The Pennsylvania State University

The Graduate School

A HYBRID DEEP LEARNING MODEL OF PROCESS-BUILD INTERACTIONS IN ADDITIVE MANUFACTURING

A Thesis in

Industrial Engineering

by

Reza Mojahed Yazdi

Submitted in Partial Fulfillment

of the Requirements

for the Degree of

Master of Science

August 2020

The thesis of Reza Mojahed Yazdi was reviewed and approved by the following:

Hui Yang

Harold and Inge Marcus Career Associate Professor of Industrial and Manufacturing Engineering

Thesis Advisor

Soundar Rajan Tirupatikumara

Allen E. Pearce and Allen M. Pearce Professor of Industrial and Manufacturing Engineering

Clyde Lee Giles

Professor of Information Sciences and Technology

Steven Landry

Professor of Industrial and Manufacturing Engineering

Department Head of Industrial and Manufacturing Engineering

Abstract

Laser powder bed fusion (LPBF) is a technique of additive manufacturing (AM) that is often used to construct a metal object layer-by-layer. The quality of AM builds depends to a great extent on the minimization of different defects such as porosity and cracks that could occur by process deviation during printing operation. Therefore, there is a need to develop new analytical methods and tools to equip the LPBF process with the inspection frameworks that assess the process condition and monitor the porosity defect in real-time. Advanced sensing is recently integrated with the AM machines to cope with process complexity and improve information visibility. This opportunity lays the foundation for online monitoring and assessment of the in-process build layer. This study presents a hybrid deep neural network structure with two types of input data to monitor the process parameters that result in porosity defect in cylinders' layers. Results demonstrate that statistical features extracted by wavelet transform and texture analysis along with original powder bed images, assist the model to reach a robust performance. In order to illustrate the fidelity of the proposed model, the capability of the main pipeline is examined and compared with different machine learning models. Eventually, the proposed framework identified the process conditions with an F-score of 97.14%. This salient flaw detection ability is conducive to repair the defect in real-time and assure the quality of the final part before the completion of the process.

Table of Contents

List of Figures

List of Tables

Acknowledgments

I would first like to express my gratitude to my supervisor Dr. Hui Yang for the useful guidance and support through the learning process of this master thesis. His expert advice and friendly remarks have been invaluable throughout all stages of the work.

I would also like to acknowledge my sincere appreciation to the readers of my thesis, Dr. Soundar Kumara and Dr. Lee Giles for their precious insights and constructive feedback on this work.

I would also like to thank my colleague, Farhad Imani in the Complex Systems Monitoring, Modeling and Analysis Laboratory, for his help and directions in this project.

Finally, my deep and sincere gratitude to my family for their endless and unparalleled love, help and support. I am forever grateful to my mother Maryam Matin, my father Kamran Mojahed Yazdi and my sister Helia. This journey would not have been possible without their support, and I dedicate this milestone to them.

Chapter 1

Introduction

The latest industrial revolution, known as industry 4.0, is offering new solutions to enable global enterprises to move toward bringing automation as well as data-driven intelligence into production processes [1, 2]. Among the nine main technologies defined in industry 4.0 [3], additive manufacturing (AM) is one of the widely growing approaches that fabricates a customized 3D object, layer by layer [4]. This technology provides the flexibility of producing 3D components with complex geometry from their CAD model [5]. This, offers new potential for building customized devices in a short period of time.

Laser powder bed fusion (LPBF), also known as selective laser sintering (SLS), is a technique of AM that constructs an object by employing a laser beam as an energy source to fuse certain regions of a powder material which has been already applied to a build plate. Then, the build plate is lowered and the process is repeated to build the next layer upon the previous one. This operation continues until the complete object is printed [6]. Fig. 1.1 shows the schematic diagram of the LPBF process. The laser with the power in the range of 200 W to 400 W, scans the powder bed with linear scan speed in the order of 200 mm/s – 2000 mm/s. In addition, the laser spot size is in a range of 50 µm – 100 µm in diameter. LBPF has gained attention due to its capability to build objects with complex geometry.

However, the performance of the printer is highly dependent on process parameters

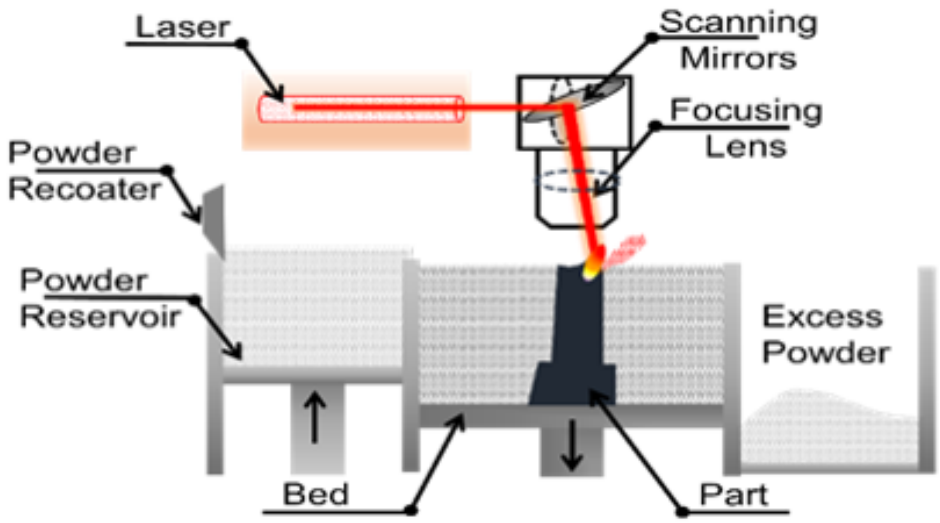

Figure 1.1. The schematic diagram of LPBF process.

since a small deviation in these elements can lead to the appearance of different defects on the part. Among various defects, porosity is one of the major flaws that reduce the quality of the final object. Three parameters including hatch spacing, laser print velocity, and laser power are known as the main causes for emerging porosity [7]. These parameters together establish the energy density measure [8] which has a direct effect on part quality. If the energy density grows excessively, the powder material is vaporized, whereas insufficient energy density leads to large and misshaped pores. These defects can change the functionality of the build. Therefore, the LPBF process is urgently needed to be integrated with the inspection frameworks that assess the process condition and monitor the porosity defect in real-time.

This study aims to present a hybrid model to monitor the corresponding parameters of the porosity defect in AM objects in real-time and assess the quality of final build. Here, convolutional neural networks, wavelet transform and texture analysis are the key instruments to extract the salient features from these layerwise images with minimum loss of information [9, 10]. Experimental results showed that combining these entries with powder bed images is conducive to generating a higher fidelity model compared to

traditional models.

The remainder of this thesis is organized as follows:

Chapter. 2 presents a literature review of previous studies on sensing methods and AM process monitoring models.

Chapter. 3 elaborates on research methodology and proposed model for defect inspection. Here, we discuss the data prepossessing steps as well as proposed model architecture.

Chapter. 4 illustrates and analyzes the results obtained after implementing the hybrid model on the layerwise image of the LPBF process and makes a performance comparison between the proposed model and traditional models.

Chapter. 5 summarizes this research and concludes the study.

Chapter. 6 discusses the future work.

Chapter 2
Literature Review

In this chapter a literature review of different sensing and data gathering methods in manufacturing systems and previous researches focused on AM process quality inspection will be presented.

2.1 Data gathering in manufacturing

Comprehensive reviews of LBPF process monitoring using different process signatures are discussed in Ref. [11, 12]. Advanced sensing technologies, create new grounds for further studies to explore rich information in the manufacturing process such as acoustic signals, melt pool pictures, powder bed images, and temperature distribution [13]. Cerniglia et al. [14] conducted a numerical analysis to show the capability of ultrasonic sensors for identifying porosity defects during the LPBF process. Rieder et al. [15] utilized the same tool to perform the in-process measurement of residual stress in a single layer and track the build status. Zhao et al. [16] utilized the high-speed X-ray images as well as diffraction techniques to gather quantitative details about different phenomena that appear during the LPBF process such as rapid solidification and powder ejection which can result in different defects in a layer and affect the quality of the entire object. Craeghs et al. [17, 18] devised the optical-based approaches for tracking the thermal behavior at

the melt pool. Among different types of sensors, the application of optical cameras is significantly growing in recent studies because they are capable of realizing the precise, real-time and cost-effective monitoring.

2.2 Process monitoring in Additive Manufacturing

To analyze the build quality and control of the manufacturing process, scientists have been exploring different data-driven and statistical solutions in recent years. Seifi et al. [19] performed a layerwise quality control in order to detect the healthy and unhealthy melt pools and conduct the online prediction of defect distribution in the printed layer. Yao et al. [20] utilized the constrained Markov decision process framework to model the variation of flaws in AM build and predict the evolving dynamics of defect state from one layer to the next layer. Imani et al. [21] performed multifractal analysis on layerwise images to define new features for capturing defect patterns and assess the build quality. Khanzadeh et al. [22] captured the thermal profile of melt pool and compared the ability of different machine learning algorithms in anomaly detection for AM parts.

Deep learning is a subset of machine learning methods in which we use sets of deep neural network (DNN) to build the predictive models. In the past few years, promising results have proved the capability of the DNN-based solutions to establish the AM process monitoring methods [23]. Wang et al. [24] discussed the application of different types of deep learning frameworks in developing diagnostics and predictive analytics models for AM process. Among them, the convolutional neural network (CNN) has progressed significantly in creating image-guided defect inspections frameworks [25]. Zhang et al. [26] developed a system that employs CNN to identify the level of quality PBF. Shevchik et al. [27] collected acoustic signals in the manufacturing process and performed the wavelet analysis on initial signals to create a new representation of data in time-frequency domain. Then, the CNN is utilized as a classifier to monitor the quality in real-time.

Scimeand and Beuth [28] collected the layerwise images and leveraged the AlexNet CNN to create the autonomous anomaly detection model for AM process.

2.3 Contribution

As it is mentioned above, in the machine learning approaches, AM defects are identified by statistical features extracted manually from original images. The selection of these features relies on researchers' experience and requires domain knowledge. Therefore, the performance of the traditional machine learning model is highly dependant on the effectiveness of handpicked extracted features, and missing one crucial feature can affect the model accuracy.

On the other hand, DNN approaches automatically capture key features of interests from input data and construct the final feature map. However, there are some obstacles that hinder these models from learning higher-level features and thus decrease the final detection performance. First, DNN needs lots of data to be trained. But, in the AM field, it is often hard to prepare huge data. Particularly, obtaining sufficient data for all classes of defect and condition is difficult, as it is expensive to reproduce the AM build. This creates an imbalanced dataset which leads to an overfitting problem. Moreover, CNN models utilized the pooling layer to downsample the feature map. This process can cause loss of information, particularly in complex surfaces and low contrast images of AM parts. Therefore, this work aims to address these challenges. To the best of our knowledge, it is the first of its kind where the original layerwise image is integrated with its statistical feature to achieve a hybrid input DNN-based process monitoring system.

Chapter 3

Research Methodology

This chapter presents the new pipeline for real-time AM process condition monitoring. Here, setting of sensing system and machine setup under which the input data gathered are considered. Next, we provide the detail of wavelet transform and texture analysis methods which are used to extract statistical features from build images. Finally, the structure of proposed DNN model is defined.

3.1 Data Acquisition

This research is aimed at developing a model to perform AM process monitoring and defect detection. Therefore, we need to establish a mechanism to analyze the image data as well as extract salient statistical features. The image dataset is prepared by conducting the laser powder bed fusion AM process that receives Titanium alloy and ASTM B348 Grade 23 Ti-6Al-4V powder material to print a cylinder that consists of 348 layers and has a length of 25 mm and diameter of 20 mm. Three main parameters are defined before operating the 3D printer machine including Hatching Space (Lh), laser print velocity(Lv) and laser power(Lp). The initial values of this settings are selected as $Lh_0 = 0.12$ mm, $Lv_0 = 1250$ mm/s, and $Lp_0 = 340$ W. In the experiment, hatching space and laser print velocity are increased by 25 percent and 50 percent of their initial

values and laser powder is decreased by 25 percent and 50 percent of its initial value. These parameters shape the energy density measure, i.e., Andrew number which can be stated as,

$$EA = \frac{Lp}{Lh \times Lv} J/(mm^2)S \tag{3.1}$$

Imani et al. [29] conducted the porosity analysis on XCT scans images to illustrate that the variations of process influence the size and number of pores presented on the layer's surface. As the Andrew number increases, the number of pores decreases, thus, the quality of the printed layer and the final build is improved. This concept enables us to label the collected data based on the energy density under which the object is printed. To acquire images of layers produced in different process conditions, cylinders are printed through using 7 different selections of machine settings which result in 5 distinct Andrew numbers. Table. 3.1 shows process conditions and final energy densities under which objects are printed. During the process, the layer-by-layer bed images are captured just after laser exposure by a digital single-lens reflex camera (DSLR, Nikon D800E) whose shutter is controlled by a proximity sensor to monitor the location of the re-coater blade. Multiple flash modules are embedded in different spots of the build chamber. Fig. 3.1 illustrates the imaging system's setup. The layewise images of LBPF process are captured under five different light scheme.Fig. 3.2 represents sample of parts printed under these flash settings. In this work, the cylinder's images taken in light scheme shown in in 3.2(a) are analyzed.

As a result, considering five classes of process condition, 3132 layerwise images are collected. To obtain non-image input features of the hybrid model, we utilize wavelet transform and texture analysis which will be elaborated in the subsequent sections.

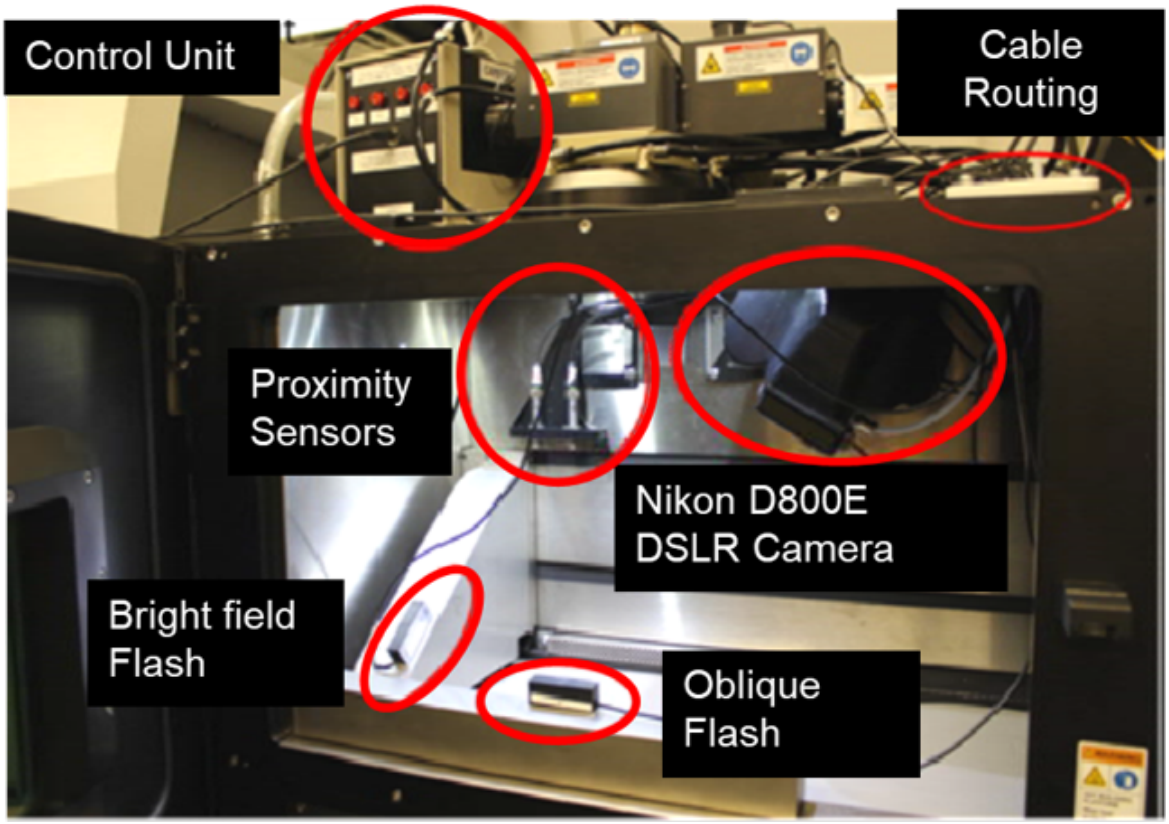

Figure 3.1. The Setup of optical imaging system in the LPBF machine to capture in-process layerwise images.

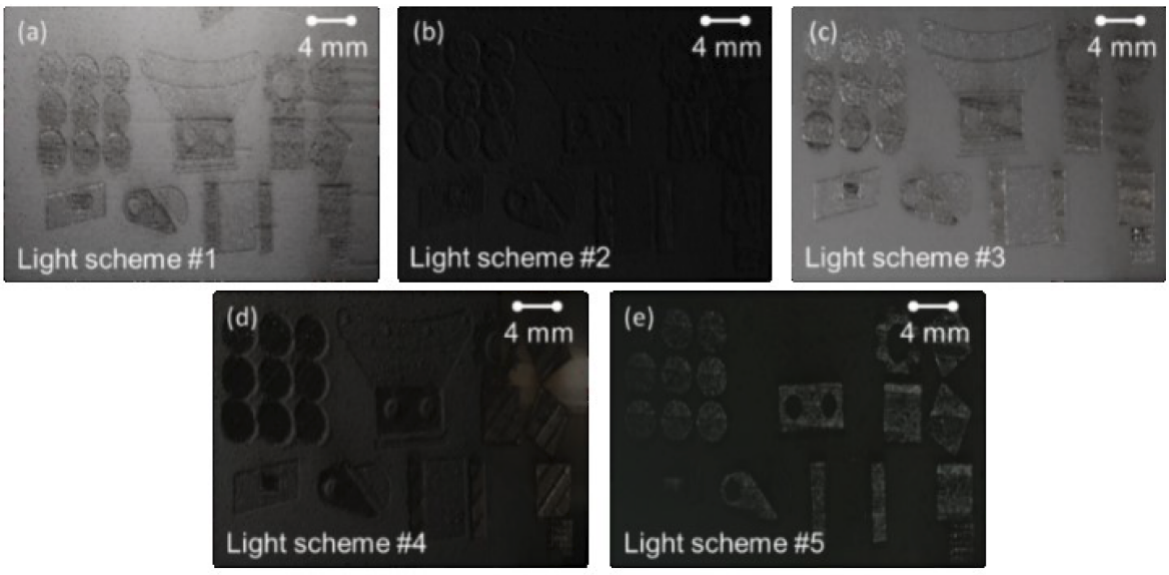

Figure 3.2. Image of powder bed in different light scheme

3.2 Wavelet Transform

Wavelet transform is one of the powerful tools in signal and image processing that analyze data to provide some insights about the temporal and spatial information [30,31]. Wavelet analysis generates new components named wavelet decomposition coefficients through the multi-resolution analysis. The wavelet function processes a signal at different resolutions in order to define the time-scale mapping which is useful for capturing low frequencies in large windows while localizing sharp fluctuations in small windows.

There are two main types of wavelets, i.e. Continues Wavelet Transform (CWT) and Discrete Wavelet Transform (DWT) [32]. CWT generates correlated wavelets that result in redundant information. In contrast, by applying DWT, we can achieve mutually orthogonal wavelets. Therefore, we choose DWT to produces a compact set of features. Hereafter, a basic wavelet called as mother wavelet should be selected among a variety of selections. Among different wavelets, Haar wavelet is chosen because it is the only wavelet which is compactly supported, orthogonal and has symmetry [33]. Moreover, it is computationally efficient. Haar is powerful for detecting sudden changes, while other wavelets are more useful for image and texture smoothing. In eq. (3.2), $W_{(\alpha,\beta)}$ represents the continuous wavelet transform of signal x(t).

$$W_{(\alpha,\beta)} = \int_{-\infty}^{+\infty} x(t) \frac{1}{|\alpha|^{\frac{1}{2}}} \psi^*(\frac{t-\beta}{\alpha}) dt \tag{3.2}$$

During the wavelet transform process, mother wavelet $\psi^*(t)$ is scaled by parameter α and translated by parameter β. Constant $\frac{1}{|\alpha|^{\frac{1}{2}}}$ is used for energy normalization. Here, the input signal x is the 2-dimensional powder bed image. Now, if we confine these two values to the discrete domain such that α and β can only be increased in the power of two and integer steps respectively, we attain the discrete version of the wavelet transform

Table 3.1. The energy density (EA) computed by different selection of laser parameters.

Process Condition			
(P, H, V)	[W, mm, mm/sec]	EA [J.mm 2] Andrew's number	Class
P0, H0, V0	(340, 0.12, 1250)	2.27	A
P0, H0,V0+25%	(170, 0.12, 1562)	1.81	B
P0, H+25%, V0	(170, 0.15, 1250)	1.81	B
P-25%, H0, V0	(255, 0.12, 1250)	1.70	C
P0, H+50%, V0	(170, 0.18, 1250)	1.51	D
P0, H0,V0+50%	(170, 0.12, 1875)	1.51	D
P-50%, H0, V0	(170, 0.12, 1250)	1.13	E

that is expressed as,

$$A_{i,k} = \sum_{n} x[n] G_i[n - 2^i k] \tag{3.3}$$

$$B_{i,k} = \sum_{n} x[n] H_i[n - 2^i k] \tag{3.4}$$

where G is a function that represents the high pass filter (wavelet filter) that is useful for localizing high frequencies in signal and capturing sharp edges. It will be multiplied by the input signal to create a coefficient of detail components that is shown by A. H denotes a function for the low pass filter (scaling filter), which creates a down-sampled version of the powder bed image and generates the coefficient of approximation component which is referred by B. These two steps together shape a complete process of single-level one dimensional DWT. As we are dealing with the image data here, the 1D DWT operates on the rows and columns of images separately. Consequently, each powder bed image is decomposed into four sub-bands. Fig. 3.3 illustrates the schematic diagram of single-level 2D DWT. LH, HL and HH are detail coefficients that localized horizontal, vertical and diagonal features of the image respectively. LL is the approximation coefficient and comprises the compressed version of the image.

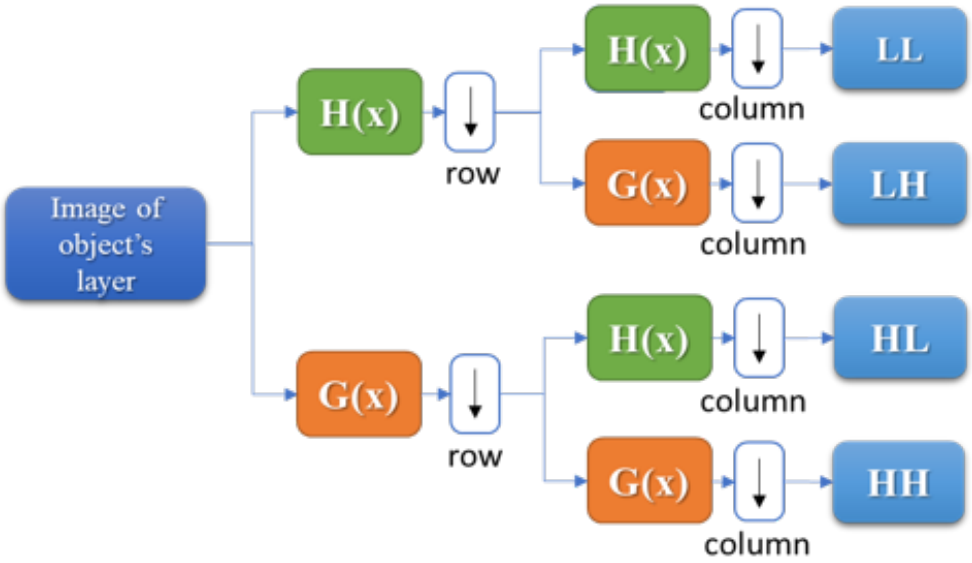

Figure 3.3. The Schematic Diagram of single-level two-dimensional DWT. To implement the 2D DWT function on an image, the 1D DWT is applied along the rows and columns of the image respectively to produce four sub-bands.

3.3 Texture Analysis

In this section, we propose to identify the textures of different regions in wavelet by using Shannon's entropy which first introduced by Claude Shannon in 1948 [34]. The entropy concept is used to measure the information contains in the signal. Data with more information (uncertainty) give a higher entropy. Hence, in 2D wavelet coefficients, higher entropy represents a greater difference among regions in terms of pixel intensity. Entropy can be expressed as,

$$E(x) = -\sum_{i=1}^{n} P_i \log_2 P_i \tag{3.5}$$

where

$$P_i = \frac{\gamma_i}{\Gamma} \tag{3.6}$$

Here, P_i is the marginal distribution of gray values in the 2D wavelet. In eq. (3.6),

number of pixels with gray level i denoted by γ_i and Γ refers to the total number of pixels. As the transformed wavelets are 8-bit images, the range of gray-level resolution could be a value between 0 to 255. Mean and standard deviation of wavelet coefficients, are representative measures for noise effect and sharp variations respectively [35]. These two statistics are extracted from decomposition coefficients to finalize the feature database. In eq. (3.7) and eq. (3.8), p and q are used to refer to the wavelet's row size and column size respectively and gv(m,n) represents the gray value of a pixel located in coordination gv(m,n) in wavelet.

$$Mean = \frac{1}{pq} \sum_{(m,n)} gv(m,n) \tag{3.7}$$

$$STD = \sqrt{\frac{1}{pq-1} \sum_{(m,n)} \left(gv(m,n) - \frac{1}{pq-1} \sum_{(m,n)} gv(m,n) \right)^2} \tag{3.8}$$

3.4 The hybrid Model Architecture for the Analysis of Build Interactions

The DNN models are created by embedding multiple hierarchical hidden layers between input and output layers. The algorithm designs the feature extraction tool to learn and retain the underlying architecture in each class of data and retain them for creating the final feature map. As shown in Fig. 3.4, the proposed model is hybrid as it encompasses two parallel DNNs and utilizes hybrid inputs.

In the initial network, we adopt the transfer learning to utilize a pre-defined configuration that is liable to devise a feature map through processing the original layerwise images. VGG-16 structure [36] is selected based on its robust performance compared to other image classification models. The input layer of VGG-16 CNN originally accepts

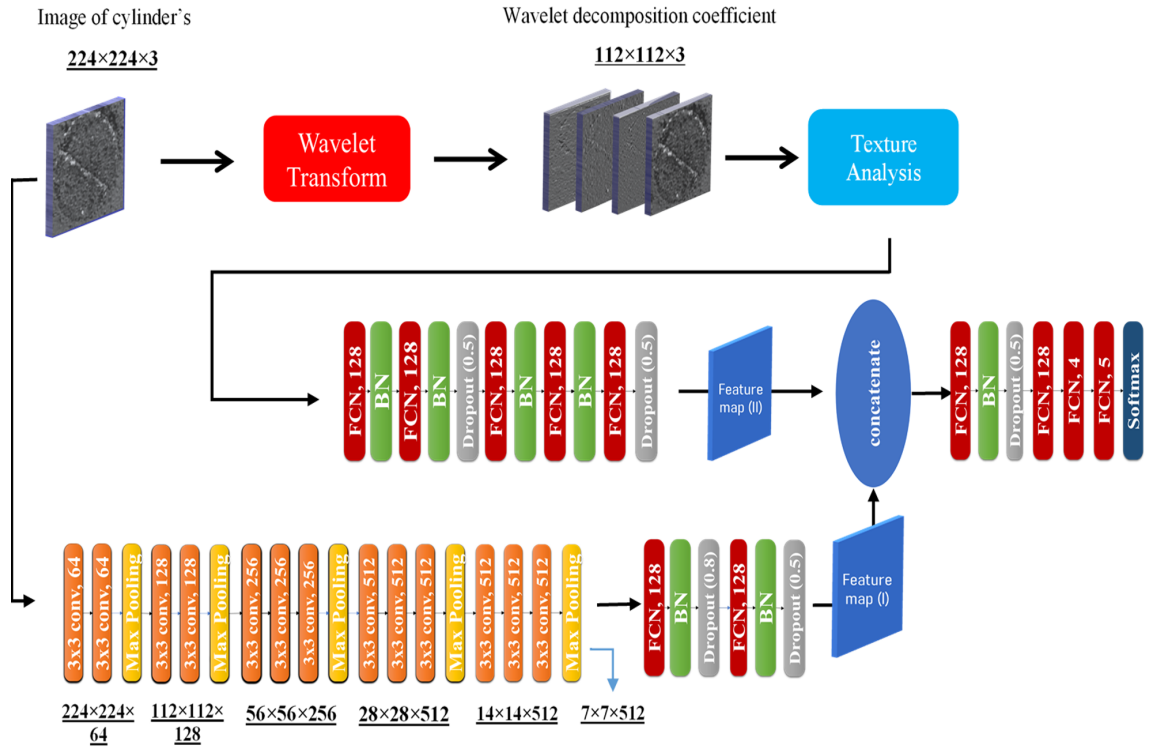

Figure 3.4. The hybrid model pipeline. Each cylinder's image is passed through the first branch of hybrid model which consists of VGG 16 structure followed by dense layers. Simultaneously, statistical features obtained by performing texture analysis on wavelet coefficients are fed to the second branch that includes the MLP layers. Then, two generated feature maps will be merged together and traverse the fully connected layers for predictive modeling. BN: Batch Normalization.

color images with a size of 224 × pixels × 224 pixels × 3 pixels. Here, as the captured images are 1-channeled, we need to repeat and stack gray-scale images along the third axes to make the input compatible with model structure. Once the data inserted into the network, mathematical operations are applied sequentially in hidden layers to create feature maps that are often non-trivial and cannot be explained by human. The learning process begins with a convolution layers which consists of multiple kernels(filters). Then, each weight of kernels is assigned to a pixel of input image and weighted sums are calculated. Next, the weighted sum will be added to a bias value and passed through the

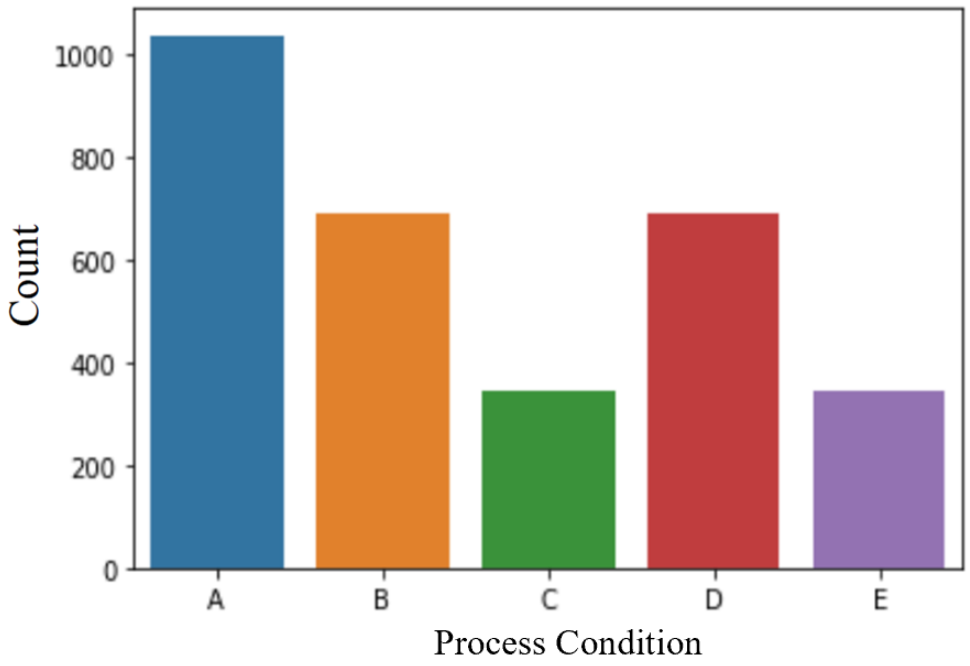

Figure 3.5. Distribution of images by classes of process condition

activation function. The convolution function is given in eq. (3.9).

$$Y_{e,r}^{L} = \Theta\left(\sum_{k=1}^{K}\sum_{v=1}^{V} Y_{(e-k+K,r-v+V)}^{(L-1)} * W_{k,v}^{L} + b^{L}\right) \tag{3.9}$$

where, $Y_{e,r}^{L}$ shows the calculated feature map of L^{th} layer created in the size of e $\times$ r, b^{L} represents the bias, Θ is a non-linear activation function, * denotes the convolutional operator and $W_{k,v}^{L}$ shows the weight with the position (k,v) in feature map of L^{th} convolution layer. This weight will be multiplied by the pixel located in coordination (e-k+K , r-v+V) in calculated feature map of $(L\text{-}1)^{th}$ layer shown by $Y_{(e-k+K,r-v+V)}^{(L-1)}$. Next, based on a predefined stride size, the convolution kernel moves to the next position and convolution operation is performed. This process is then repeated and operated on full depth of the given layer. The width and height of feature map generated by

convolution operation can be determined as follows:

$$Z_0 = \frac{Z - \zeta + 2\upsilon}{S} \tag{3.10}$$

where Z_0 represents the output layer height (or width), Z denotes the input layer height (or width), ζ is the height (width) of the kernel, υ is the padding size and S is the stride size of the kernel in the height (width) direction. Also, the depth of the output feature map is equal to the number of kernels. To fine tune the model and impart the non-linearity, the rectified linear unit (ReLU) is selected as the activation function [37]. The Relu function with input feature map σ is given in eq. (3.11). In addition, the Max Pooling layer is embedded throughout the model to downsample the size of data and reduce the training time. This operator only passes the maximum value of the given window on to the next layer. VGG 16 makes use of window size of 3 pixels $\times$ 3 pixels with the stride of two in all Max Pooling layers. The top layers used in the pre-trained structure of VGG-16 will be replaced by two new fully connected (FCN).

$$ReLU(\sigma) = max(0, \sigma) \tag{3.11}$$

In another separate branch of hybrid model, statistical features obtained from the texture analysis step, will traverse the multilayer perceptron (MLP) neural network to generate the second feature map and boost the VGG-16 branch. Hereafter, two feature maps are merged together to design a unit that will be flattened and followed by the 4 fully connected layers. In addition, The batch normalization and dropout layers are embedded in the model to accelerate the learning procedure and tackle the overfitting problem [38, 39].

For the learning phase, the softmax activation function is added to estimate the probability of each label using output of final FCN layer. The output of softmax layer is the 1$\times$c vector where c is the number of class labels. The sum of values in this

Table 3.2. Model training parameters

Parameter	Value
Final layer activation function	Softmax
Optimizer	RMSprop
Learning Rate	0.001
Number of epoch	300
Early stopping patience	20

vector is unity. Then the parameters of the network, i.e, weights in convolution kernels and the bias values, are updated by minimizing the cross-entropy loss function with back-propagation algorithm [40]. The equation of Softmax and Cross-entropy functions are as follows:

$$f(\delta) = \frac{e^{\delta_i}}{\sum_i^C e^{\delta_i}} \tag{3.12}$$

$$CE = -\sum_i^C t_i \log(f(\delta)_i) \tag{3.13}$$

where, δ_i is the i^{th} value in output vector of last FCN layer, f(δ) is the softmax score,st_i denotes the ground truth of class i and CE represent the cross-entropy cost function.

During the model training phase, the RMSProp method is employed to optimize the variables.The learning rate is $\iota = 0.001$ and batch size n=64 is selected. the maximum number of epochs is limited to 300. However, the early stopping technique [41] is utilized to compare the performance of model in different epochs and terminate the training process earlier if the model does not show any more capacity to decrease the test loss after 20 consecutive epochs. For example, if the model obtains the test loss value of 0.5 in the 10th epoch and cannot reduce this loss before the 30th epoch, the learning process will be ended. The network architecture is implemented with TensorFlow [42]. The training parameters are summarized in Table. 3.2,

As shown in Fig. 3.5, the number of available data among different classes are imbalanced. This can lead to considerable bias to large classes and increases the error of the final trained model. To address this issue, different weights are defined for each class. We incorporate the weights of each class into the loss function. In fact, these weights which are inversely proportional to the size of the class, will be multiplied by the loss value of the corresponding class. It will force the learning process to impose more penalty on false prediction when the true label belongs to the classes with less data. These weights can be computed as:

$$\phi(i) = \frac{T}{C \times T_i} \tag{3.14}$$

where $\phi(i)$ denotes the weights of the ith class, T is the total number of samples, Ti is the number of samples in class i and C in the number of classes. Note that, test and validation samples are not considered for calculating class weights.

Chapter 4

Experimental Design and Results

Thus far, we designed a framework to extract meaningful statistical features and classify the build-image with regard to the condition process. In this chapter, we represent the result of feature analysis steps. Different classifiers with different entry are utilized to detect the process condition. Considering different evaluation metrics, the classification performance of all models are assessed and compared with the result of proposed hybrid model.

4.1 Feature analysis of Powder bed Images

Fig. 4.1 illustrates sample images of the cylinder's layer printed under different process conditions with different energy densities. These images are used as the first input to the proposed model. Then, the single-level DWT of these images creates four wavelet sub-bands. Fig. 4.2 depicts wavelet decomposition coefficients extracted from a captured powder bed image. In the next step, each wavelet sub-image underwent through the texture analysis process and 3 features including mean, standard deviation and entropy are obtained per each sub-band.

As a result, for each image, we generate 12 statistical features. This statistical data will be leveraged to be the second input to the model. In fact, by concatenating

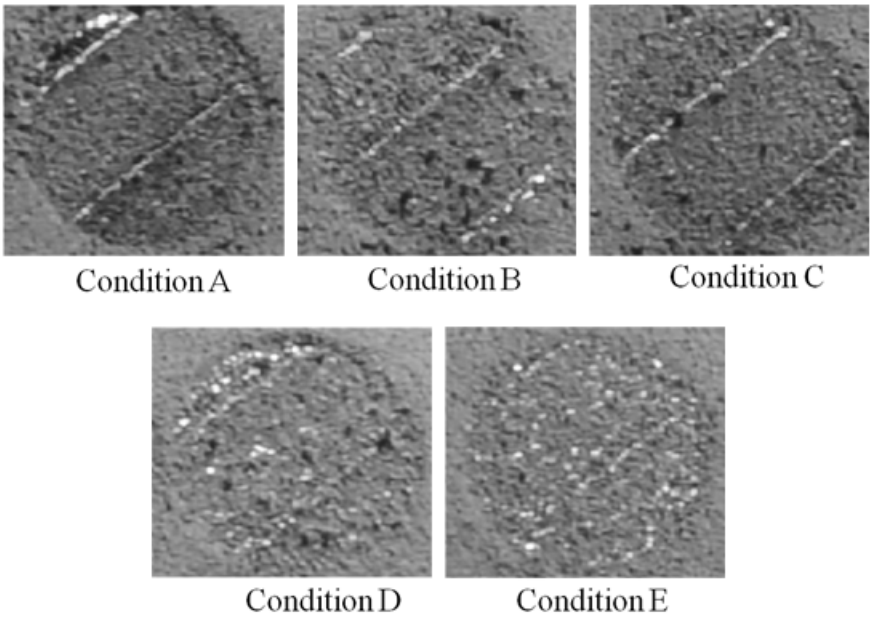

Figure 4.1. Samples of a single cylinder's layer (10th layer) that built under different process conditions.

additional information to image data. we expect the model to gain higher detection power by considering a broader range of information.

4.2 Online monitoring of Process variations

After processing input layerwise images using wavelet transformation and texture analysis, non-image statistical data are inserted to the MLP branch and the underlying features of image data will be retained by CNN branch. Then, the model starts to optimize hyperparameters. To evaluate the performance of trained models, three metrics of F-score, Precision, Recall and are reported using. In addition, the confusion matrix is prepared to provide more detail about the number of correct and incorrect predictions made by the trained model.

$$Precision = \frac{TP}{TP + FP} \tag{4.1}$$

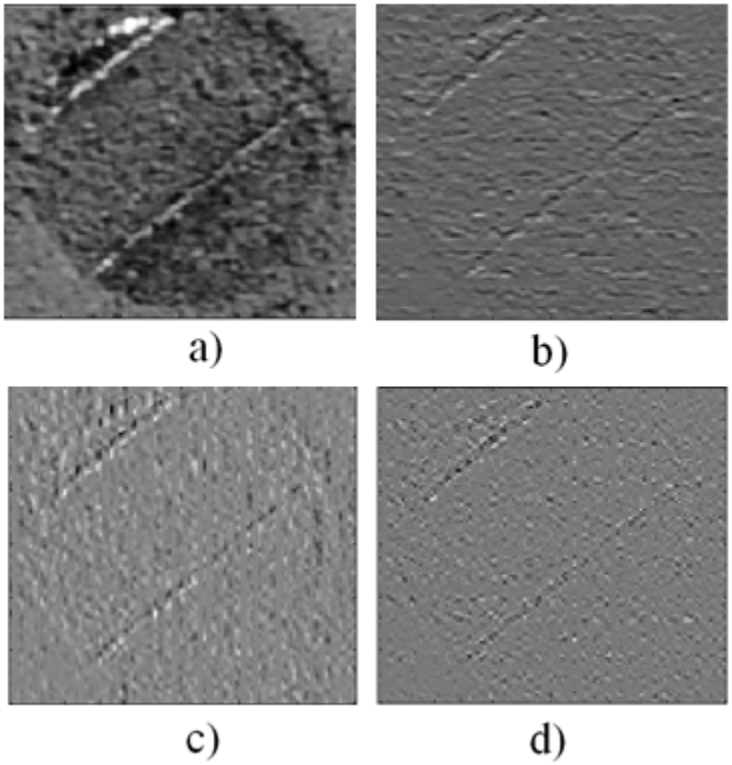

Figure 4.2. An Example of wavelet decomposition coefficient of the layerwise image. (a) Approximation detail (LL); (b) Horizontal detail (LH); (c) Vertical Detail (HL); (d) Diagonal detail (HH).

$$Recall = \frac{TP}{TP + FN} \tag{4.2}$$

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \tag{4.3}$$

$$F - score = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}} \tag{4.4}$$

where TP denotes the true positive, FP is false positive, FN is false negative and TN is true positive. We can calculate these values for each class of quality condition. For example, considering class E, recall refers to the percentage of objects in class E which

have been correctly detected by model as having quality condition E. Precision value indicates what percentage of objects that are classified by model as class E, have had this quality condition as ground truth.

The accuracy metrics is also used to show the model performance on train and test data in each epoch. However, the overall classification result will be reported by f-score since it is capable to examine the validity of the model when the data set in imbalanced. In better words, the high F-score demonstrates not only the appropriate classification power of the predictive model, but also the equal level of attention to all class labels. The hybrid model pipeline is assessed and compared by different machine learning models under various configurations. We utilized SVM, k-nearest neighbors and logistic regression (with statistical features extracted by wavelet transform and texture analysis) as well as VGG-16 (with powder bed image data) as benchmark models. Moreover, the performance of the hybrid model with different statistical features is examined.

To perform a fair comparison, the final outputs of all models are compared using 6-fold cross-validation and the average results are reported. Table. 4.1 displays the ability of different models in identifying process conditions. When entropy value, mean and standard deviation extracted by wavelet transformation are utilized as non-image inputs, the model yields 97.14% F-score. Among Machine learning models, logistics regression demonstrated the highest result with an f-score of 89.75%. Training a hybrid model with different statistical inputs, the detection performance decreased to 90.43%. Moreover, When the MLP branch removed from the hybrid model, the f-score decreased to 92.58%. Considering these results, it is proven that the MLP branch with a suitable selection of statistical input can provide informative details to empower the CNN branch. Also, this hybrid model can successfully outperform previous machine learning solutions.

Fig. 4.3 and Fig. 4.4 represents the effect of early stopping method during the training process of proposed hybrid model. As it is demonstrated for each fold, the models starts

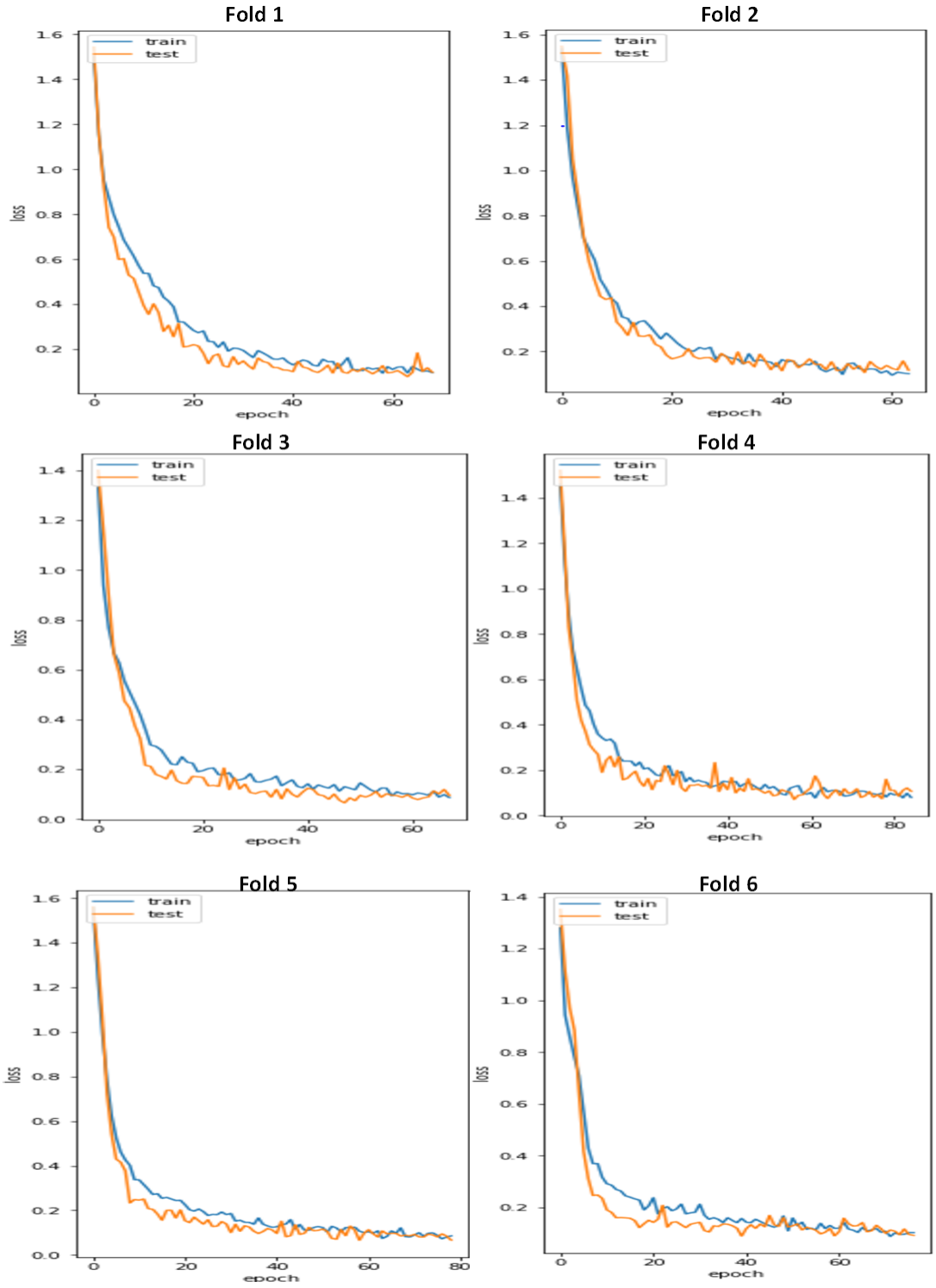

Figure 4.3. Change in loss value of train and test data during the training process of proposed hybrid model. To prevent overfitting, the early stopping method is applied to monitor the learning process and terminate the learning process if no improvement appears in test loss reduction after 20 consecutive epochs.

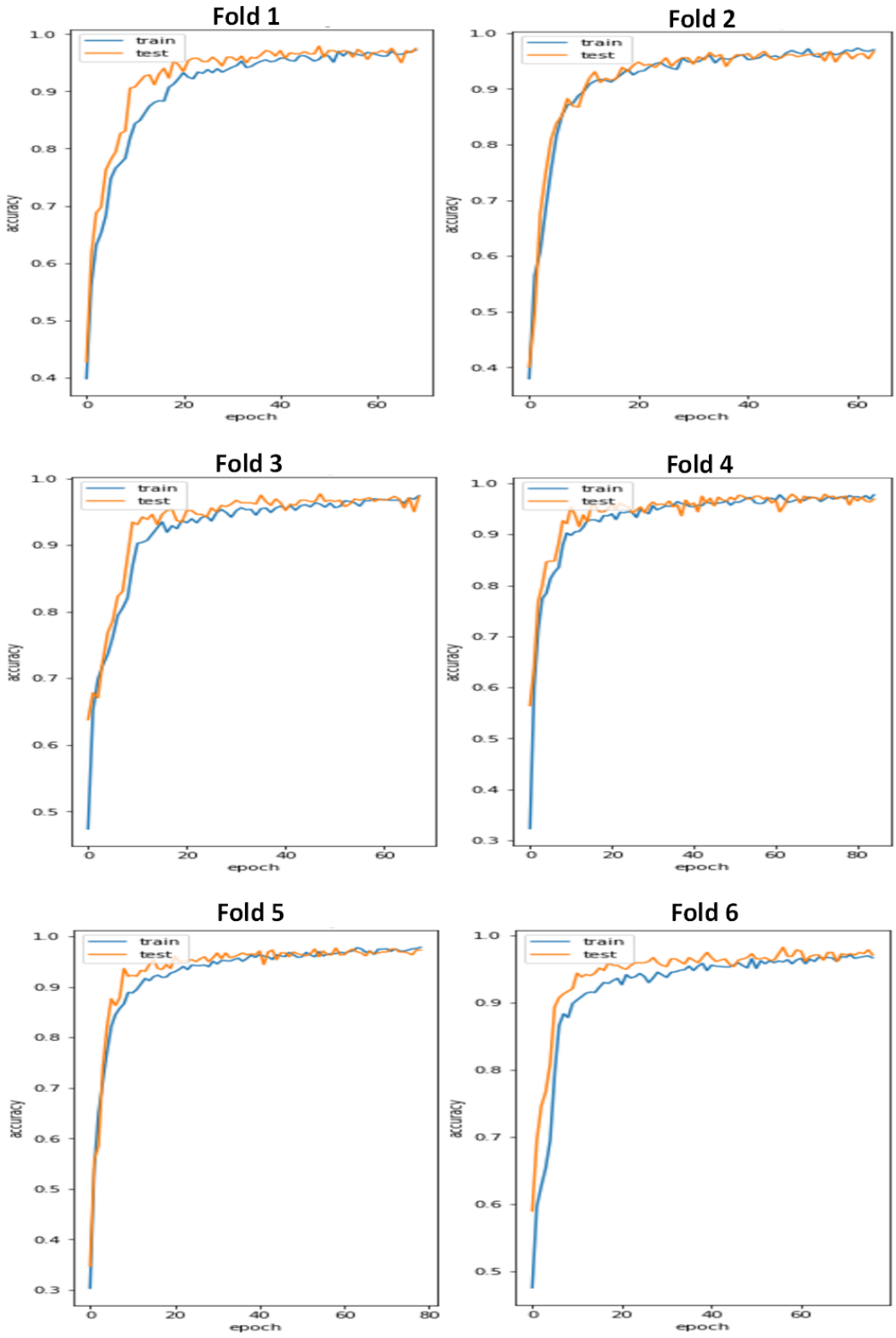

Figure 4.4. Change in accuracy value of train and test data during training process of proposed hybrid model.

Table 4.1. The process condition detection result achieved by various classifiers with different inputs

Model 1	Model Input	F-Score
Hybrid Model	Image, entropy, mean, std of wavelet	97.14%
Hybrid model	Image, mean, std of image multi-fractal, spectral graph	90.43%
VGG 16	Image	93.51%
K-nearest Neighbor	entropy, mean, std of wavelet	87.57%
SVM	entropy, mean, std of wavelet	84.71%
Logistics regression	entropy, mean, std of wavelet	89.75%

with sharp improvement in accuracy and reduced the great amount of train and test loss value. However, as the model comes closer to loss value of 0, reduction in loss value becomes difficult. As result, when the model reach to its highest capacity(i.e, the lowest loss value and the highest accuracy level), the overfitting problem appears. Here, by using early stopping, when the model couldn't decrease the test loss value after 20 epochs, the learning process was stopped to prevent the generalization error.

Fig. 4.5 illustrates six confusion matrices obtained in different folds of test data during the cross-validation process. Various sizes of data have been distributed throughout the class labels in each fold and the differences between majority and minority classes shows that the data is imbalanced. However, the class weight technique has demonstrated its power to compensate for this gap and provide the appropriate number of true positive

Table 4.2. The classification result obtained by proposed hybrid model and with test data in different folds of cross-validation technique

Fold number	F-score	Precision	Recall	Loss
1	97%	97%	97.45%	0.0976
2	96%	96%	97.72%	0.1165
3	97%	97%	97.30%	0.0954
4	97%	96%	96.91%	0.1072
5	97%	97%	97.30%	0.0660
6	96%	97%	97.10%	0.0924

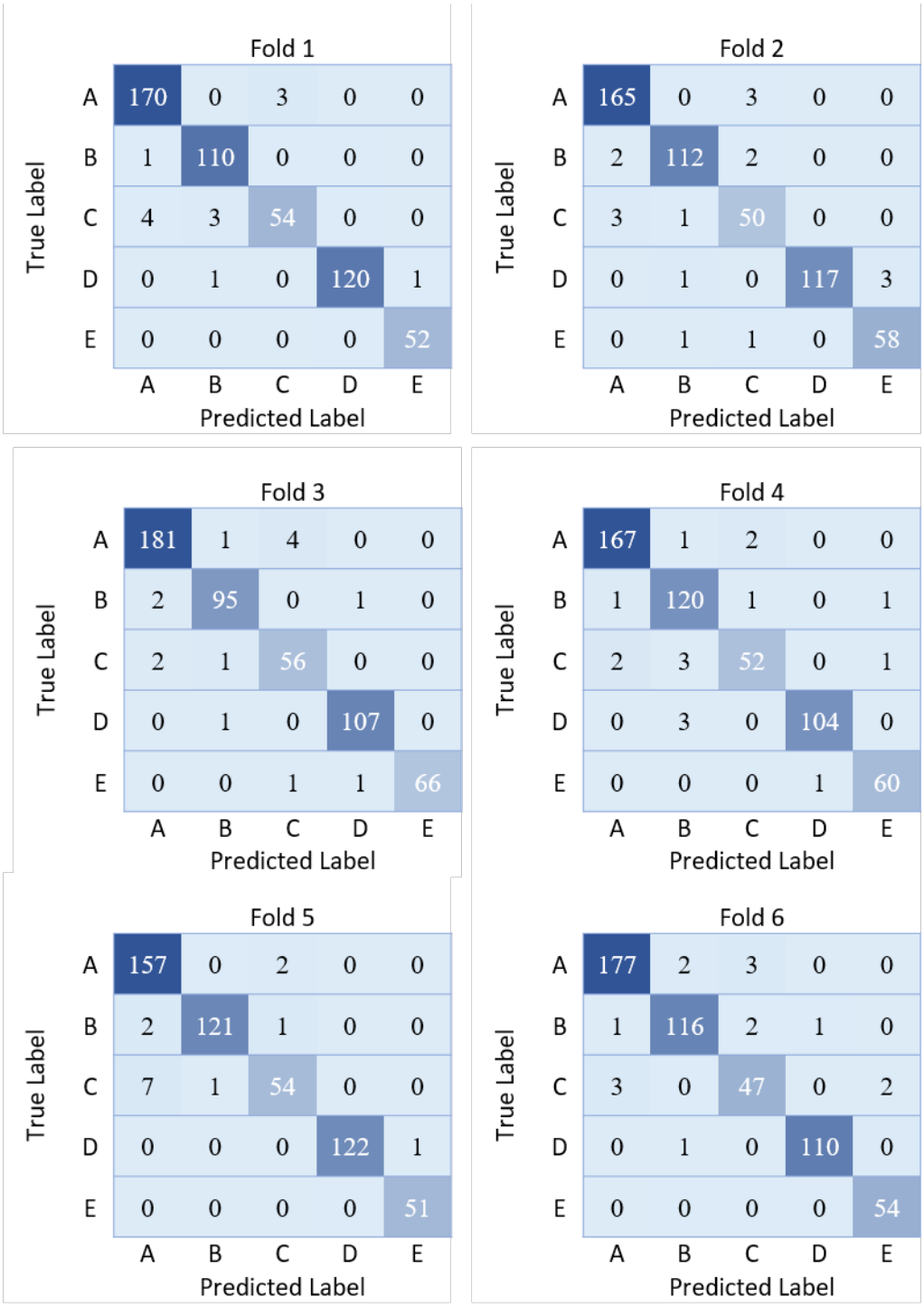

Figure 4.5. Confusion Matrix obtained by using the proposed hybrid model on the test data set. Diagonal parts in all confusion matrices comprise more concentration of data which represents the appropriate detection power of the model for all classes.

for all classes. It is worth noting that the concentration appears on the diagonal of all 6 folds, indicates that the final model performed the classification task with little bias to majority classes. These results, proved the strong potential of proposed hybrid model to monitor the AM and help mitigate defects in the process.

Chapter 5
Conclusion

This study sets forth a novel hybrid model with two heterogeneous inputs for inspection of LPBF process condition and established a new ground for the characterization of porosity defects. The method can be integrated with LBPF machine to monitor the 3D printing process. To this end, two parallel networks were merged together to form a single classification framework. The first network employs CNN and accepts the image dataset, whereas the second branch utilizes MLP and receives the non-image input of statistical features. In our experiments, a cylindrical build was produced under different process conditions and powder bed images are captured. Then, the wavelet-based features of images are extracted. It is demonstrated that wavelet decomposition coefficients effectively represent original images as they can capture different frequencies in an image with minimum loss of information. This fundamental property became even more useful when an additional texture analysis was performed on the wavelet coefficients to prepare the statistical features of the model.

Finally, the proposed framework achieved the F-score of 97.14% that is significantly higher than the state-of-the-art approaches. The results demonstrated the capability of the final architecture to address the imbalanced class difficulty and make an unbiased decision. In addition, the integration of statistical features with the powder bed image data played a paramount role to build a reliable framework to assess the build quality

during the process and provided an opportunity to make required adjustments for quality improvements in AM processes.

Chapter 6
Future Work

In this work we employed different approaches to improve the performance of proposed model. However, the size and structure of accessible data is a significant factor that can affect the reliability of a data-driven model. It worth nothing that by preparing a larger dataset of AM objects, the proposed hybrid model can achieve a higher fidelity. Besides, In future studies, to empower the classification pipeline, it would be interesting to investigate new ways to conduct more repetition of production process to gather more data of sequential layers. Then, it might be possible to assess the capacity of Recurrent Neural Network (RNN) classifier in process inspection task and consider previous layer's condition to make a final decision about current layer.

Bibliography

[1] Wu, D., S. Liu, L. Zhang, J. Terpenny, R. X. Gao, T. Kurfess, and J. A. Guzzo (2017) "A fog computing-based framework for process monitoring and prognosis in cyber-manufacturing," *J. Manuf. Syst.*, **43**, pp. 25–34.

[2] Cai, Y., Y. Wang, and M. Burnett (2020) "Using augmented reality to build digital twin for reconfigurable additive manufacturing system," *J. Manuf. Syst.*

[3] Mittal, S., M. A. Khan, D. Romero, and T. Wuest (2018) "A critical review of smart manufacturing Industry 4.0 maturity models: Implications for small and medium-sized enterprises (SMEs)," *J. Manuf. Syst.*, **49**, pp. 194–214.

[4] Yi, L., C. Glässner, and J. C. Aurich (2019) "How to integrate additive manufacturing technologies into manufacturing systems successfully: A perspective from the commercial vehicle industry," *J. Manuf. Syst.*, **53**, pp. 195–211.

[5] Zhang, B., A. Goel, O. Ghalsasi, and S. Anand (2019) "CAD-based design and pre-processing tools for additive manufacturing," *J. Manuf. Syst.*, **52**, pp. 227–241.

[6] Khairallah, S. A., A. T. Anderson, A. Rubenchik, and W. E. King (2016) "Laser powder-bed fusion additive manufacturing: Physics of complex melt flow and formation mechanisms of pores, spatter, and denudation zones," *Acta Mater.*, **108**, pp. 36–45.

[7] Mani, M., B. Lane, A. Donmez, S. Feng, S. Moylan, and R. Fesperman (2015) "Measurement Science Needs for Real-time Control of Additive Manufacturing Powder Bed Fusion Processes„" *US Department of Commerce, National Institute of Standards and Technology.*

[8] Carter, L. N., X. Wang, N. Read, R. Khan, M. Aristizabal, K. Essa, and M. M. Attallah (2016) "Process optimisation of selective laser melting using energy density model for nickel based superalloys," *Mater. Sci. Technol.*, pp. 1–5.

[9] Yang, H., S. T. Bukkapatnam, and R. Komanduri (2007) "Nonlinear adaptive wavelet analysis of electrocardiogram signals," *Phys. Rev. E*, **76**(2), p. 026214.

[10] Haralick, R. M., K. Shanmugam, and I. Dinstein (1973) "Textural Features for Image Classification," *IEEE Trans. Syst. Man. Cybern.*, **SMC-3**(6), pp. 610–621.

[11] Grasso, M. and B. M. Colosimo (2017) "Process defects and in situ monitoring methods in metal powder bed fusion: a review," *Meas. Sci. Technol.*, **28**(4), p. 044005.

[12] Tapia, G. and A. Elwany (2014) "A Review on Process Monitoring and Control in Metal-Based Additive Manufacturing," *J. Manuf. Sci. Eng.*, **136**(6).

[13] Lu, Y. and Y. Wang (2018) "Monitoring temperature in additive manufacturing with physics-based compressive sensing," *J. Manuf. Syst.*, **48**, pp. 60–70.

[14] Cerniglia, D., M. Scafidi, A. Pantano, and J. Rudlin (2015) "Inspection of additive-manufactured layered components," *Ultrasonics*, **62**, pp. 292–298.

[15] Rieder, H., A. Dillhöfer, M. Spies, J. Bamberg, and T. Hess (2015) "Ultrasonic online monitoring of additive manufacturing processes based on selective laser melting," pp. 184–191.

[16] Zhao, C., K. Fezzaa, R. W. Cunningham, H. Wen, F. De Carlo, L. Chen, A. D. Rollett, and T. Sun (2017) "Real-time monitoring of laser powder bed fusion process using high-speed X-ray imaging and diffraction," *Sci. Rep.*, **7**(1), p. 3602.

[17] Craeghs, T., S. Clijsters, E. Yasa, F. Bechmann, S. Berumen, and J.-P. Kruth (2011) "Determination of geometrical factors in Layerwise Laser Melting using optical process monitoring," *Opt. Lasers Eng.*, **49**(12), pp. 1440–1446.

[18] Craeghs, T., S. Clijsters, J.-P. Kruth, F. Bechmann, and M.-C. Ebert (2012) "Detection of Process Failures in Layerwise Laser Melting with Optical Process Monitoring," *Phys. Procedia*, **39**, pp. 753–759.

[19] Seifi, S. H., W. Tian, H. Doude, M. A. Tschopp, and L. Bian (2019) "Layer-Wise Modeling and Anomaly Detection for Laser-Based Additive Manufacturing," *J. Manuf. Sci. Eng.*, **141**(8).

[20] Yao, B. and H. Yang (2018) "Constrained Markov Decision Process Modeling for Sequential Optimization of Additive Manufacturing Build Quality," *IEEE Access*, **6**, pp. 54786–54794.

[21] Imani, F., A. Gaikwad, M. Montazeri, P. Rao, H. Yang, and E. Reutzel (2018) "Layerwise in-process quality monitoring in laser powder bed fusion," in *ASME 2018 13th International Manufacturing Science and Engineering Conference*, American Society of Mechanical Engineers Digital Collection.

[22] Khanzadeh, M., S. Chowdhury, M. Marufuzzaman, M. A. Tschopp, and L. Bian (2018) "Porosity prediction: Supervised-learning of thermal history for direct laser deposition," *J. Manuf. Syst.*, **47**, pp. 69–82.

[23] Wang, T., T.-H. Kwok, C. Zhou, and S. Vader (2018) "In-situ droplet inspection and closed-loop control system using machine learning for liquid metal jet printing," *J. Manuf. Syst.*, **47**, pp. 83–92.

[24] Wang, J., Y. Ma, L. Zhang, R. X. Gao, and D. Wu (2018) "Deep learning for smart manufacturing: Methods and applications," *J. Manuf. Syst.*, **48**, pp. 144–156.

[25] Imani, F., R. Chen, E. Diewald, E. Reutzel, and H. Yang (2019) "Deep Learning of Variant Geometry in Layerwise Imaging Profiles for Additive Manufacturing Quality Control," *J. Manuf. Sci. Eng.*, **141**(11).

[26] Zhang, Y., G. S. Hong, D. Ye, K. Zhu, and J. Y. Fuh (2018) "Extraction and evaluation of melt pool, plume and spatter information for powder-bed fusion AM process monitoring," *Mater. Des.*, **156**, pp. 458–469.

[27] Shevchik, S., C. Kenel, C. Leinenbach, and K. Wasmer (2018) "Acoustic emission for in situ quality monitoring in additive manufacturing using spectral convolutional neural networks," *Addit. Manuf.*, **21**, pp. 598–604.

[28] Scime, L. and J. Beuth (2018) "A multi-scale convolutional neural network for autonomous anomaly detection and classification in a laser powder bed fusion additive manufacturing process," *Addit. Manuf.*, **24**, pp. 273–286.

[29] Imani, F., A. Gaikwad, M. Montazeri, P. Rao, H. Yang, and E. Reutzel (2018) "Process Mapping and In-Process Monitoring of Porosity in Laser Powder Bed Fusion Using Layerwise Optical Imaging," *J. Manuf. Sci. Eng.*, **140**(10).

[30] Daubechies, I. and C. Heil (1992) "Ten Lectures on Wavelets," *Comput. Phys.*, **6**(6), p. 697.

[31] Kamarthi, S. V., S. R. Tirupatikumara, and P. H. Cohen (1995), "Wavelet representation of Acoustic Emission in turning process," .
URL `https://pennstate.pure.elsevier.com/en/publications/wavelet-representation-of-acoustic-emission-in-turning-process`

[32] Rioul, O. and P. Duhamel (1992) "Fast algorithms for discrete and continuous wavelet transforms," *IEEE Trans. Inf. Theory*, **38**(2), pp. 569–586.

[33] .

[34] Shannon, C. E. (1948) "A Mathematical Theory of Communication," *Bell Syst. Tech. J.*, **27**(3), pp. 379–423.

[35] Materka, A., A. Materka, and M. Strzelecki (1998) "Texture analysis methods – a review," *Inst. Electron. Tech. Univ. LODZ.*

[36] Liu, S. and W. Deng (2015) "Very deep convolutional neural network based image classification using small training sample size," in *2015 3rd IAPR Asian Conf. Pattern Recognit.*, IEEE, pp. 730–734.

[37] Xu, B., N. Wang, T. Chen, and M. Li "Empirical evaluation of rectified activations in convolutional network," *arXiv preprint*.
URL http://arxiv.org/abs/1505.00853

[38] Ioffe, S. and C. Szegedy (2015) "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," in *Proceedings of the 32nd International Conference on Machine Learning*, PMLR, pp. 448–456.
URL http://proceedings.mlr.press/v37/ioffe15.html

[39] Srivastava, N., G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov (2014) "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *J. Mach. Learn. Res.*, **15**(1), p. 1929–1958.
URL http://jmlr.org/papers/v15/srivastava14a.html

[40] Rumelhart, D. E., G. E. Hinton, and R. J. Williams (1986) "Learning representations by back-propagating errors," *Nature*, **323**(6088), pp. 533–536.

[41] Caruana, R., S. Lawrence, and L. Giles (2000) "Overfitting in Neural Nets: Backpropagation, Conjugate Gradient, and Early Stopping," in *Proceedings of the 13th International Conference on Neural Information Processing Systems*, p. 381–387.
URL https://dl.acm.org/doi/10.5555/3008751.3008807

[42] Abadi, M., P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. (2016) "Tensorflow: A system for large-scale machine learning," in *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, pp. 265–283.