et.al.. "Implementation of Advanced Laser Control Strategies for Powder Bed Fusion Systems." *46th SME North American Manufacturing Research Conference*. College Station, TX, 2018.

[28] Yeung, H., Lane, B., Fox, J., et.al. "Continuous Laser Scan Strategy for Faster Build Speeds in Laser Powder Bed Fusion System." *Solid Freeform Fabrication*. Austin, TX, 2017.

[29] Kingma, D. P., and Ba, J. "Adam: A Method for Stochastic Optimization." *3rd International Conference for Learning Representations*. San Diego, CA, 2015.