A Machine Learning Approach to Accelerated Prediction of Statistics of Microstructures Produced by the Laser Powder Bed Fusion Process

By

Mason Avery Jones
Dissertation

Submitted in partial satisfaction of the requirements for the degree of

DOCTOR OF PHILOSOPHY

In

Mechanical and Aerospace Engineering

in the

OFFICE OF GRADUATE STUDIES

of the

UNIVERSITY OF CALIFORNIA

DAVIS

Approved:

Jean-Pierre Delplanque, Chair

Michael Hill

Jeremy Mason

Committee in Charge

2026

This work was supported by the Laboratory Directed Research and Development program at Sandia National Laboratories, a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia LLC, a wholly owned subsidiary of Honeywell International Inc. for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Contents

List of Figures 6
Abstract 15
Acknowledgements 16
Chapter 1 Introduction 1
1.1 Background and Motivation 1
1.2 Laser Powder Bed Fusion 4
1.3 LPBF Microstructures 9
1.4 Modeling And Simulation of LPBF 11
1.4.1 Thermal Modeling 11
1.4.2 Microstructure Modeling 13
1.5 Machine Learning 15
1.5.1 Neural Networks 17
1.5.2 Machine Learning Surrogate Modeling 21
1.6 Research Goals and Dissertation Structure 22
Chapter 2 Surrogate Model Framework 24
2.1 Introduction 24
2.2 Framework For Model Development 24
2.3 Thermal Model and Data 31
2.3.1 Thermal Characteristics for Model Input 32
2.4 Microstructure Simulations and Data 37
2.4.1 Voxel-by-Voxel Microstructure Statistics 40
2.4.2 A Better Way to Visualize Our Data 48
2.5 Framework Final Words 52
Chapter 3 The Surrogate Model 53
3.1 Introduction 53
3.2 Model Architecture 53
3.3 Training Dataset 57
3.4 Data Regularization and Augmentation 63
3.5 Model Training 67
3.6 Understanding The Mean Squared Error 70
3.6.1 Method 71

3.6.2 Results 72
3.7 Model Optimization 74
3.8 Conclusions 78
Chapter 4 Surrogate Model Accuracy 79
4.1 Introduction 79
4.2 Initial Model Results 79
4.3 Final Model Results 83
4.4 Influence of Ensemble Size on Model Accuracy 89
4.4.1 Method 90
4.4.2 Results 90
4.5 Predictive Power Analysis 97
4.5.1 Methods 97
4.5.2 Limitations 99
4.5.3 Results 101
4.5.4 Conclusion 121
4.6 Supplementing the Training Data with Small Ensembles 121
4.6.1 Discussion 122
4.6.2 Results 123
4.7 Conclusions 126
4.7.1 A Discussion on The Importance of Accuracy 127
Chapter 5 Understanding The Surrogate Model 129
5.1 Introduction 129
5.2 Measuring the Importance of the Selected Thermal Characteristics 129
5.2.1 Method 130
5.2.2 Intermediate Results 133
5.2.3 Results 139
5.2.4 Takeaways 148
5.3 Coverage of the Thermal Characteristic Space 149
5.3.1 Simplified Thermal Characteristic Space 150
5.3.2 Dimensionality Reduction 153
5.3.3 Takeaways 158
Chapter 6 Conclusions 159
6.1 Future Work 161

Appendix A Table of All Case Parameters 163
Appendix B Table of Hyperparameter Optimization Options 164
References 165

List of Figures

Figure 1 – Example applications of metal AM that would not be practical with conventional manufacturing techniques. Cross section of an additively manufactured aerospike engine designed by Leap 71 [15,22], reprinted with permission from Leap 71. 3

Figure 2 – Schematic of the LPBF process adapted from [24] under the Creative Commons Attribution-Share Alike 3.0 Unported license [25]. Left) Cross-sectional diagram depicting the major components of the LPBF process. Right) Detail view depicting a cross-section of the powder bed as it is processed into a part. 4

Figure 3 – Top) Delamination caused by residual stresses, from [35], reprinted under Creative Commons. Bottom) Delamination of Ti6Al4V parts from their support structure during manufacturing, from [30], reprinted with permission of Informa UK Limited, trading as Taylor & Francis Group. 6

Figure 4 – Illustration of a raster scanning path for a square, with the laser switching off at turns. Top) The first layer. Bottom) One of the scan strategies which can be employed on the layers of a print: the direction is rotated by 90 degrees with each layer. 7

Figure 5 – Flowchart illustrating the conceptual process optimization loop. Microstructure simulations are used to inform the microstructure surrogate model, without the main optimization loop relying on them. 9

Figure 6 – Illustration of the behavior of the laser light source for the LPBF process, as simulated by a high-fidelity multi-physics simulation with ray tracing. From Khairallah et al [83], reprinted with permission from AAAS. 12

Figure 7 – Flowchart illustrating the conceptual process optimization loop. Microstructure simulations are used to inform the microstructure surrogate model, without the main optimization loop relying on them. 24

Figure 8 – Framework for the creation of the surrogate microstructure model. 25

Figure 9 – Data-wise comparison between our model and an image classification network 27

Figure 10 – Illustration of the relationship between the thermal data, simulated microstructures, and microstructure statistics. Top) The relationship between the moving window of the thermal inputs, the point of interest of the microstructures, and the statistics provided at that point by an ensemble of simulations. Bottom) Illustration of how moving the window changes the point of interest and the corresponding statistics. 29

Figure 11 – A snapshot illustrating some of the complexities of the thermal histories experienced by points during the LPBF process, with repeated heating and cooling cycles. The actual process results in significantly more heating and cooling cycles. 34

Figure 12 – Plot showing the relationship between temperature and rate of grain growth, with the location of the "knee" labeled. 36
Figure 13 – Number of times each voxel on the top surface of a 1-layer raster scan has melted. 37
Figure 14 – Example microstructure of cross section of a 1mm cube, including the unprocessed region around the cube. Note the large columnar grains in the central region, and the smaller grains around the edges, with the random noise around the edges random noise representing unprocessed powder. 39
Figure 15 – Plots showing the diameters of each grain (left) and the corresponding bin number of each grain (right) for the top surface of a one-layer print (top) and the cross section of a 1mm cube(bottom). 43
Figure 16 – EBSD images of LPBF microstructures by Roehling et al, which appear to show small grains entrapped in larger grains [138], reproduced under Creative Commons [139]. 44
Figure 17 – Relative probabilities that each voxel belongs to a 32-40um grain on the top of a 1-layer raster scan (left), and a 240-248um grain on the cross section of a 1mm cube (right). 44
Figure 18 – Grain size probability fields of the cross section of a 1mm cube, showing the relative probability that each voxel belongs to a 64-72um, 160-168um, and 272-280um grain respectively. 45
Figure 19 – Plot showing the apparent repetitive pattern of the frequency of occurrence of 32-40µm grains on the top surface of a 1-layer raster scan across two thousand simulations (top right). The plot on the left shows the actual magnitude of the frequency for each column of the image, with each line representing a column, and the plot on the bottom shows the same but with rows. 47
Figure 20 – Comparison between the cross section of a single microstructure and the equivalent statistics. Left) Statistics from an ensemble of 5 microstructure simulations. Center) Statistics from the full ensemble of 198 simulations. Right) Single simulated microstructure for comparison. 48
Figure 21 – Plots showing the probability of medium sized grains (left) and large grains (right) for the cross section of a many-layer print, where each of the 3 base-colors (RGB) represent adjacent groupings of 3 adjacent bins each. These smaller groupings of bins provide a more detailed look at the variation present across the part; however, for the purposes of simplicity, we do not explore such representations in this work. 49
Figure 22 – Three color plot of the grain size probabilities for an ensemble of SPPARKS simulations, before (left) and after (right) adjusting the cutoff point to be 176µm larger using the 99.5^{th} percentile. 50
Figure 23 – Color wheel for reading statistics from tri-bin plot, where the 3-value labels indicate the probability associated with each of the three bins, along with a diagram indicating reference points on the color wheel. 50
Figure 24 – Red component of the colorwheel. 51

Figure 25 –Network diagrams showing original ConvNeXt architecture (left), and the surrogate model architecture (right). Where N indicates how many blocks occur in each stage of the model. 55
Figure 26 – Plot of the powers and velocities used for training data generation, other parameters such as the hatch spacing and layer thickness were varied independently. 58
Figure 27 – Illustration depicting the rotation of the thermal gradients around the z-axis. The blue vector illustrates how the correct rotation changes the sign and magnitude of the components of the gradient with respect to the global axes. The red arrow depicts the incorrect global direction of the thermal gradient achieved with a naive rotation. 64
Figure 28 – Illustration of the rules applied to the thermal gradients during when rotated. The global axes depict the plane on which the rotation occurs and the directions of the global axes. The axes within the boxes depict the initial positive directions of these axes, and the orientation of these axes after their corresponding rotations, but before corrections. The equalities depict the swaps which need to be made after the tensor has been rotated, where the left side of the equality is the global axis, and the right is the data which needs to be moved to that axis. 65
Figure 29 – Plot of the model's MSE against test data vs the percentage of the full dataset used for training the model. 69
Figure 30 – Flowchart depicting the experimental procedure. We start with a distribution obtained from an ensemble, which we will call our ground truth. We then sample two distributions from the ground truth and calculate the MSE between all 3 pairings. We then repeat this process for the distribution corresponding to all voxels in the case. 72
Figure 31 – Whisker plot showing the mean squared error between the sampled distributions and the full distribution. Shows the decrease in the mean squared error as the sample size increases. 73
Figure 32 – Whisker plot showing the mean squared error between two sampled distributions. Shows the decrease in the mean squared error as the sample size increases. 74
Figure 33 – Plot showing results of all the trials of the network optimization process, where the objective is to minimize the loss and maximize the performance. The best trials are shown in red, and the shade of the points corresponds to the trial number. 77
Figure 34 – Probability that each voxel in the top surface of a 1-layer raster scan printed with a 100W laser moving at 1m/s (P100V100Case1) belongs to a 30-32µm diameter grain, with results of the 144 simulation SPPARKS ensemble used for training on the left and the surrogate model's predictions on the right. The irregular striated patterns in the ensemble data are due to the stochastic nature of the SPPARKS model. 81

Figure 35 – Probability that each voxel in the top surface of a 1-layer raster scan printed with a 100W laser moving at 1m/s (P100V100Case1) belongs to a 30-32µm diameter grain, with results of the larger 1188 simulation SPPARKS ensemble used for testing on the left and the surrogate model's predictions on the right. Note the significant decrease in appearance of the irregular striated patterns with the larger ensemble, which more closely matches the model's predictions. 82
Figure 36 – Prediction from the final version of the model, for the same case presented in the previous figure(P100V100Case1), using the 3-color plotting method presented in Section 2.4.2. Red:0-16µm, Green:16-32µm, Blue:32-48µm. We continue to see that the model's predicted probability maps are smoother than the training data, and achieve a very high accuracy. 84
Figure 37 – Prediction from the final version of the model, for the same case presented in the previous figure (P100V100Case1), compared against a very large ensemble of 2000 simulations. Red:0-16µm, Green:16-32µm, Blue:32-48µm. We once again see that the measured accuracy of the model's predictions improves as we increase the size of the ensemble against which we measure. 85
Figure 38 – The top surface and cross section of a 15-layer print, P100V100Case2, showing good agreement between the model's predictions and the corresponding ensemble data. Red:0-40µm, Green:40-80µm, Blue:80-128µm. 86
Figure 39 –Cross sections from two different 30-layer cases, both of which exhibit banding artifacts in the model's predictions, though there is no significant increase in the MSE around these artifacts. (Top) P225V72 Red:0-96µm, Green:96-192µm, Blue:192-296µm. (Bottom) P250V75 Red:0-128µm, Green:128-256µm, Blue:256-400µm.) 87
Figure 40 – Cross section from the single case included in our dataset which used a 90-degree rotation of the laser scan path between layers. Red:0-104µm, Green:104-208µm, Blue:208-320µm. 88
Figure 41 – Accuracy of the model vs the largest available SPPARKS ensembles as the size of the SPPARKS ensembles used for training changes. The largest ensemble varies case to case but is generally around 200 simulations. The accuracy of the training ensembles vs the large ensembles is shown for comparison; shows that the model outperforms the data it is trained on. At an ensemble size of 75, the model and training data have similar accuracies vs the largest ensembles 91
Figure 42 – Accuracy of the model vs the largest available ensembles, as the training ensemble size grows, zoomed in on the region in which the accuracy appeared to plateau. While the rate of improvement does decrease significantly, the model's accuracy does continue to improve in this region. 92

Figure 43 – Model's prediction for the P225V72 case when trained on 5 simulation ensembles, compared against the full ensemble of 220 simulations. Red:0-96µm, Green:96-192µm, Blue:192-296µm. 93
Figure 44 – Model's prediction for the P225V72 case when trained on 10 simulation ensembles, compared against the full ensemble of 220 simulations. Red:0-96µm, Green:96-192µm, Blue:192-296µm. 94
Figure 45 – Model's prediction for the P225V72 case when trained on 25 simulation ensembles, compared against the full ensemble of 220 simulations. Red:0-96µm, Green:96-192µm, Blue:192-296µm. 95
Figure 46 – Model's prediction for the P225V72 case when trained on 50 (top), 75 (middle) and 100 (bottom) simulation ensembles, compared against the full ensemble of 220 simulations. Red:0-96µm, Green:96-192µm, Blue:192-296µm. 96
Figure 47 – Plot showing the process parameters used for both training and testing the model. For a given scenario, all data except the corresponding testing set was used for training the model. 98
Figure 48 – Stacked histograms showing the distributions of the thermal characteristics across all voxels in several cases. Each case is represented by a different color. This is a simplified view that does not account for the fact that each voxel has a specific pairing of the nine thermal characteristics. 100
Figure 49 – The model's predictions for the top surface of all three interpolation cases: Top) P95V95 Middle) P95V97.5 Bottom) P102.5V92.5. All three predictions are in good agreement with the results from the simulation-based ensembles and have a low MSE. Red:0-16µm, Green:16-32µm, Blue:32-48µm. 102
Figure 50 – Model's predicted probabilities for the top surface of the P100V100H45 case, a 1-layer raster scan with a hatch spacing of 45µm. The model's predictions are reasonably accurate when compared to our test data, but it does not capture some of the features around the turns. Red:0-16µm, Green:16-32µm, Blue:32-48µm. 103
Figure 51 – Model's predicted probabilities for the top surface of the P100V100H40 case, a 1-layer raster scan with a hatch spacing of 40µm. The model's predictions are slightly less accurate than for the 45µm case, but the discrepancies in the vicinity of the turns are less obvious. Red:0-16µm, Green:16-32µm, Blue:32-48µm. 104
Figure 52 – Top surface of the 3rd layer of the extrapolation case P130V110. Red:0-24µm, Green:24-48µm, Blue:48-80µm. 105
Figure 53 Top surface of the 3rd layer of the extrapolation case P135V112. Red:0-24µm, Green:24-48µm, Blue:48-80µm. 105
Figure 54 – Top surface of the 3rd layer of the extrapolation case P140V115. Red:0-24µm, Green:24-48µm, Blue:48-80µm. 105

Figure 55 – Top surface of the 4th (top) and 5th (bottom) layers of the extrapolation case P130V110. Top) Red:0-32µm, Green:32-64µm, Blue:64-112µm. Bottom) Red:0-40µm, Green:40-80µm, Blue:80-128µm..... 106
Figure 56 – Top surface of the 4th (top) and 5th (bottom) layers of the extrapolation case P135V112. Top) Red:0-32µm, Green:32-64µm, Blue:64-112µm. Bottom) Red:0-40µm, Green:40-80µm, Blue:80-136µm..... 107
Figure 57 – Top surface of the 4th (top) and 5th (bottom) layers of the extrapolation case P140V115. Top) Red:0-32µm, Green:32-64µm, Blue:64-112µm. Bottom) Red:0-40µm, Green:40-80µm, Blue:80-136µm..... 108
Figure 58 – Top surface of the 1st (top) and 2nd (bottom) layers of the P225V72.5 sparse interpolation case. Top) Red:0-24µm, Green:24-48µm, Blue:48-80µm. Bottom) Red:0-32µm, Green:32-64µm, Blue:64-96µm. 109
Figure 59 – Top surface of the 3rd layer of the P225V72.5 sparse interpolation case. The model significantly overpredicts the size of the largest likely grains, by 248µm, resulting in the majority of every distribution falling in the smallest (red) super-bin. Red:0-112µm, Green:112-224µm, Blue:224-352µm..... 110
Figure 60 – Top surface of the 3rd layer of the P225V72.5 sparse interpolation case. Determining the upper cutoff using the 0.5% method results in a much more reasonable representation, with the model overpredicting the grain size by just 16µm, though some distributions are cut off. Red:0-40µm, Green:40-80µm, Blue:80-120µm..... 110
Figure 61 – Top surface of the 4th (top) and 5th (bottom) layers of the P225V72.5 sparse interpolation case, with the upper cutoff for visualization calculated using the 99.5% method. The model overpredicts the maximum size of likely grains by 96µm and 72µ respectively. Top) Red:0-64µm, Green:64-128µm, Blue:128-208µm. Bottom) Red:0-64µm, Green:64-128µm, Blue:128-200µm..... 111
Figure 62 – Top surface of the 4th (top) and 5th (bottom) layers of the P225V72.5 sparse interpolation case, with the upper cutoff for visualization calculated using the 0.5% method. The model overpredicts the maximum size of likely grains by 16µm for both cases. Top) Red:0-40µm, Green:40-80µm, Blue:80-128µm. Bottom) Red:0-48µm, Green:48-96µm, Blue:96-144µm. 112
Figure 63 – Cross section of the full 30-layer P225V72.5 sparse interpolation case. The model overpredicts the maximum likely grain size by 96µm when the 99.5% cutoff is used. Red:0-128µm, Green:128-256µm, Blue:256-392µm. 113
Figure 64 – Cross section of the full 30-layer P225V72.5 sparse interpolation case. Using the 0.5% cutoff results in the model's prediction looking more reasonable, though it still overpredicts the maximum likely grain size by 72µm. Red:0-112µm, Green:112-224µm, Blue:224-352µm. 113

Figure 65 – Cross section of the P76V104R90 90-degree rotation case, using the 99.5% cutoff (top) and the 0.5% cutoff (bottom). The model significantly underpredicts the size of the largest likely grains, which results in a large black area in the ensemble predictions when the 0.5% cutoff is used. Top) Red:0-104µm, Green:104-208µm, Blue:208-320µm. Bottom) Red:0-48µm, Green:48-96µm, Blue:96-144µm. 115
Figure 66 – Top surface of the P76V104R90 90-degree rotation case, using the 99.5% cutoff (top) and the 0.5% cutoff (bottom). The model fails to accurately predict the direction of some of the patterns, seemingly overemphasizing the importance of the previous layer's scan direction. Top) Red:0-104µm, Green:104-208µm, Blue:208-320µm. Bottom) Red:0-48µm, Green:48-96µm, Blue:96-144µm. 116
Figure 67 Top surface (top) and cross section (bottom) of the P90V110R45 45-degree rotation case. Red:0-56µm, Green:56-112µm, Blue:112-176µm. 117
Figure 68 – Cross section of the 60-layer P80V130 (Far Extrapolation) case. The model notably overpredicts the size of likely grains by 104µm, and has significant banding artifacts with a high likelihood of smaller grains. Red:0-120µm, Green:120-240µm, Blue:240-368µm. 118
Figure 69 – Top surface of the 60-layer P80V130 (Far Extrapolation) case. The model notably underpredicts the size of the grains along the top surface, with the exception of a small region with very large grains. Red:0-120µm, Green:120-240µm, Blue:240-368µm. 118
Figure 70 – Top surface of the 10-layer P80V130 (Far Extrapolation) case, plotted using the 99.5% cutoff (top) and the drop below 0.5% cutoff (bottom). Using the 99.5% cutoff results in almost the entire part being binned in the "small" grain size red super-bin due to the predicted likelihood of very large grains in the corner of the part. Using the less refined cutoff method of finding the last bin to cross below the 0.5% threshold shows a much clearer picture, though it results in a region with "0%" probability of anything. Top) Red:0-104µm, Green:104-208µm, Blue:208-320µm. Bottom) Red:0-40µm, Green:40-80µm, Blue:80-136µm. 119
Figure 71 – Cross-section of the 60-layer P80V130 (Far Extrapolation) case as predicted by the first model created for testing this scenario, plotted using the 0.5% cutoff. The model significantly underpredicts the size of likely grains, resulting in the ensemble-based statistics having a large "0%" probability region due to the grains being larger than the cutoff. The magnitude of this under-prediction is unknown. Red:0-32µm, Green:32-64µm, Blue:64-120µm. 120
Figure 72 – Cross-section of the 60-layer P80V130 (Far Extrapolation) case as predicted by the second version of the model created for this scenario, plotted using the same cutoff scheme as was used in the previous figure with the first model. Red:0-112µm, Green:112-224µm, Blue:224-336µm. 121

Figure 73 – Cross section showing the model's prediction for the P225V72.5 case when not trained on it, i.e. an ensemble size of zero (top) and when all training data is 5-simulation ensembles (bottom). Despite significantly overpredicting the size of the largest likely grains, the model has a much better overall accuracy in the predictive case than in the 5-ensemble dataset case. Top) Red:0-128µm, Green:128-256µm, Blue:256-392µm. Bottom) Red:0-96µm, Green:96-192µm, Blue:192-296µm. 123
Figure 74 – Cross section of the P225V72.5 case, showing the model's prediction when a 5-simulation ensemble for this case is added to the training data (top) and the 5-simulation ensemble itself (bottom). The model's prediction has a lower MSE than the 5-simulation ensemble in the training data, when compared to the full ensemble for this case. Visualized using the 0.5% cutoff. Top) Red:0-88µm, Green:88-176µm, Blue:176-272µm. Bottom) Red:0-64µm, Green:64-128µm, Blue:128-200µm. 124
Figure 75 Cross section of the P225V72.5 case, showing the model's prediction when a 10-simulation ensemble for this case is added to the training data, visualized using the 0.5% cutoff. Red:0-88µm, Green:88-176µm, Blue:176-264µm. 125
Figure 76 – Plots showing the accuracy of the model when a 25-simulation ensemble is used as training data for this test case. 126
Figure 77 – Illustrations from Selvaraju et al [161] depicting saliency maps generated using several methods, reprinted with permission from Springer Nature. 131
Figure 78 – A volumetric render of the deviation of the melt count for every voxel in the input window, measured across all 77 million input windows in the dataset. Left) The whole input window. Right) The cross-section of the input window, which reveals more information. 133
Figure 79 – Deviation of the temperature gradient in the z-direction. 134
Figure 80 – Deviation of the time spent above the nucleation temperature. 134
Figure 81 – The mean Jacobian of the surrogate model's predictions with respect to the nucleation time, averaged across all grain size bins and model inputs. 135
Figure 82 – The variation of the Jacobian with respect to the nucleation time, plotted using a linear and log scale to better highlight small variations. 137
Figure 83 – The variation of the nucleation time Jacobian, cut along the same plane as the mean. The variation does not mimic the distinct pattern seen in the mean. 137
Figure 84 – Deviation of the maximum reheat temperature Jacobian, which clearly illustrates the influence of the first down-sampling layer of the model architecture. 138
Figure 85 – Deviation of the Jacobians with respect to the three directional temperature gradients. The model assigns the most importance to the neighboring voxels in the same direction as the gradient components, indicating that it has learned the directionality of the three components. 139

Figure 86 – Mean and variance (left and right respectively) of the Jacobians with respect to the Nucleation time, scaled by the variance of the Nucleation time..... 141
Figure 87 – Mean and variance (left and right respectively) of the Jacobians with respect to the Solidus time, scaled by the variance of the Solidus time..... 142
Figure 88 – Mean and variance (left and right respectively) of the Jacobians with respect to the Annealing time, scaled by the variance of the Annealing time. 142
Figure 89 – Variances of the nucleation time and annealing time thermal characteristics. The maximum variance of the annealing time is 6.3 times that of the nucleation time..... 143
Figure 90 – Mean and variance (left and right respectively) of the Jacobians with respect to the Maximum Reheat Temperature, scaled by the variance of the Maximum Reheat Temperature..... 144
Figure 91 – Mean Jacobian with respect to the Maximum Reheat Temperature, scaled by the variance of the Maximum Reheat Temperature, adjusted to emphasize pattern created by the down sampling layer of the model..... 144
Figure 92 – Mean and variance (left and right respectively) of the Jacobians with respect to the Melt Count, scaled by the variance of the Melt Count..... 145
Figure 93 – Mean and variance (left and right respectively) of the Jacobians with respect to the Cooling Rate, scaled by the variance of the Cooling Rate..... 146
Figure 94 – Mean and variance (left and right respectively) of the Jacobians with respect to the Temperature Gradient in the x-direction, scaled by the variance of the Temperature Gradient in the x-direction..... 147
Figure 95 – Mean and variance (left and right respectively) of the Jacobians with respect to the Temperature Gradient in the y-direction, scaled by the variance of the Temperature Gradient in the y-direction..... 147
Figure 96 – Mean and variance (left and right respectively) of the Jacobians with respect to the Temperature Gradient in the z-direction, scaled by the variance of the Temperature Gradient in the z-direction..... 148
Figure 97 – Stacked histograms showing the distributions of the thermal characteristics across all voxels for several LPBF process parameter cases. 151
Figure 98 – Stacked histograms showing the distributions of the thermal characteristics, across a much larger subset of the LPBF process parameter cases, which have been normalized before and after stacking. 152
Figure 99 – Plots showing the relationships between the distances of the cooling rate and each of the other 8 thermal characteristics for a sample of datapoints from a subset of the LPBF cases. 154
Figure 100 – Plots showing the relationships between each of the thermal characteristics and the combined total of the remaining 8 characteristics for 1000 datapoints from each LPBF parameter set. 156

Abstract

Laser powder bed fusion (LPBF) additive manufacturing is a mesoscale process that uses small scale processing to create large scale parts. During this process, unconventional microstructures can form, which can have both favorable and adverse effects on part performance. Due to the complexity of the LPBF process and the large number of process parameters, controlling the resulting microstructures on a part specific basis has largely been impractical. While conventional simulation capabilities have shown promise for making predictions about the resulting microstructures, avoiding the time and monetary costs of printing and experimentally characterizing LPBF microstructures, these simulations are too slow for practical process tuning. This research aimed to develop a surrogate model, capable of making predictions about the microstructures produced by LPBF, fast enough to be used for process optimization.

To achieve this, a framework for the creation of machine learning based surrogate models capable of making localized predictions about the statistical distribution of grain morphology metrics of interest was developed. This framework was then used to create models that can predict spatially defined grain size distributions for LPBF printed stainless steel, using a handful of thermal characteristics, calculated by a fast thermal model, that are representative of the LPBF process. The model is shown to not only be accurate in process regimes that are well represented in the training data, but also to improve on the accuracy of the training data by removing the stochastic error present in the data. The speed of the model, combined with the improved accuracy, allows for the prediction of the outcome of ensembles several thousand conventional microstructure simulations in a small fraction of the time.

To ensure the framework and model are behaving as desired, a variety of methodologies for the analysis of the model were developed. These analyses provide insight into the behavior of the model with respect to the training data, the influence of the thermal characteristics chosen to represent the LPBF process, and the underlying assumptions used to create the model framework. The findings of these analyses support the validity of the framework and the usefulness of surrogate models created using it, showing promise for their future application to LPBF process optimization.

The code and trained surrogate model are available from:
https://github.com/sandialabs/lpbf-microstructure-surrogate

Acknowledgements

This work was partially funded by Sandia National Laboratories.

Thank you, Dan, for letting me work on such an interesting project and providing guidance throughout. Thank you, Theron, for answering so many questions about SPPARKS.

Thanks to Brian Weston for giving me someone to discuss machine learning with as I learned.

Thank you, Mrs. Brunn and Mrs. Torok, for letting a high schooler perform their own experiments. Thank you, Mr. Hunter, for reinforcing my curiosity and teaching me that sometimes a long walk is the answer. Thank you Mr. Williams for your patience and for reminding me that math can be fun even when it's hard. Thank you, Mrs. Parker, for having such enjoyable classes for computer nerds.

Thank you, Professor Handa, and Professor McGown for encouraging me to explore math further. Thank you, Doc (Professor O'Connor), for encouraging and guiding my first forays into computational research. Thank you, Professor Alexander, and Professor Watkins, for being great teachers.

Thank you, mom and dad, for always encouraging my curiosity and giving me the opportunities to do so. Thanks to my dad and brother, without whom I would not have the love of computers which got me here.

Thanks to my high school friends who have kept me going for the past several years, and thanks to Clair for all the support.

Thanks to my committee members Professor Mike Hill and Professor Jeremy Mason.

And of course, thanks to my advisor, JP Delplanque, for his support and patience, and all the many fascinating conversations we've had over the years.

In loving memory of Panda

Chapter 1 Introduction

This work is motivated by the need for fast models of the laser powder bed fusion additive manufacturing process to better guide process parameter selection and engineering design. Specifically, this work focuses on the creation of a surrogate model for the fast prediction of the statistics of microstructures produced by the laser powder bed fusion (LPBF) additive manufacturing (AM) process. Such models will ultimately enable engineers to better target specific performance metrics when designing additively manufactured parts, and help manufacturers to avoid problematic LPBF process parameter regimes. This work occurs at the intersection of the fields of additive manufacturing, materials science, modeling and simulation, and machine learning. In addition to introducing and providing background for the problem addressed by this work, this chapter acts as a brief primer on these four subjects.

1.1 Background and Motivation

In the field of manufacturing there are several classes of manufacturing processes, such as subtractive, casting, forming or deformation, joining, and additive manufacturing [1]. Subtractive processes include conventional techniques such as cutting and milling, as well as more modern techniques like electrical discharge machining, where material is removed from a larger piece to create a part. Casting processes are processes by which molten material is formed into the desired geometry using molds or dies. Forming processes, such as forging, extruding, rolling, and bending, work by physically deforming solid material to achieve a desired shape. Joining processes include processes in which two or more pieces of material are connected to create a larger part or assembly, such as fastening with bolts or screws, adhering with an adhesive, or fusing via welding, brazing, or soldering. Additive manufacturing (AM), often called 3D printing, refers to a class of processes in which material is selectively deposited, solidified, or otherwise consolidated, to form a part, without the need for a mold or die. This allows for both much more rapid iteration of the design and testing of parts that would otherwise be cast, and for the creation of much more complex parts than many of the other techniques could produce.

Additive manufacturing is among the newest fields of manufacturing, with the first patent on the topic awarded in 1956 for a photo-polymer based object replication device, though the first patent

which describes the process in full was awarded later, in 1977 [2–4]. Additionally, the first readily available academic literature on the subject of additive manufacturing did not appear until the 1980s[1] [7,8], and the first commercialized version of the process was only introduced in 1987. As such, it is one of the least understood but most rapidly advancing classes of manufacturing.

Like most classes of manufacturing, there are a variety of AM techniques which utilize a diverse set of physical processes, allowing for a variety of materials to be additively manufactured; though many of the individual AM processes can only process specific types of materials. While there are many disparate AM processes, most of them operate in a layer-by-layer fashion, depositing one layer or "slice" of the part before moving on to the next; however, this is not true for all additive manufacturing processes. For example, recent advances in photopolymerization based stereolithography processes[2] allow for volumetric printing – printing the whole volume at once – through holographic patterning and tomography [9–11].

Many joining processes could be considered a form of additive manufacturing in the literal sense, since they add material to produce the final part; however, additive manufacturing generally refers to processes in which all of the material used to create a part is selectively fused together. For example, welding two pieces together is joining, whereas creating a part entirely out of weld material is a form of additive manufacturing [12]. *Likewise, laminating many layers of wood with glue to form plywood is joining, but bonding many thin, pre-shaped, layers to form a more complex geometry could be considered additive manufacturing*.

Additive manufacturing has recently gained consumer awareness – under the name 3D printing – with the advent of mass-market fused deposition modeling (FDM) printers[3] which allow for the creation of plastic items using small tabletop machines. While AM techniques for plastics have gained a foothold in industrial settings, particularly for rapid prototyping, much of the industrial focus is on metal AM techniques. AM processes are growing in popularity in industry application because they allow for the simplification of complex assembly designs such as rocket engines and rocket bodies [14–16], for

[1] The first known description in fiction of what we would consider an additive manufacturing process appears in the 1945 short story "Things Pass By" by Murray Leinster [5], though the first know description of the general concept appeared 6 years earlier in the 1939 short story "The 4-sided Triangle" by William F. Temple [6].
[2] Stereolithography processes use UV light to selectively solidify material in a vat of polymer resin, this is traditionally done in a layer-by-layer fashion using a laser or projector.
[3] FDM printers are the most common consumer 3D printer[13], which operate by extruding molten thermoplastic polymer filament through a heated nozzle, whereupon it bonds to any previously deposited material and solidifies.

instance the aerospike rocket engine in Figure 1, significantly reducing the number of parts and manufacturing steps [17]. Additionally, additive manufacturing processes allow for part designs that would be difficult or impossible to produce using conventional methods, such as porous hip implants that allow for better biocompatibility [18–20]. This has proven to be particularly useful with the recent advances in automated design optimization, which often yield part designs that would be hard or impossible to manufacture without additive manufacturing [15,20,21].

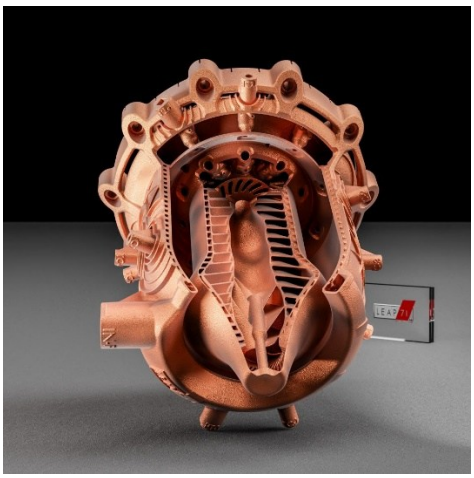

Figure 1 – Example applications of metal AM that would not be practical with conventional manufacturing techniques. Cross section of an additively manufactured aerospike engine designed by Leap 71 [15,22], reprinted with permission from Leap 71.

Because it is still a nascent field, the strategies and techniques needed to streamline additive manufacturing processes are still being developed. This is particularly true with metal additive manufacturing techniques where much of the existing knowledge about controlling the process outcomes of traditional manufacturing techniques is not directly transferrable due to the substantial differences in the physical processes. Not only are the relationships between metal AM process inputs and the properties of the manufactured part not fully understood, but these processes provide an unprecedented level of flexibility, with the large number of process parameters to control. This flexibility presents challenges to developing our understanding experimentally, and to developing models which can predict these relationships. However, this flexibility also offers the prospect of extremely fine-tuned, outcome-based process control once these relationships are better understood, and models fast enough to optimize the process have been developed. This work is motivated by this goal and explores the creation of a fast model for predicting process-structure relationships for the laser powder bed fusion process.

1.2 Laser Powder Bed Fusion

Laser powder bed fusion (LPBF) is one of many additive manufacturing processes employed for producing metallic parts. LPBF, illustrated in Figure 2, works by repeatedly laying down thin layers of metallic powder and selectively consolidating regions via melting with a laser, repeating this process until a part has been built up, at which point, the remaining powder is removed and the part is separated from the supporting build plate. These layers are only tens of micrometers thick, and the melt pool formed by the laser is only tens to hundreds of micrometers across. As a result, printing a cube just a centimeter across may require the laser to follow a path more than a kilometer long, at a pace on the order of one meter per second, and the deposition of more than 100 layers of powder. Because of these disparities in scale, the process is quite slow, particularly in practical applications, where parts are often much larger and more complicated. This limitation can be somewhat reduced through the use of multiple lasers or in the form of large-area pulsed laser powder bed fusion, though large-area pulsed LPBF is still a developing technology [23].

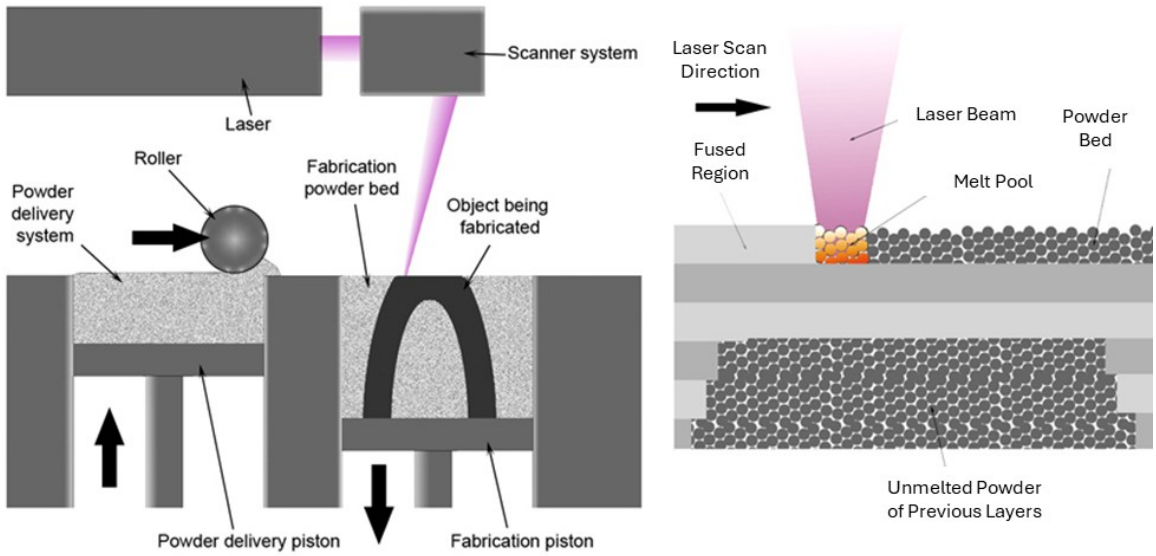

Figure 2 – Schematic of the LPBF process adapted from [24] under the Creative Commons Attribution-Share Alike 3.0 Unported license [25]. Left) Cross-sectional diagram depicting the major components of the LPBF process. Right) Detail view depicting a cross-section of the powder bed as it is processed into a part.

The LPBF process has around 100 parameters [26] which can be controlled, allowing for fine tuning of the process. These parameters include the operating conditions inside the machine, such as the ambient temperature and atmospheric makeup, as well as process specific parameters such as the layer thickness, powder feedstock, and the many parameters that control the laser, such as the beam

width, shape, scanning speed, and power. While all manufacturing techniques have the potential to influence the properties of parts to some extent, a significant number of LPBF process parameters have been demonstrated to have significant influence on performance impacting properties of the finished part [27]. By contrast, outside of forming or casting techniques, most manufacturing techniques have only surface level or locally constrained influences on the part. For example, the process parameters selected for milling operations primarily affect the process speed and the surface roughness, though they can have some influence on the surface microstructure [28].

In the case of LPBF, the selection of these process parameters is considered quite important, and poorly selected process parameters can have a myriad of detrimental impacts on the characteristics of the finished part. These detrimental characteristics include high porosity or incomplete fusion, adverse residual stresses, and adverse microstructures. Porosity refers to the capture of gas bubbles within the material as it solidifies and can reduce the yield stress of the material, though controlled porosity and incomplete fusion can allow for the creation of intentionally porous parts [19,29]. Adverse residual stresses, caused by the uneven heating, cooling, and solidification which occurs during the process, can result in deformation of parts from the desired geometry when removed from the build plate [30,31], deformations during the printing process leading to print failure [31], poorer than desired part performance [32–34], or the delamination of parts, as shown in Figure 3, where the residual stresses are sufficient to overcome the bonding between layers and cause catastrophic failure [30,35]. However, residual stresses can be intentionally used to improve part performance, for example the conventional process of peening parts to improve their fatigue performance [33,36]. It is widely hypothesized that with the proper tools to control the residual stresses in LPBF and other AM processes, it would be possible to design parts and processes which have advantageous residual stresses which improve part performance. The final example, microstructures – the crystalline atomic structures which form during solidification processes in metals – can have large impacts on the behavior of the bulk material [32,37–43]. These are of particular interest in the field of additive manufacturing, as the small melt regions and high rates of solidification produced by these processes can result microstructures with distinct morphologies compared to those produced by conventional processes. This is discussed in more detail in Section 1.3. The potential for these microstructures to have adverse influence on part performance, or to provide desired characteristics if well controlled, is the primary motivator of this research.

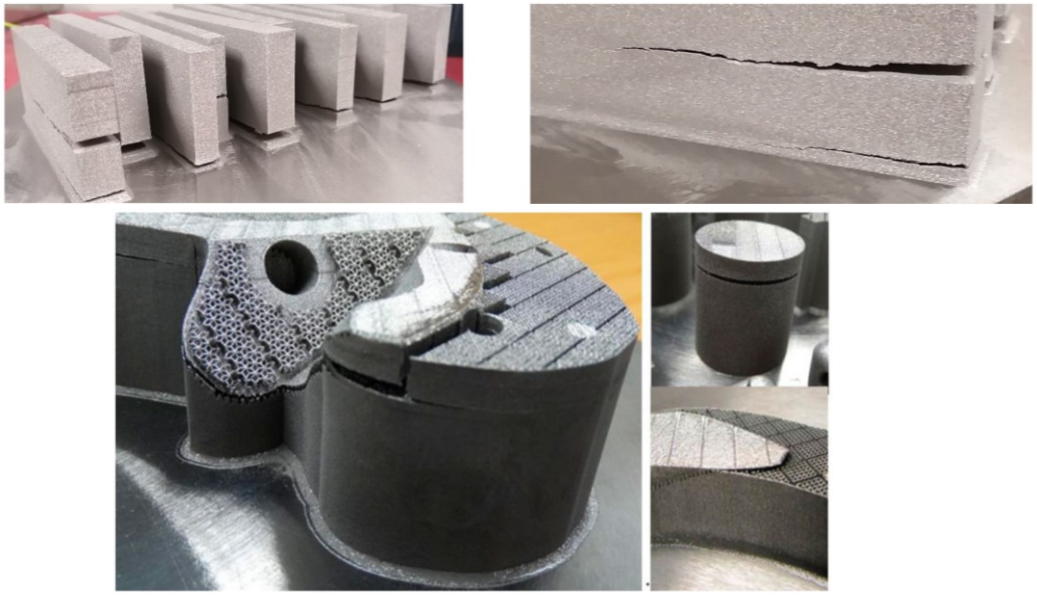

Figure 3 – Top) Delamination caused by residual stresses, from [35], reprinted under Creative Commons. Bottom) Delamination of Ti6Al4V parts from their support structure during manufacturing, from [30], reprinted with permission of Informa UK Limited, trading as Taylor & Francis Group.

Like with casting processes, in LPBF the geometry of the part can influence properties of the finished part due to the influence of cooling rates and temperature gradients on microstructures, residual stresses, and crack formation [44–46]. However, unlike casting, the LPBF process requires the creation of a geometry dependent laser scan path, similar to the toolpaths used in milling and turning operations. One of the most basic scan path types, a raster scan, can be seen in Figure 4, along with an illustration of the common practice of rotating the scanning direction on subsequent layers. These scan paths can influence the properties and performance of the part, in the same way that the geometry can, because the scan path has a significant influence on the thermal cycles every point in the part undergoes. It is for this reason that laser scan strategies are a common area of research in LPBF, and similar processes like electron beam melting[4] and laser cladding[5], which has resulted in an assortment of scanning strategies in literature and commercial AM software so that users can better control the process outcome [44,47–49]. Similarly to other processes like milling and turning, many parameters can

[4] Electron beam melting is a very similar process to LPBF, with the primary difference being the use of an electron beam, instead of a laser, to melt the material.

[5] Laser cladding is a method for adding material to existing parts by spraying metallic powder which is melted by a laser.

be made to vary along the scan path, such as the scan velocity or the laser power, making the process extremely configurable in theory.

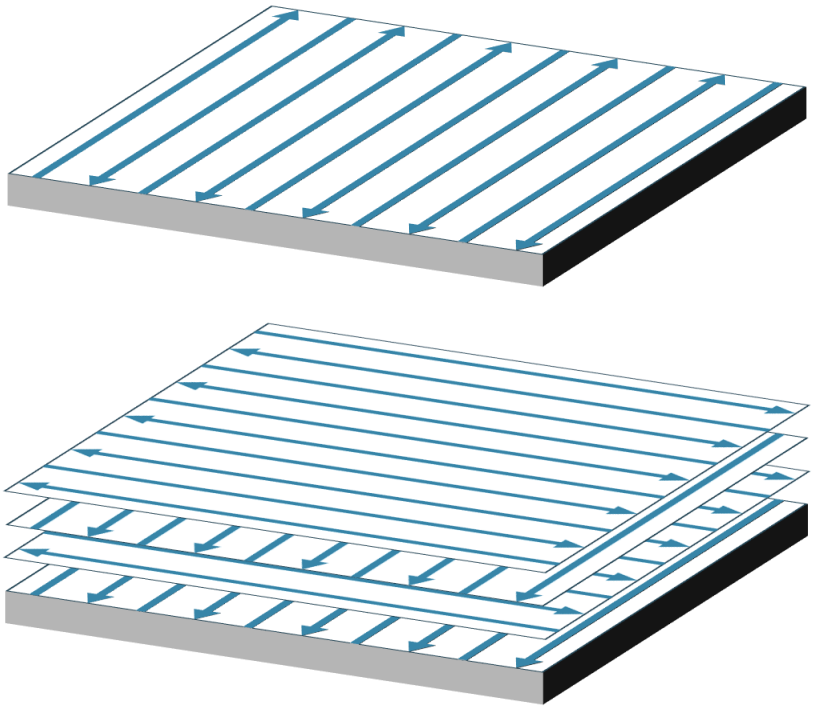

Figure 4 – Illustration of a raster scanning path for a square, with the laser switching off at turns. Top) The first layer. Bottom) One of the scan strategies which can be employed on the layers of a print: the direction is rotated by 90 degrees with each layer.

This combination of a highly configurable process, along with the extremely small size of the melt pool found in LPBF, allows for precise control of the material processing at every point in a part. In theory, this allows for the individual tuning of the properties of every point in a part, given sufficiently advanced understanding of the process-structure-property relationships and a sufficiently advanced control system [50–52]. This could mean specifying that different regions have different properties or ensuring that the variations in part properties caused by the part geometry and scan path are negated. Accounting for the influences of process uncertainty, part geometry, and scan path could prove to be particularly useful, as poorly selected process parameters are shown to have significantly detrimental impacts on part performance. Alternatively, well selected process parameters might be used to provide properties advantageous to part performance which might be hard or impossible to achieve via conventional means. This precise level of spatial control has been demonstrated to varying extents, most prominently by modifying the crystallographic texture to make specific patterns, including text, appear in electron backscatter diffractions (EBSD) scans [51–53].

Because the process is so slow, LPBF is not usually used for mass manufacturing[6] [57], but instead for creating complex parts, which are difficult or impossible to produce with traditional manufacturing techniques, such as specialized parts for the aerospace industry or customized implants for the medical industry [19]. As a result, producing test parts to dial in the process parameters for maximum part performance could require significantly more parts than the actual production run. Additionally, because of the newness of the process, along with the number of parameters which may have an influence on the part, it is not yet well understood how small variations in those parameters, may influence the performance of parts [58–60]. This uncertainty can be accounted for to some degree through the use of quality control testing and in-situ monitoring, though these come with their own challenges.

While testing and in-situ monitoring can help provide confidence in printed parts, these do not directly help with selection of the baseline LPBF process parameters for a given part. To better facilitate this portion of the process design, without the need for extensive experimental iterations, we can turn to computational modeling and simulation. Though modeling and simulation of the LPBF process are active areas of research, they have been shown to provide valuable insight into the physical processes, and show promise for predicting outcomes prior to printing. These models range from expensive high-fidelity multi-physics simulations, which allow us to study the process in detail, to data-based process maps, which provide fast heuristics-based estimates of the process outcomes. In between these are reduced-order models, which make approximations for at least some of the physics of the process, as well as a growing number of machine-learning based models, which rely on large amounts of data to make fast predictions about specific process outcomes. Due to the computational expense of high-fidelity simulations, and the non-specificity of process maps, these two types of models are not generally practical for use in LPBF process optimization.

While there have been significant advances in modeling and simulation of LPBF process outcomes in the past several years, much work is still needed to develop these models into tools which

[6] Apple announced in late 2025 that they are using LPBF to mass manufacture watch cases out of titanium to cut back on the volume of scrap material produced by conventional subtractive manufacturing, significantly reducing their raw material usage and helping them reach their sustainability goals [54–56]. This is the first known use of LPBF for the mass manufacture of a widespread consumer product that we were able to find.

are practical as part of the engineering design process. In this research we explore the creation of a surrogate model for predicting statistics of microstructures produced by the LPBF process which is fast and accurate enough to use for process optimization as part of the engineering design process. A diagram depicting what this process might look like can be seen in Figure 5, where an initial set of process parameters is provided and fed into a thermal model, the results of which are fed into our fast microstructure model, which is then used to inform an optimization algorithm. As part of the creation of our microstructure surrogate model, we will use high-fidelity microstructure simulations to inform and validate our surrogate model. Because the microstructure simulations which we wish to replace do not have any obvious paths to the development of a reduced order model, it was known that the most likely path to the creation of our surrogate model would be through the use of machine learning algorithms. Before introducing the proposed framework for the creation of such a model in Chapter 2, the rest of this chapter is devoted to providing relevant background information.

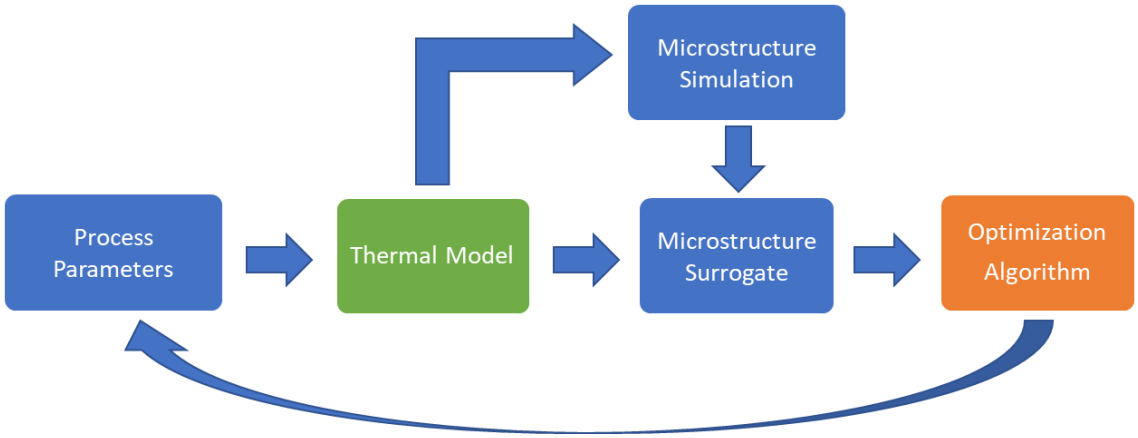

Figure 5 – Flowchart illustrating the conceptual process optimization loop. Microstructure simulations are used to inform the microstructure surrogate model, without the main optimization loop relying on them.

1.3 LPBF Microstructures

Microstructures are the mesoscopic scale structures that make up materials. In metals, these structures generally consist of crystalline atomic lattices composed of repeating patterns. While it is possible for an object to consist of a single crystal structure, the processes which form these structures often result in misalignments between individual sections of atomic lattice. As these crystals grow, these misalignments result in crystal structures which do not link together to form one crystal, but instead remain as separated, but bonded, structures due to the misalignments causing disruptions in the lattice.

These individual crystals are called grains and can range from the micrometer scale all the way up to the centimeter scale[7]. The size of these grains, frequency of these grain boundaries, alignment of the crystal structures, and existence of other sub-structures caused by the proportions of the constitutive elements, can all have impacts on the bulk properties of the material, such as the yield strength and fatigue performance [40,61–66]. This is a particularly interesting area of research in LPBF due to both the relatively unusual microstructures which can form during the process, as well as the potential to control these in a precise manner [26,39,51,52,67–70].

In LPBF, the small melt pool combined with the comparatively high speeds at which the laser heat source moves, results in this process having extremely high cooling rates and thermal gradients. These cooling rates are often in excess of 10^4 K/s, making it a rapid solidification process, which can result in the formation of non-equilibrium microstructures [71,72]. Additionally, due to the high thermal gradients and repeated layering, the resulting microstructures often have high proportions of long columnar grains [73,74]. These microstructures are uncommon in processes like casting, where the comparatively large amount of molten material takes longer to cool down[41,75–77].

The microstructures which result from the LPBF process are unusual in the world of conventional metallurgy due to the high cooling rates, large thermal gradients, small melt pools, and many other factors including interactions with the laser [29]. Most frequently discussed in literature is the high propensity for the formation of long columnar grains, which can have negative impact on part performance [32,78]. Also of interest is the tendency of the very high solidification rates involved in the process to result in very fine, equiaxed[8] grain structures, which in certain use cases can be advantageous for part performance. Being able to control when and where these microstructure regimes occur when printing parts would therefore be advantageous for modifying part characteristics during the design process. However, the mechanisms of formation of these grain structures are complex, and the process parameters interact in ways that make it hard to define simple rules for precise control of the microstructure formation.

[7] Zinc crystals which form the surface of galvanized steel are often visible to the naked eye.
[8] Equiaxed grains are grains which have dimensions which are nearly equal in all directions.

1.4 Modeling And Simulation of LPBF

There are a wide variety of models with which the LPBF process can be simulated, with varying levels of modeling fidelity and tradeoffs. The simplest models available are regression based models or process maps, which do not actually simulate the process, but instead rely on experimental and simulation based data to predict characteristics like how deep the melt pool will be based on the process parameters [69]. Jumping up to high-fidelity models, there are multi-physics codes which simulate the heat transfer process, the fluid dynamics of the melt pool, and the interaction of the laser light with the melt pool and the powder [79,80]. Such models have the added advantage of allowing us to explore the impact of each individual physical phenomenon by turning on or off specific physics [81], which can provide insights on how best to modify the process to achieve a desired outcome. In between, there are reduced order simulations which approximate or remove some aspects of the physics involved. Additionally, there is a growing interest in machine learning based surrogate models, which use large amounts of data to approximate the output of physics-based simulations. While machine learning based models need large amounts of data on which to learn, the resulting models are much faster than traditional simulations and much more precise than traditional data-based methods.

The foundational model required for simulating LPBF is a thermal process model because the LPBF process is fundamentally the melting and subsequent solidification caused by a moving heat source. These models allow for the direct prediction of the melt pool geometry, and resulting characteristics such as lack of fusion and surface roughness, and can be coupled with other types of simulations for the prediction of other part properties. For instance, combining a thermal model with a mechanical model allows for the prediction of residual stresses in printed parts by modeling the influence of different regions heating, cooling, and solidifying at different rates and times [82]. Alternatively, combining the thermal model with certain computational materials science models allows for the prediction of material phase and probability of solidification cracking [46], or the prediction of complete microstructures as printed by the LPBF process.

1.4.1 Thermal Modeling

Thermal process modeling of LPBF is a continually expanding field, with a significant amount of interest in surrogate models. There is so much interest because thermal modeling can be used in

conjunction with structural models to predict residual stresses, and in conjunction with materials science models to make predictions about microstructures. While there is continued interest in high fidelity thermal models, which allow us to study the complex phenomena which occur during the LPBF process, the extremely large disparity in time and length scales between the melt pool and a full printed part make surrogate models an increasingly popular topic.

High fidelity models such as those by Heo et al [26,79,83] can resolve individual powder particles, the reflections of the laser light in the denudation zone of the melt pool, as illustrated in Figure 6, and the convective currents in the melt pool. These models allow us to observe phenomena which would otherwise be hard or impossible to observe directly, such as the influence of Marangoni convection [84], and the influence of keyholing on pore formation [85].

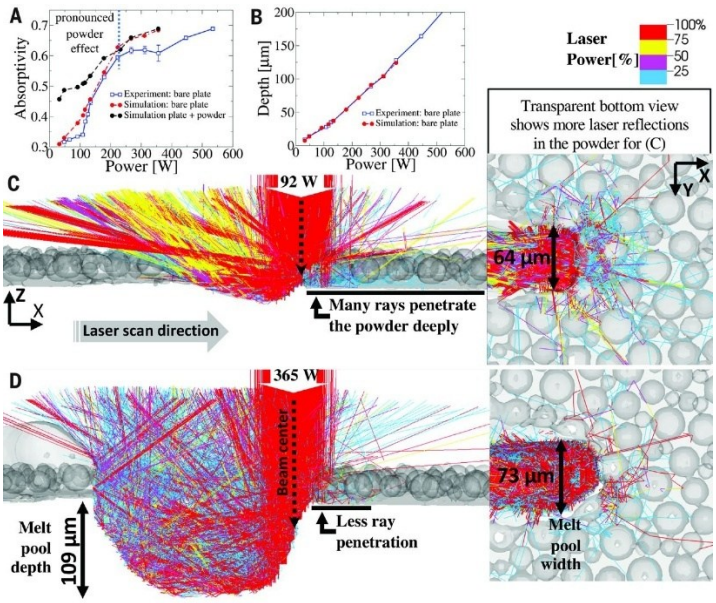

Figure 6 – Illustration of the behavior of the laser light source for the LPBF process, as simulated by a high-fidelity multi-physics simulation with ray tracing. From Khairallah et al [83], reprinted with permission from AAAS.

Because of the complexity and computational cost of such high-fidelity simulations, there are few such models in existence, and they are predominantly used for studying the process. Instead, most LPBF thermal models found in literature attempt to simulate the minimum amount of physics necessary to achieve accurate predictions. One such method for the simulation of the LPBF process is to model only conduction – excluding explicit convective and radiative heat transfer – and treat both the loose powder and the consolidated part as a homogeneous material with constant effective material properties which have been calibrated such that the model can accurately predict the behavior of the

melt pool [86]. While such models do not accurately describe the physics of the process, the overall temperature predictions of the regions around the melt pool have been found to have a reasonable degree of accuracy. Additionally, due to the nature of the equations which describe conduction mode heat transfer, it is possible to find analytical solutions, allowing for the creation of semi-analytical models that can simulate the LPBF process much faster than conventional finite difference or finite element methods at the expense of temporal resolution [86].

While these reduced-order models allow for the simulation of the LPBF process much faster than high-fidelity simulations, they can still be too computationally expensive for some applications. One option for making predictions without the need for simulations is the use of data driven models or process maps, which provide a heuristic approach to quickly making predictions about the LPBF process such as the melt pool depth and width, both of which are common metrics analyzed using experimentation and simulation[87]. Another approach that has been gaining popularity is the use of machine learning to either replace or augment thermal simulations. These models are trained on large datasets of thermal simulation results, and generally allow for the prediction of future states given a set of inputs [88].

1.4.2 Microstructure Modeling

As computing power advances and materials science algorithms get better, there is growing interest in using computational modeling to model the influence of LPBF process parameters on microstructures. This includes such things as targeted models of phase segregation during the solidification process, to help us understand some of the finer details of the process [46], as well as much larger and more complex models and simulations of the complete microstructures which are formed during the printing process. Such models can be combined with other models for the prediction of physical characteristics such as fatigue performance and stress-strain responses [78,89], which allows for the prediction of part performance; however, the focus of this research is solely on the creation of a surrogate microstructure models, so we will not explore these more.

There are three primary methods for direct simulation of the solidification and grain growth process: phase fields, cellular automata, and kinetic Monte Carlo methods. Each of these methods has benefits and tradeoffs, as well as slightly different use cases. Phase fields, which numerically solve

differential equations that have been configured to behave similarly to physical processes, are generally the highest fidelity form of microstructure simulation. While this provides more detail about the solidification process than the other methods, this comes with increased computational cost, and phase fields are typically used for simulating smaller domains than the other methods. In contrast, cellular automata and kinetic Monte Carlo methods both rely on voxel-based representation of the domain, and use simpler relationships between neighboring voxels, instead of attempting to solve continuous differential equations. This makes these methods much faster, and better suited for the simulation of larger domains, at the expense of the accuracy of the physical process being modeled. Generally speaking, cellular automata methods update the state of voxels based on a set of rules in relation to the neighboring voxels. Kinetic Monte Carlo methods are similar in that voxels are updated based on the state of their neighboring voxels, with the distinction that such updates are probabilistic.

While all three approaches to the simulation of solidification dynamics have been used successfully in the modeling of the LPBF process and other similar processes [51,90–95], we were able to quickly rule out the use of phase field approaches for our purposes, as they are not conducive to the goal of this research. Because their high computational cost makes them slower than other methods, and less suitable for simulation of large domains on the same scale as LPBF printed parts, phase fields would not be a practical method for generating the large amounts of data needed to train a surrogate model [96]. While cellular automata have some advantages over Monte Carlo methods for the purposes of predicting solidification dynamics [97], Monte Carlo methods are generally less computationally expensive and better suited for the modeling of large domains [66].

The avenues for the speedup of these conventional microstructure models are limited due to the reliance of these methods on small time steps and high resolutions to capture the multiple time and length scales of nucleation and grain growth. Models for some physical processes such as heat transfer, can be sped up by reformulating them to allow for larger time steps or coarser resolutions via simplified physics and semi-analytical solutions to the underlying differential equations [86]. However, in the case of microstructure simulations, particularly cellular automata and Monte Carlo methods, the stochasticity and cascading effects created by consecutive updates of voxel states likely make it impossible to develop a closed form solution.

However, while these models do not lend themselves to mathematical acceleration strategies, a variety of known relationships between the solidification process and the resulting microstructures have made it possible to create fast surrogate models which approximate the general behavior of the microstructure formation process. The most prominent of these relationships, Hunt's criteria for the columnar to equiaxed transition of grain growth, and subsequent variations, have been adapted to approximate which regions of a weld will contain columnar grains [98,99]. Additionally, advances in machine learning techniques have made data-driven surrogate models and mixed simulation-machine learning models [100] viable options for the acceleration of microstructure predictions.

1.5 Machine Learning

Due to the complexity of the models used for predicting microstructures in additive manufacturing, it was anticipated that mathematical simplifications of the underlying physics alone would not be sufficient for producing a fast surrogate model. The alternative methods for producing fast surrogate models can all broadly be classed as machine learning. Here we will provide a background on machine learning, including some of the applications and underlying algorithms.

Generally speaking, machine learning refers to methods and algorithms "which can automatically detect patterns in data,"[101] and use those patterns to make predictions related to the data. There are three primary types of problems to which machine learning is applied: regression problems, where the goal is the prediction of numerical values, clustering problems where sets of data are automatically assigned to groups, and classification problems, where the goal is to predict the most likely element of a set.

Though it is not what first comes to mind for many people when discussing machine learning, linear regression is the foundation on which many machine learning methods are built. One of the simplest types of regression problems is simple linear regression: drawing a best fit line through a set of 2D data points. While simple linear regression refers to fitting straight lines, broadly speaking, linear regression is used to fit functions, such as a polynomials, to data of any dimensionality. This contrasts with non-linear regression, which uses non-linear equations to predict more complex relationships.

A similarly simple example of a clustering problem is drawing a line between two groups of data points on a scatter plot, trying to find the line that best divides them. Examples of classification problems

are identifying what species of animal is in an image, or using traits of an animal to determine its species. Classification problems dealing with image data are often treated as a subset of classification called image classification, and often use different methods than non-image classification. While they do not intuitively appear as classification problems, the most prominent example of the application of classification algorithms in the public eye at the time of writing is the use of Large Language Models (LLMs) for text generation in the form of chatbots or autocompletion, which predict the most likely word based on contextual data [102,103].

Conceptually, classification problems are quite similar to clustering problems, in that the data is being sorted by which group it belongs to; however, the key distinction is that clustering is used to identify the groups which exist in data, whereas classification uses pre-defined groups to learn how to sort data. To put this in the terminology of machine learning: classification is supervised learning, meaning it uses pre-labeled data to learn specific categories, and clustering is unsupervised learning, meaning it is only given the raw data without any concept of what the categories are.

The methods and algorithms required to solve all of these problem types can be quite complex for all but the simplest problems. While many machine learning methods are best suited to specific problem types, there are a few that stand out for their broad applicability. Two of the most broadly applied classes of machine learning methods, are Gaussian processes and neural networks, which have gained popularity in the field of surrogate modeling for a variety of scientific and engineering problems. These non-linear methods can generally be considered universal function approximators, regardless of if the functions to be approximated are known [104–108].

Gaussian processes are a method for applying the principles of Bayesian statistics to regression problems, which allows for predictions of both values and uncertainty of the predictions. While Gaussian processes are very powerful tools for modeling, the algorithms involved have practical limitations when applying them to very large problems, in terms of both the size of the data set and the number of independent variables used as inputs. While there have been recent improvements to the algorithms available for Gaussian processes, allowing for their application to larger problems, these methods are new enough that the choice was made not to pursue these methods. This is because it was not well understood going into this research how much data would be needed to create a model that covers the full process space or how much information would be needed as inputs to the model, or what form the

model would need to take. By contrast, neural networks are particularly well suited to handling large amounts of data, including large model inputs, and offer much more flexibility in implementation than Gaussian processes.

1.5.1 Neural Networks

Neural networks (NN) are a class of machine learning algorithm which broadly mimic the signal-activation methods of neurons in a brain [109]. The most basic building block of a NN is the perceptron, often referred to as a neuron or node, which multiplies its inputs by learned weights, sums them along with a learned bias term, then applies a non-linear activation function to the resulting value. In this basic form the perceptron has limited functionality, but combining many of them, each with their own weights and biases allows for the NN to learn complex relationships in data. Combining multiple of these in parallel, each with the same inputs, creates what is referred to as a "layer" of a neural network. Several of these layers can then be combined in series, with the outputs of one layer serving as the inputs to the next, to create a full neural network; this basic form of neural network is called a multi-layer perceptron (MLP). Neural networks with multiple layers between the input and output, called hidden layers, are more generally referred to as deep neural networks, which gives rise to the name "deep learning."

The learned parameters – the weights and biases – of neural networks are set using a process called back-propagation – commonly referred to as training – in which an optimization algorithm iteratively tunes the parameters. This optimization process attempts to fit the overall behavior of the neural network to a set of training data that has been selected to represent whatever problem is being solved, in much the same way that linear regression fits an equation to a set of data points. Unlike with linear regression, however, the complexity of neural networks makes it generally impossible to find the global optimal solution, and practitioners instead rely on advances in optimization and training techniques to ensure their NN models are accurate. There are a variety of optimization algorithms which have shown promise for training neural networks; however, they all rely on the same basic principles of adjusting the model parameters proportionally to the gradient needed to minimize the error – called the loss –of the model's predictions for a given set of inputs, against the target values given by the training data. This process often iterates over subsets of the training data – called batches – allowing the model to train on much larger datasets than would otherwise be possible. This batch training has also been

shown to have benefits for the overall accuracy of neural networks, particularly with respect to data not included in the training data.

In addition to tuning the weights and biases of neural networks, the parameters which describe the overall neural network – known as the hyperparameters – can be tuned to make NNs better suit the problem being addressed. These hyperparameters include the width – the number of perceptrons – of each layer, the number of layers, and sometimes other NN features such as the activation function. While MLPs are the prototypical type of neural network, other forms of neural networks can be created by adding other mathematical operations alongside the basic perceptron. Two of the most prominent and distinct classes of neural network are convolutional neural networks (CNNs), and transformers. We will provide a brief background on these two classes of NN in the next 2 sub-sections; however, as this research does not focus on the development of a new NN architecture, we will not get into the details of how they work. These other classes of neural networks introduce their own hyperparameters, such as the kernel width and kernel stride in the case of convolutional neural networks. Generally, larger networks – as described by the total number of learned parameters – are considered more capable of learning large amounts of information or more complex relationships, but come with tradeoffs of increased computational cost for both training and inference, and can be more challenging to train. To balance these tradeoffs, a second layer of training optimization can be carried out in the form of hyperparameter optimization, where a NN is trained multiple times on the same data, with an optimization algorithm controlling the NN hyperparameters.

In the case of both NN training and hyperparameter optimization, the goal is generally not to find the model with the highest possible accuracy with respect to the training data, but instead to find a sweet spot where the model predicts the training data with sufficient accuracy without overfitting the data. Much like with polynomial regression, where a poorly tuned polynomial can have erratic behavior both between and beyond the data points to which it is fit, a neural network which can perfectly duplicate the data it is being trained on is often indicative of the model learning to encode the training data, or overfit, rather than learning meaningful relationships. One of the primary methods for mitigating this when training neural networks is to split the data available for training into test and training sets. By letting the neural network train on a subset of the data, a user can monitor the performance of the neural network with the remaining test data to determine if the model starts to

overfit. Additionally, a variety of techniques have been developed to minimize overfitting and help NNs generalize to new data not included in the training data. These techniques largely revolve around augmenting the training data, such as adding small amounts of random noise to numerical or image data, flipping or rotating array-based data, or randomly removing portions of the data. Other techniques such as dropout – which randomly skips over neurons during training – add variation to the model itself during training process, resulting in the model effectively being an ensemble average [110].

In addition to the already mentioned convolutional neural networks and transformers, other mathematical constructs and connections between layers can be incorporated to create many types of neural networks, including Bayesian NNs, MLP mixers, recurrent neural networks, and physics informed neural networks (PINNs). Additionally, by combining multiple neural networks, NNs can be created which are particularly good at certain tasks. For example, autoencoders are good at dimensionality reduction tasks, and variational autoencoders and generative adversarial networks are useful as generative models which can do things like creating images from a text description.

1.5.1.1 Convolutional Neural Networks

Convolutional neural networks are a class of machine learning neural network most commonly associated with image classification tasks, which use convolution operations instead of linear layers as their primary building block. Convolutions are a mathematical tool which, in the functional space, operates on two functions by effectively sliding one function across another to produce a third function. In the world of discrete mathematics, where we operate on arrays of numbers rather than symbolic equations, the convolution operation instead marches an array, called a convolution kernel, across another array, creating a new array with values at every step the kernel takes. The values at these points are the dot product of the values of the kernel with the values of the section of the array that the kernel is overlapping. In convolutional neural networks these kernels are learned in the same way as the weights of perceptron-based networks, and operate on the data as it passes through the network.

First proposed in 1998 for the purposes of text recognition, ConvNets originally consisted almost entirely of convolution layers, only including linear layers at the very end [111]. These neural networks gained popularity for image classification problems in 2012 with the advent of AlexNet [112], due to its performance in the ImageNet challenge [113]. While these classes of NN gained popularity for their performance classifying images, they have since grown in popularity in many fields that work with multi-

dimensional spatial data. These include applications in modeling physical systems, such as fluid flow and additive manufacturing processes, and processing 2D and 3D medical scans.

1.5.1.2 Transformers

Transformers are a deep learning architecture which serve as the basis of many state-of-the-art large language models – such as ChatGPT, Gemini, LLaMA and Claude – and image classification networks based on the Vision Transformer (ViT) architecture. These neural networks rely on the concepts of tokenization, which converts data into vector embeddings the model can learn to interpret, and multi-head attention, which allows the model to learn the relative importance of each token in a sequence to each other token. Multi-head attention is somewhat similar in concept to the convolutions used in CNNs, allowing the model to learn how certain arrangements of inputs create meaning. However, they are significantly different in that convolutions use kernels to learn patterns of adjacent input data, whereas multi-head attention learns to predict the importance of every token in the input to every other token. This allows transformers to learn long-distance relationships directly; in contrast, convolutions inherently assume that closer data is more important – except with very large kernels – and only learn long-distance relationships via several layers of these short distance relationships. This explanation is a simplified version of how transformers work, as the architecture is quite complex in practice, but the end result of this combination of token embedding and multi-head attention is that transformers are very computationally expensive and require large amounts of data to learn well.

Because transformers have been shown to have state of the art performance in multiple applications, and due to their ability to learn non-local relationships between data without assumptions of closeness, transformers were initially considered as a promising option for the development of our surrogate microstructure model. The non-local relationships of transformers were deemed interesting because while this is a physical process where the only direct interactions are between adjacent points, it was not known if the complexity of the process might make the ability to learn long-distance relationships advantageous. However, the high computational and training costs ultimately led to the decision not to pursue a transformer as the basis of our surrogate model.

Interestingly, the success of the transformer architecture has led to the revisiting of many other architectures, including former state of the art models, to apply lessons learned from and since the advent of transformers. This has resulted in MLP and convolution-based image classification networks

which have comparable accuracy to state-of-the-art transformer-based models despite often being less computationally expensive [114,115]. It is one of these resulting architectures which was ultimately used as the basis for our surrogate model, as is discussed in Section 3.2.

1.5.2 Machine Learning Surrogate Modeling

As machine learning techniques and computational power have advanced over the past several years, there has been a growing interest in using these methods to create surrogate models which can quickly make predictions about physical phenomena without the need for full simulations. While basic regression-based curve fitting and empirically based heuristic models have been in use for much longer than machine learning has existed, machine learning models offer the prospect of much more advanced predictions, based on a larger number of parameters. Such machine learning based surrogate models have been explored in a variety of scientific and engineering domains [116], from fluid mechanics to protein folding predictions [117].

One of the most interesting machine learning methods for the purposes of surrogate modeling is the Physics Informed Neural Network (PINN) – which use mathematically defined physical relationships to constrain the predictions of the NN to ensure physical consistency. While these have gained traction in many areas of physics-based modeling, the lack of clear and consistent relationships that can be used to constrain microstructure evolution along arbitrary time scales make PINNs an impractical choice for this stage of developing surrogate microstructure models for the LPBF process.

These methods have found growing use in the field of additive manufacturing for the prediction of the geometry of melt pools and modeling the thermal process [118–120], the prediction of residual stress and stress-strain responses [50,121], and to some degree for process optimization [122–124] and prediction of microstructures [96,125]. As the focus of this work is on the prediction of microstructures using a machine learning based surrogate microstructure model, we will briefly touch on two of the most effective of such methods found in our review of the literature[9].

One of the most similar methods for the creation of a surrogate microstructure model to that presented here comes from Knapp et al at Oak Ridge National Laboratory [125]. In this paper they use an

[9] Both of these models were published after the development of the surrogate model presented in this research was complete. As these models were developed concurrently, this provides an interesting comparison of other methods for the development of such models given the same advances in machine learning and computational power.

unsupervised clustering approach to make predictions about the heterogeneity of microstructures using characteristic thermal data from groups of voxels. While we do not use a clustering approach in this research, there are many directly comparable concepts between their methods and those presented here.

A less directly comparable method was developed by Ma et al, and uses machine learning to create a surrogate model that emulates phase field simulations [96]. For this model, a generative model called a diffusion neural network was used; such NNs are used by many popular image generation models. The resulting model is capable of predicting small regions of a microstructure based directly off the LPBF process parameters. Because it does not attempt to emulate or represent the full thermal process in any way, and was trained on very small phase field simulations with specific initial conditions, the practical applications of this model are limited. However, it serves as an interesting proof of concept, and may provide an interesting way to make quick predictions about specific characteristics.

1.6 Research Goals and Dissertation Structure

The goal of this research is to develop and analyze a framework for the creation of surrogate microstructure models for the laser powder bed fusion additive manufacturing process, which are fast enough to be used for the optimization of process parameters based on the resulting microstructures. To do this, after developing the framework, it is used to create a surrogate model for the prediction of grain size distributions produced by the LPBF manufacturing of 304 stainless steel. We then explore the usefulness of this model, the implications of observed phenomena, and the underlying assumptions used when creating the model and model framework.

Chapter 2 introduces the framework and the theory used to develop it, then introduces the thermal and microstructure models that are used for the creation of the surrogate microstructure model. Additionally, the form of the data used by the surrogate model is introduced and explored.

Chapter 3 delves into the specifics of the surrogate model, including the architecture of the model, and the actual process of creating the model. Additionally, the method used to measure the model's accuracy is explored through numerical experiments, with particular emphasis on the influence and implications of the stochastic nature of microstructure formation.

Chapter 4 analyzes the accuracy of the model, both in the context of the process parameter sets used to train the surrogate model, and in the context of making predictions about new process parameters. In order to better explore the implications of some of these results, the impact of the number of simulations used to generate training data is also explored.

Chapter 5 directly analyzes the model's understanding of the thermal characteristics used to represent the LPBF process to the surrogate model, and explores how well the dataset used to train the model represents the full LPBF process space.

Chapter 6 summarizes the findings of this work and their implications, and provides some thoughts on the future paths this research could take.

Chapter 2 Surrogate Model Framework

2.1 Introduction

This chapter discusses the details of the framework and tools used to create the proposed microstructure surrogate model. In Chapter 1 we introduced the envisioned process optimization use case for the surrogate microstructure model, which is informed by microstructure simulations, with inputs from a thermal process model, as illustrated in Figure 7. Here we will explore these relationships between the surrogate model and the simulations which inform it in more detail. First, we will explore how the flow of data acts as a guiding framework for the creation of the model, including how the form of the data can guide the methods used by our model, and finally, discuss the data and simulations used to create it in detail.

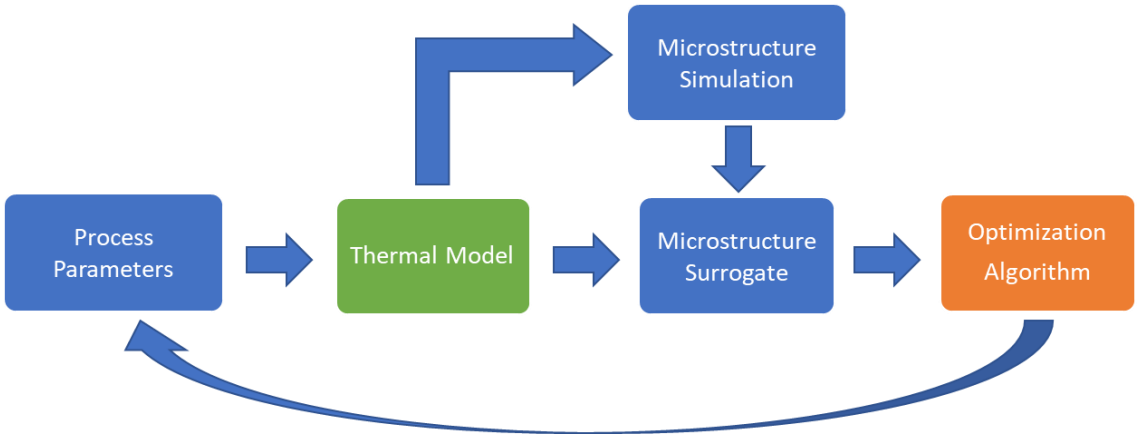

Figure 7 – Flowchart illustrating the conceptual process optimization loop. Microstructure simulations are used to inform the microstructure surrogate model, without the main optimization loop relying on them.

2.2 Framework For Model Development

As previously mentioned, the avenues for accelerating microstructure simulations via a reduced order or reduced fidelity approach are limited due to the nature of the methods used by these simulations. Instead, the framework illustrated in Figure 8 was designed for the development of a surrogate microstructure model which could statistically approximate microstructure metrics based on characteristic representations of the thermal history of the printing process. The process for the creation of the surrogate model starts with a thermal model of the LPBF process, which is used both as part of a traditional microstructure simulation and to produce thermal characteristics used as inputs to our

microstructure surrogate model. The results of the microstructure simulations are post-processed to produce statistics about grain morphology metrics. These statistics are then fed along with the thermal characteristics, into the microstructure surrogate model for the purposes of learning a mapping from the thermal data to the microstructure data. The resulting microstructure surrogate model can then be used in conjunction with the thermal model to make predictions about microstructure statistics for new combinations of LPBF process parameters and part geometries.

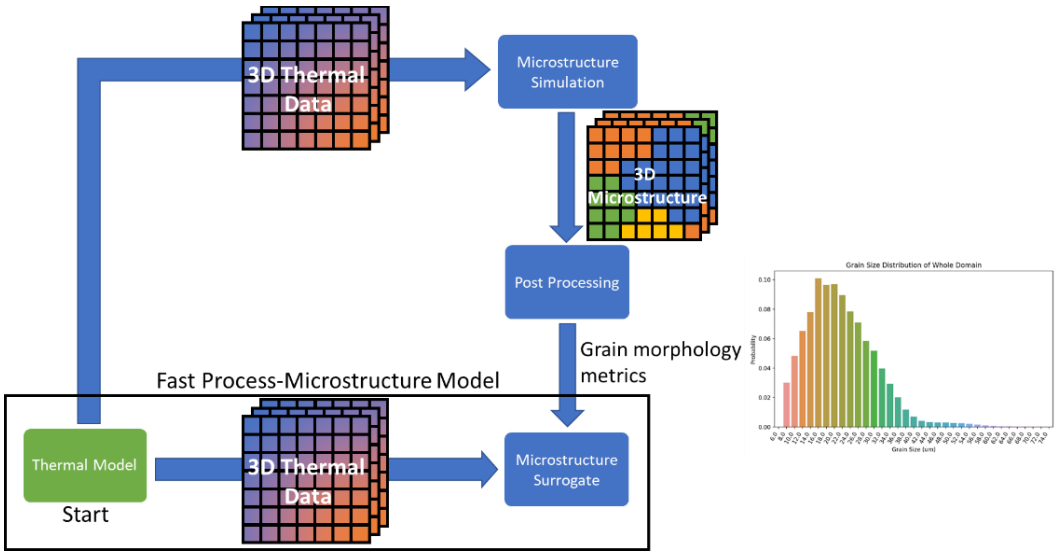

Figure 8 – Framework for the creation of the surrogate microstructure model.

For both the thermal modeling and microstructure simulation components of this framework, any of the many available LPBF models can be utilized so long as they can produce the requisite thermal and microstructure data. Interestingly, this also hypothetically allows for the substitution of the thermal model after the microstructure surrogate model has been created, as the surrogate model can in theory accept any physically valid thermal history, in the same way that microstructure simulations can. However, doing so may result in unpredictable behavior from the surrogate microstructure model if the thermal history is sufficiently different from any of the data used to create it[1].

Because the goal of this research is to create a model fast enough to optimize the LPBF process parameters, which requires running both the thermal model and the surrogate microstructure model, the best candidates for the thermal model are those which can simulate large process volumes relatively

[1] This may be possible to mitigate via fine tuning a trained model on new data, but this is beyond the scope of this work.

quickly. This also helps accelerate data generation for the purposes of creating the surrogate model, though the microstructure simulations will generally be the limiting factor for reasons which will be made clearer shortly.

While the framework depicts the thermal model separately from the microstructure simulation, with data flowing from one to the other, in practice, microstructure simulations generally require running the thermal model as part of the microstructure simulations. This is because many methods for microstructure simulations model the solidification process in full, requiring access to nearly every time step of the simulated thermal process, something which is impractical to achieve by saving and loading the simulated thermal process. However, an identical standalone thermal model is necessary for making predictions about the microstructure using our surrogate model without running a full microstructure simulation. Unlike with the microstructure simulations, we do not wish to couple this directly with our microstructure surrogate model, but will instead find a way of representing the full thermal history using a collection of summarizing thermal characteristics.

As discussed in Section 1.4.2, there are a variety of methods for simulating the formation of microstructures during the LPBF process. All such models should be compatible with the framework presented here, though slower models may not be feasible for producing the large amounts of data needed to train many machine learning methods. Additionally, not all models are capable of simulating all aspects of the microstructure which may have an impact on the performance of a printed part, so the microstructure model must be chosen with the desired grain morphology metrics in mind. For the purposes of this research, we will use the SPPARKS additive manufacturing model developed by Theron et al [126], which we will discuss more in Section 2.4. Like with the thermal model, this framework allows for the use of any microstructure model which relies on thermal simulations of the LPBF process.

The microstructures produced by the microstructure model then need to be post processed into statistical representations of the grain morphology which the surrogate microstructure model will learn from and attempt to reproduce. For the purposes of the work presented here, the microstructure statistic used is the grain size distribution. This can in theory be swapped for any other grain morphology metric such as the aspect ratio, average grain size, or even texture.

To create the microstructure surrogate model itself we started by reviewing what classes of non-linear regression algorithms exist. Gaussian processes were considered but ultimately rejected due to

the high dimensionality of the problem. Turning instead to neural network architectures we decided the methods used for image classification would be most fitting, as the formation of microstructures is highly dependent upon spatial relationships of values in an array, something that image classification networks also rely on. However, this alone is insufficient for applying these networks to these problems; the key is that image classification networks generally output a vector of probabilities – though it is normally hidden, with only the maximum value returned – which we knew could be a viable representation of our microstructures in the form of grain size distributions. Figure 9 illustrates the similarity between our expected data structures and the data structures used by an image classification network. Here our input data is an array representing 3D space with multiple channels and a probability distribution represented by a vector as an output, and an image classification model has an array representing a 2D image, with multiple color channels, as an input and a vector of probabilities as an output. While our proposed method differs from image classification networks in that image classifiers are designed to correctly predict the most likely option, whereas our model is intended to predict probability distributions, using neural networks to predict probabilities or distributions is not a novel concept [127–130].

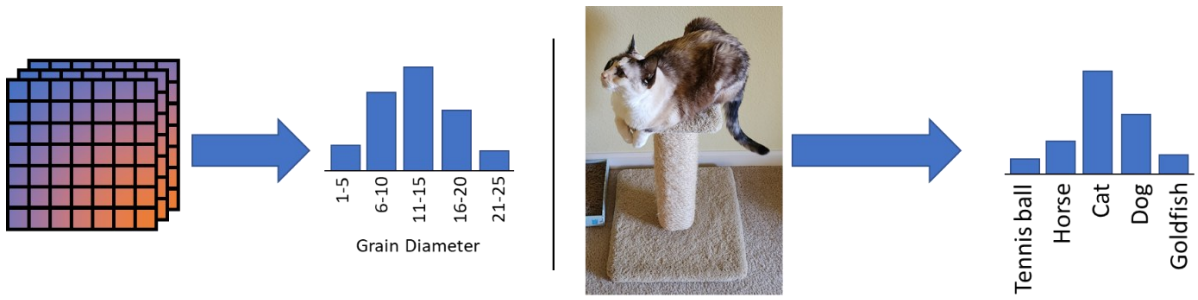

Figure 9 – Data-wise comparison between our model and an image classification network

This still left us with problems to address, namely, how to apply our model to problems with variable print or simulation domain sizes. In many neural networks used for array-based data, the size of the input is fixed due to mathematical constraints, changing the size of the input results in mismatched matrix dimensions. To address this, we use a moving input window, feeding the model only a fixed size portion of the thermal data and having it make predictions about just that portion, in this case just the centermost point or voxel of the input window. This has the added benefit of then allowing us to make

localized predictions about each voxel of the domain individually[2], providing detailed insight and theoretically allowing for very precise process optimization. However, this requires that we have statistical representations about each voxel of the print for training the model; to do this, we need to run the microstructure simulation several times using the same parameters, relying on the stochastic nature of the microstructure simulations to produce an ensemble of microstructures which can be converted into spatially defined statistics. It is for this reason that the computational costs of the microstructure simulations used to create our training data are the limiting factor for data generation.

This combination of a moving window and spatially defined statistics is illustrated in Figure 10, the top half shows the thermal data with a moving window outlined in red, we then see how this corresponds to a single voxel in two microstructures from an ensemble of simulations, and finally the corresponding grain size distribution for that voxel. In the bottom half of Figure 10 we see how shifting the moving window of the thermal data over by one voxel corresponds to a different grain size distribution, one voxel over in the microstructures.

[2] The moving window approach used here is not the only approach to solving the variable input size problem, or for making local predictions; CNNs and U-nets can both be made to accept variable input sizes, but would come with their own trade-offs.

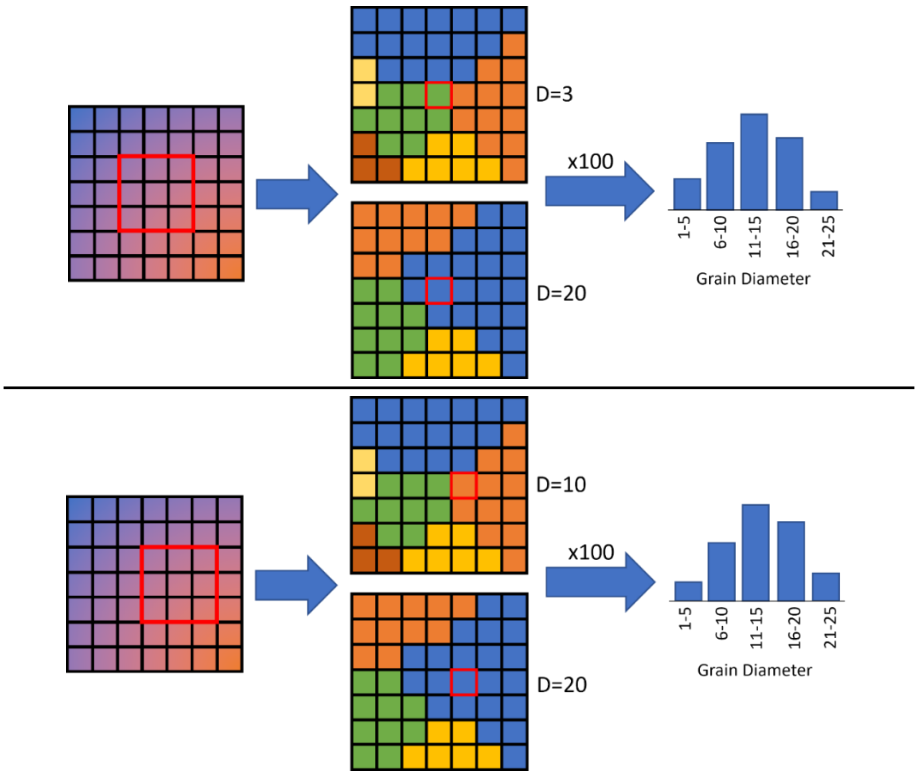

Figure 10 – Illustration of the relationship between the thermal data, simulated microstructures, and microstructure statistics. Top) The relationship between the moving window of the thermal inputs, the point of interest of the microstructures, and the statistics provided at that point by an ensemble of simulations. Bottom) Illustration of how moving the window changes the point of interest and the corresponding statistics.

While the need for ensembles increases the computational cost of training the model, this approach is beneficial for a problem which has not been addressed yet: neural networks typically need very large amounts of data to learn from. Through this moving window approach, we now have a distinct datapoint, consisting of a thermal field and microstructure distribution, for each voxel that has been processed in the simulated LPBF process. While this is an unusual way of representing microstructures, ensembles are a mainstay of probabilistic simulations, particularly for weather forecasting [129–131].

This turns a single LPBF thermal process simulation, with one set of process parameters, along with the corresponding hundred or so microstructure simulations that might be needed for an ensemble, into tens of thousands to millions of datapoints. By contrast, if we were only to simulate each set of process parameters once, and create one microstructure distribution from the whole simulation, we would need tens of thousands of such simulations for an equivalent amount of datapoints.

The trade-off of this approach is that for the same number of LPBF simulations used to generate training data, we will have significantly less coverage of the LPBF process parameter space. While this

appears problematic, as the end goal of the model is to make predictions for new LPBF process parameters, one of the driving assumptions behind the framework presented here asserts that this lack of variation in LPBF process parameters does not significantly limit the model's effectiveness.

Specifically, because the surrogate microstructure model is never directly exposed to the LPBF process parameters, but is instead exposed to the thermal parameter space, we only need variation in the thermal parameter space for the model to learn the required relationships. We assume that, while it is likely that more variation in the LPBF parameter space would expose the surrogate model to a larger region of the thermal parameter space, the process is complex enough that the thermal characteristics within each simulated LPBF process vary despite the process parameters remaining constant[3]. This is because, while the laser's path is often a repetitive pattern, and we use a constant laser velocity and power, among other things, the process is very dynamic and occurs in a finite domain. For example, as the printing process proceeds on each layer, the temperature of the powder bed rises, and the cooling provided by the base plate diminishes as more layers are added. This variation, combined with the moving window approach proposed here, means that the model will be exposed to a variety of thermal characteristics while learning from each simulated combination of LPBF process parameters. We expect that this, combined with similar variations in the resulting microstructure simulations and statistics, will result in the creation of a sufficient amount of variation for the surrogate model to learn the complex relationships between the two.

Now that we have established the framework for creating our model, including the need for a thermal process model, microstructure simulations, and voxel specific microstructure statistics, we can discuss these in more detail. First, we will discuss the thermal model, and introduce how we condensed the results of our thermal simulations into data that our model could use. Then we will give details on the microstructure simulation method and code we selected, and explain how we created our voxel specific statistics. Finally, we will present a method for visualizing this unusual information. Discussion of the surrogate microstructure model itself will be left for the next chapter.

[3] Using varying process parameters within each simulated process would provide more variation in the training data, even with the same number of simulations, but could also miss relationships that appear with constant LPBF parameters. This was not explored as part of this research.

2.3 Thermal Model and Data

For this research, the thermal model serves two similar but distinct purposes: informing our microstructure simulations, and providing information as an input to our microstructure surrogate model. During the process of selecting a thermal model we prioritized the speed and simplicity of the model, while also considering how limitations of surrogate models might interact with our microstructure model. While there exist surrogate thermal process models which can produce results very quickly [132], they often do so by taking very large time-steps or making approximations. These are potentially problematic, as large time steps can create artifacts in the data due to what are effectively discontinuities in time, and surrogate models which rely on approximations may be too simplified to produce the data we need for training our surrogate model. For this reason, the decision was made to instead use a simple thermal simulation in the form of a reduced order finite difference model. This model uses homogeneous thermal conduction with calibrated effective material properties to approximate the temperature fields produced by more complex simulations. Such models are computationally cheap to run and have been shown to have reasonable accuracy within ranges of LPBF process parameters that avoid potential defect forming process regimes like keyholing [86].

The selected thermal process model uses the calibrated effective material properties found by Wolfer et al for a similar reduced order conduction only process model [86], shown in Table 1. The numerical method we use is a 2^{nd} order finite difference model with a uniform rectangular mesh with 5µm voxels, and is 1^{st} order accurate in time with 1µs time steps. The laser is modeled as a 2D Gaussian distributed heat source on the top surface of the active print layer, with no explicit cutoff for the bounds of the laser. All material is modeled as homogeneous with a constant thermal diffusivity for conduction, and no convection or emitted radiation. The boundary conditions are a constant temperature on the bottom surface, with insulated walls, and an insulated top – with the exception of the incident laser source term.

Table 1 – List of physical properties used by the reduced order thermal model for LPBF manufacturing of 316 Stainless Steel.

	Value	Units
Density	7.79E-15	kg/µm3
Thermal Conductivity	2.025E-5	W/µm*K
Specific Heat	470.6	J/kg*K
Melting Temperature	1723.0	K
Solidus Temperature	1673.0	K
Absorptivity	0.33	

The laser position is determined by a pre-calculated path file, which prescribes the laser position, the laser power(including whether it is on or off), and the length of each time step used by the simulation. These are precalculated in their entirety to make ensuring exact replication between the standalone thermal process model and the version implemented in the microstructure simulation code easier. Because the length of the scanlines or the hatch spacing is not always a multiple of the distance the laser moves in 1µs, the algorithm inserts shorter timesteps when this occurs.

While the limitations of the thermal process model we use here could limit practical applications of our surrogate model, the thermal model can be freely swapped within our framework. Additionally, as we are still exploring the viability of the proposed method for microstructure modeling, it is not necessary to have a thermal process model which maintains accuracy for all possible combinations of process parameters. Instead, we will primarily focus here on modeling the small region of LPBF process parameter combinations, within the vicinity of the process parameters used when calibrating the selected model.

2.3.1 Thermal Characteristics for Model Input

Because the purpose of the surrogate microstructure model is to be faster than a microstructure simulation, it would not make sense to process every time-step of the thermal process simulation when making microstructure predictions, as this would be a comparable computational expense to the existing microstructure simulations. Instead, we treat our thermal model as a standalone part of the prediction process, and use a set of representative thermal characteristics, calculated by the thermal model, as inputs to the surrogate microstructure model. For this purpose, nine representative thermal characteristics, shown in Table 2, were chosen with each either having known direct relationships with

microstructure formation during solidification, or which provide a representation of the heating and cooling cycles which the material undergoes during the LPBF process. While it was not known at the time that these thermal characteristics were selected, if these thermal characteristics provided sufficient information for making accurate microstructure predictions or if any were superfluous, we analyze the importance of each of these thermal characteristics to the surrogate model in Section 5.2.

Table 2 – Summary of the thermal characteristics used as surrogate model inputs.

Characteristic	**Units**
Thermal Gradient in x-direction at solidification time	K/µm
Thermal Gradient in y-direction at solidification time	K/µm
Thermal Gradient in z-direction at solidification time	K/µm
Cooling Rate at solidification time	K/µs
Time in nucleation temperature range	Time steps [count]
Time between solidus and liquidus temperatures	Time steps [count]
Time between annealing and liquidus temperatures	Time steps [count]
Maximum reheat temperature after solidification	K
Number of melt cycles	Count

The first thermal characteristics, which are known to have a strong relationship with microstructure morphology, are the thermal gradients G_{sol}, Equation (1)**,** and the cooling rates at the time of solidification $\dot{T}_{sol}$, Equation (2).

$$G_{sol} = \nabla T|_{sol} \qquad (1)$$

$$\dot{T}_{sol} = \left.\frac{\partial T}{\partial t}\right|_{sol} \qquad (2)$$

$T(\vec{x}, t)$ is the full temperature field, G_{sol} is the thermal gradient at the time of solidification, and we record the terms for each voxel only on the time step on which their temperature drops below the solidus temperature. These terms are commonly used in analysis of LPBF due to their relationship to the columnar-to-equiaxed transition (CET), and the prominence of columnar grain growth in the LPBF process [41,52,69,73]. This relationship is generally analyzed using the thermal gradient and the velocity of the solidification front v_{sol}, instead of the cooling rate, but it can be shown that these three values are directly related using the relationship in equation (3). Because the finite difference method used here provides us with no direct way to track the solidification velocity, we instead use the cooling rate.

$$v_{sol} = \frac{\dot{T}_{sol}}{G_{sol}} \qquad (3)$$

While the magnitude of the thermal gradient is typically used for such analyses, here we record the 3 directional components independently to allow the surrogate model to learn any relevant spatial relationships. Because grains grow in the direction of the thermal gradient during solidification, the direction of the thermal gradient has a large influence on the relationships between neighboring grains, and the resulting size and shape of those grains.

In contrast to the thermal gradient which conveys spatial information, the remaining thermal characteristics were selected to be representative of the complex thermal histories experienced by every voxel in the simulation, as illustrated in Figure 11. While the cooling rates do provide information about the thermal histories, we can see that they are much more complex than this single value can capture.

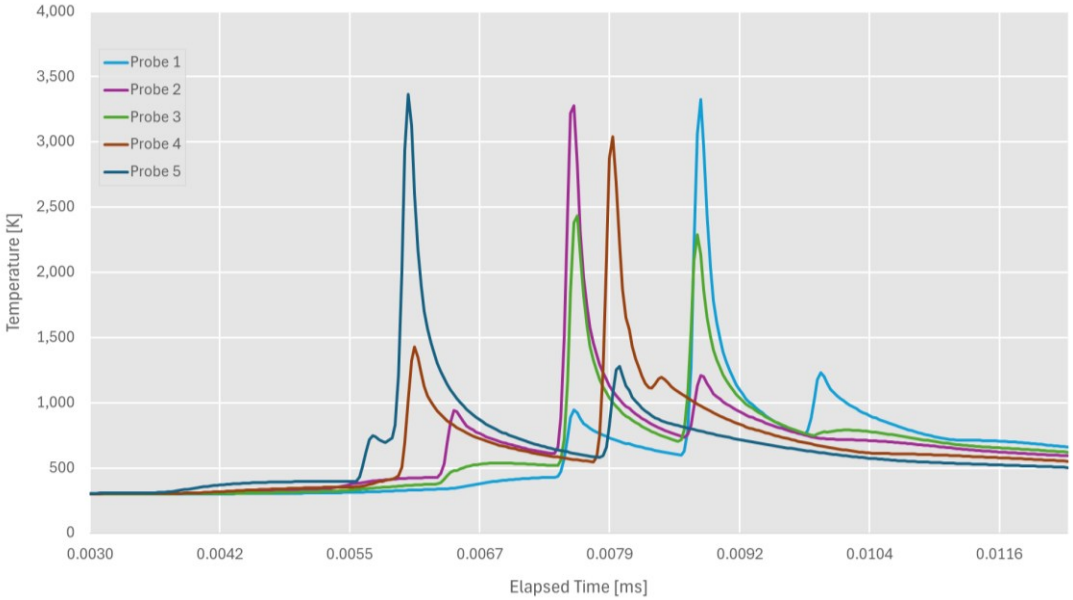

Figure 11 – A snapshot illustrating some of the complexities of the thermal histories experienced by points during the LPBF process, with repeated heating and cooling cycles. The actual process results in significantly more heating and cooling cycles.

The first three of these characteristics are the amount of time spent above three different temperatures: the nucleation temperature, the solidus temperature, and an annealing temperature. The nucleation temperature was chosen as it influences how likely grains are to nucleate in the mushy zone, and is directly modeled in our chosen microstructure simulation, as discussed in the next section. The nucleation time is defined as the amount of time each voxel spends below the liquidus temperature, but above the nucleation temperature, which we define here as the liquidus temperature minus the

undercooling temperature. The solidus temperature was chosen as it is the cutoff temperature below which grain growth is solid state, which involves different dynamics than nucleation and solidification-based growth. This solid-state grain growth regime is represented to the model by the annealing time, or the amount of time spent below the solidus temperature, but above an annealing temperature.

While the term "annealing temperature" usually refers to a prescribed temperature at which something is held for annealing during a manufacturing process – to achieve a desired change in the microstructure – rather than a material specific temperature, here we use it to refer to the temperature below which solid state grain growth is not expected to be significant. To determine the annealing temperature, we analyzed the standard equation for isothermal solid-state grain growth, equation (4) [93,133], which calculates the mean grain size of a sample held at a constant temperature, to determine the temperature below which solid state grain growth becomes negligible.

$$r^n = K_0 \exp\left(-\frac{Q}{RT}\right) t \qquad (4)$$

r is the grain size, K_0, Q, and n are growth rate parameters, R is the gas constant, and t is the elapsed time at the given temperature T. With the growth rate parameters used by the selected microstructure simulation code, discussed in the next section, we can solve for the rate of grain growth as a function of temperature, as depicted in Figure 12. Because this is an exponential curve, there is no obvious point to use as a cutoff below which grain growth stagnates. However, we can approximate the "knee," the point at which the value begins increasing rapidly, by finding the temperature at which the derivative of the growth rate is equal to the average slope across all values. This process, and the resulting cutoff point, are also depicted in Figure 12. Inspecting the numbers at this point reveals that the growth rate at this temperature is approximately one tenth of the maximum growth rate, which we deemed to be a reasonable cutoff.

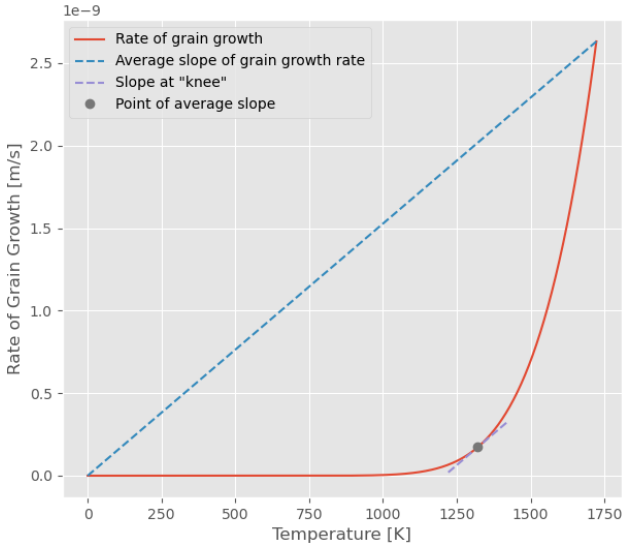

Figure 12 – Plot showing the relationship between temperature and rate of grain growth, with the location of the "knee" labeled.

While we refer to these thermal characteristics as “times,” the choice was made to represent these thermal characteristics using the number of elapsed time steps in these temperature ranges. This introduces some error to these values, as not all time steps in the thermal simulation are equal; however, this choice was made to slightly reduce the computational load of these thermal characteristics, and minimize any potential for computational error which can be introduced by adding very small numbers repeatedly. This tradeoff in errors may be worth exploring in the future, as the influence of small perturbations on machine learning models can be hard to predict, but this is not explored in this work.

The last two thermal characteristics are the number of times each voxel has melted – or the melt count for short – and the maximum temperature to which each voxel is reheated after solidifying for the last time. The melt count was chosen to provide a representation of the temporal relationship of the other thermal characteristics between neighboring voxels in a way that is largely unique to AM processes, where individual regions may undergo many melting and solidification cycles. This was included under the assumption that the model may be able to use this information to prioritize which neighboring voxels may be the most important for determining the microstructure at the current voxel. It is not obvious what the exact relationships might be, but it can be seen in Figure 13 that even in a very basic single layer print, the melt count provides useful information about which voxels have undergone

multiple solidification cycles from adjacent passes of the laser, something that is not clearly provided by other thermal characteristics.

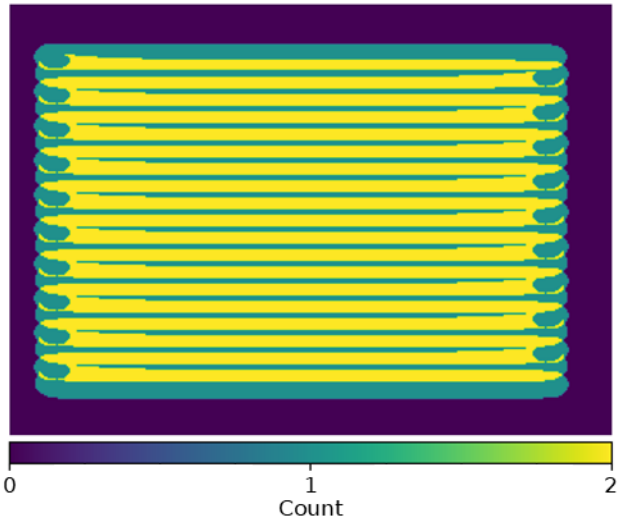

Figure 13 – Number of times each voxel on the top surface of a 1-layer raster scan has melted.

Finally, the maximum reheat temperature captures one of the complexities of the repetitive heating cycles regions undergo as the laser passes within their vicinity. While this metric is only capable of capturing the most prominent of these reheat cycles and does not capture multiple heating cycles to similar temperatures, it is hoped that the combination of all of these thermal characteristics will provide the model with enough information to have an approximate understanding of the complete thermal history.

While the calculation of these thermal characteristics adds computational cost to the thermal model, the overall impact of these calculations on the runtime of the thermal model is small. Through the use of careful optimizations which limit these calculations to only the voxels where they are relevant, the computational cost of calculating these nine thermal characteristics is much lower than the cost associated with the microstructure simulations. In addition to having lower computational complexity, these calculations are much more amenable to GPU acceleration than the complex stochastic logic often found in microstructure simulations.

2.4 Microstructure Simulations and Data

To generate the microstructures for training the model we use SPPARKS (Stochastic Parallel PARticle Kinetic Simulator), an open-source code developed by Sandia National Laboratories, that

contains kinetic Monte Carlo algorithms for simulating microstructure evolution [126,134]. A variety of modules have been developed for SPPARKS, which allow for the modeling of many different stochastic processes. For the purposes of this research, we use the finite-difference thermal additive manufacturing model developed by Rodgers et al [93,135]. This module implements a basic finite-difference thermal solver to model the thermal evolution of the LPBF process, and is calibrated for 304 and 316 stainless steels.

This module models the nucleation and growth of grains in the mushy zone, and solid-state grain growth which occurs during the LPBF process. This model does not model the crystallographic orientation of the grains, but instead tracks an artificial grain ID. At the beginning of the simulation, the grain IDs for all voxels in the simulation domain are randomized; these are randomized again when a voxel is melted. During solidification, the voxels in the mushy zone have a chance of nucleating, and the voxels around them have a chance of switching to the ID of a nucleated grain or existing solidified grain. After dropping below the solidus temperature, the grains are allowed to continue growing. An example of a resulting microstructure can be seen in Figure 14, which shows the cross section of a 1mm cube, where the colors correspond to the grain ID, and the noise around the edges of the figure are the unconsolidated powder which is not part of the cube. In this figure we see the formation of large columnar grains, which continue across multiple layers of the print, characteristic of the LPBF process.

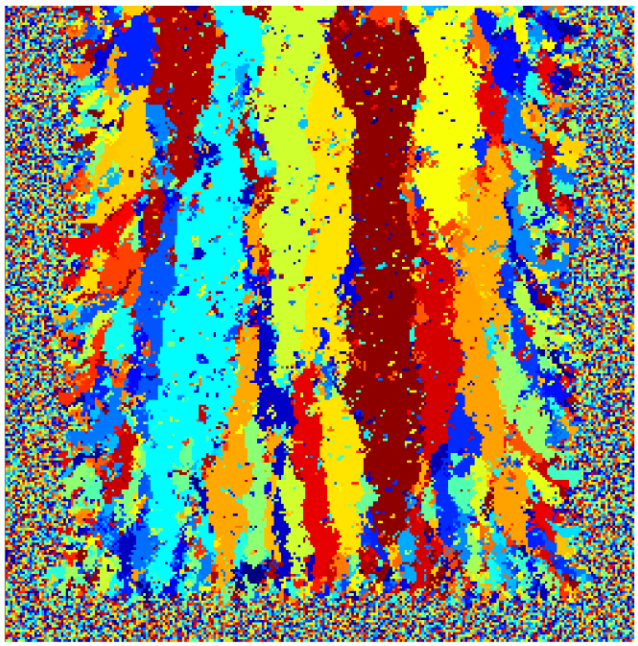

Figure 14 – Example microstructure of cross section of a 1mm cube, including the unprocessed region around the cube. Note the large columnar grains in the central region, and the smaller grains around the edges, with the random noise around the edges random noise representing unprocessed powder.

This SPPARKS module includes a finite-difference thermal model which implements convection, conduction, radiation, and state specific thermal properties. However, for the purposes of this research, this model was simplified to the thermal model outlined in the previous section. In addition to the need to match the standalone thermal model, this simplification is useful here as several thousand microstructure simulations are needed to generate our data, so reduced computational costs are magnified significantly. While this may have had some impact on the calibrated relationships between the process parameters and the resulting microstructures produced by the SPPARKS model, it does not influence the kinetics of the KMC solidification model itself. As it is these relationships between the thermal process and the microstructure we hope to learn, this is not a primary concern for the purposes of this research.

As previously mentioned, the data generation process for the surrogate microstructure model framework involves running an ensemble of identical simulations for each set of process parameters, taking advantage of the stochastic nature of the microstructure model to create variations in the simulated microstructures. This mirrors the outcome of the real LPBF process, where it is unlikely that two identically printed parts would ever have identical microstructures, though the real LPBF process has additional sources of randomness due to uncertainties inherent to LPBF machines. These sources of

randomness, such as variations in laser power and speed, and fluctuations in the ambient temperature and airflow within the machine are not included at any stage of this research. Incorporating these into future data sets may help the model to better understand the influence of small LPBF parameter variations on the resulting microstructures, allowing for better uncertainty quantification, but would likely require a new ensemble for every variation in the parameters.

2.4.1 Voxel-by-Voxel Microstructure Statistics

After running an ensemble of SPPARKS microstructure simulations, the simulation results must be post processed into voxel-by-voxel grain morphology statistics which the surrogate model can use for training. As mentioned when introducing our framework, this requires calculating the microstructure statistics for each voxel in the simulation independently, taking the statistics for each voxel across multiple simulations. Here we will outline this process and the resulting statistics in more detail. Additionally, we will discuss some of the peculiarities which arise from our method of representing microstructure statistics and relate them back to more conventional representations.

While the proposed framework should be compatible with any single grain morphology metric, or combination thereof, this work focuses specifically on the grain size distribution. Rather than attempting to represent the distributions via population statistics like a mean and standard deviation, which would make for a relatively straightforward regression problem (though the relationships would likely still be quite complex), we opt to represent it as a full discretized probability distribution. In this case, we do so using 51 bins[4], with 50 equisized bins that span from 0-400µm, and with any grains larger than 400 µm counted towards the 51st bin. While this representation is limited in its precision and cannot accurately represent probabilities beyond a certain grain size, the use of discretized probability distributions lets our model learn arbitrary distributions that are not well represented by just population statistics. Additionally, if we were to extend this representation to multiple grain morphology metrics, such as the grain size and aspect ratio, in the form of higher dimensional distributions[5], it would be possible to see the probability that a voxel has both a given grain size and aspect ratio, something that would be harder to represent with population statistics. Likewise, such a higher dimensional

[4] It was not actually our intention to use 51 bins, this is a case of an off by 1 error caused by the fencepost problem [136], though the impact of this was seen as too negligible to be worth changing it.
[5] As opposed to multiple separate distributions.

representation would make it possible to represent the probabilities of all possible crystallographic orientations simultaneously.

The 51 bins were chosen to match the data observed during this research as best as possible while keeping the data to a reasonable size, though the number was chosen out of convenience, as we could not find any appropriate heuristics for the number of bins to use with our specific dataset. While more bins could span a larger grain size range or provide better granularity, and are viable with access to sufficient computing hardware, the interplay between the number of bins and the number of samples in an ensemble must also be considered. If the grain size range covered by a single bin becomes small enough, the resulting distribution may appear excessively spiky if there are an insufficient number of simulations in each ensemble[6]. While it's possible that the model may still learn the underlying relationships from such spiky data, we opted to keep the number of bins at 51 after observing that the resulting discretized distributions appeared reasonable for our data, which generally consisted of 100-200 simulations per ensemble.

The first step in the process of calculating the grain morphology statistics is to create an array the same size as the simulation mesh, but with an extra dimension containing the 51 bins in which we will tally our results. After this array has been filled in, we will be able to inspect any voxel and see the grain size distribution associated with that voxel, as predicted by all the microstructure simulations in our ensemble.

In the case of a multi-objective model, or multi-valued metric like crystallographic orientation, it would likely be best to make it such that each voxel has an N^M array associated with it, where N is the number of bins per value and M is the number of values. This would allow for prediction of the probabilities of the combined metrics (e.g. the probability of a given grain size at a given aspect ratio), rather than separate probabilities.

To generate these statistics from our ensemble, we start with a single microstructure simulation result, on which we run an image processing algorithm to find and label each grain in the microstructure by finding all connected voxels with the same grain ID; for this we use the label function implemented in

[6] We will show in sections 4.2 and 4.3 that this is less of a concern than one might expect for this model.

scikit-image [137]. The number of voxels in each grain is then counted, and the equivalent spherical diameter (ESD) of each grain is calculated using equation (5)

$$ESD = \sqrt[3]{\frac{3}{4\pi} \times V_{voxel} \times n_{\text{voxels}}} \qquad (5)$$

where V_{voxel} is the volume of each voxel ($V_{voxel} = 125\mu m^3$ here), and n_{voxels} is the number of voxels belonging to a given grain. Each grain diameter is then sorted into the corresponding grain size bin, as shown in Figure 15, and each voxel belonging to the grain has 1 added to the tally of the corresponding bin. This is repeated until all grains in the simulation have been accounted for, with the entire process repeated for each simulation in the ensemble.

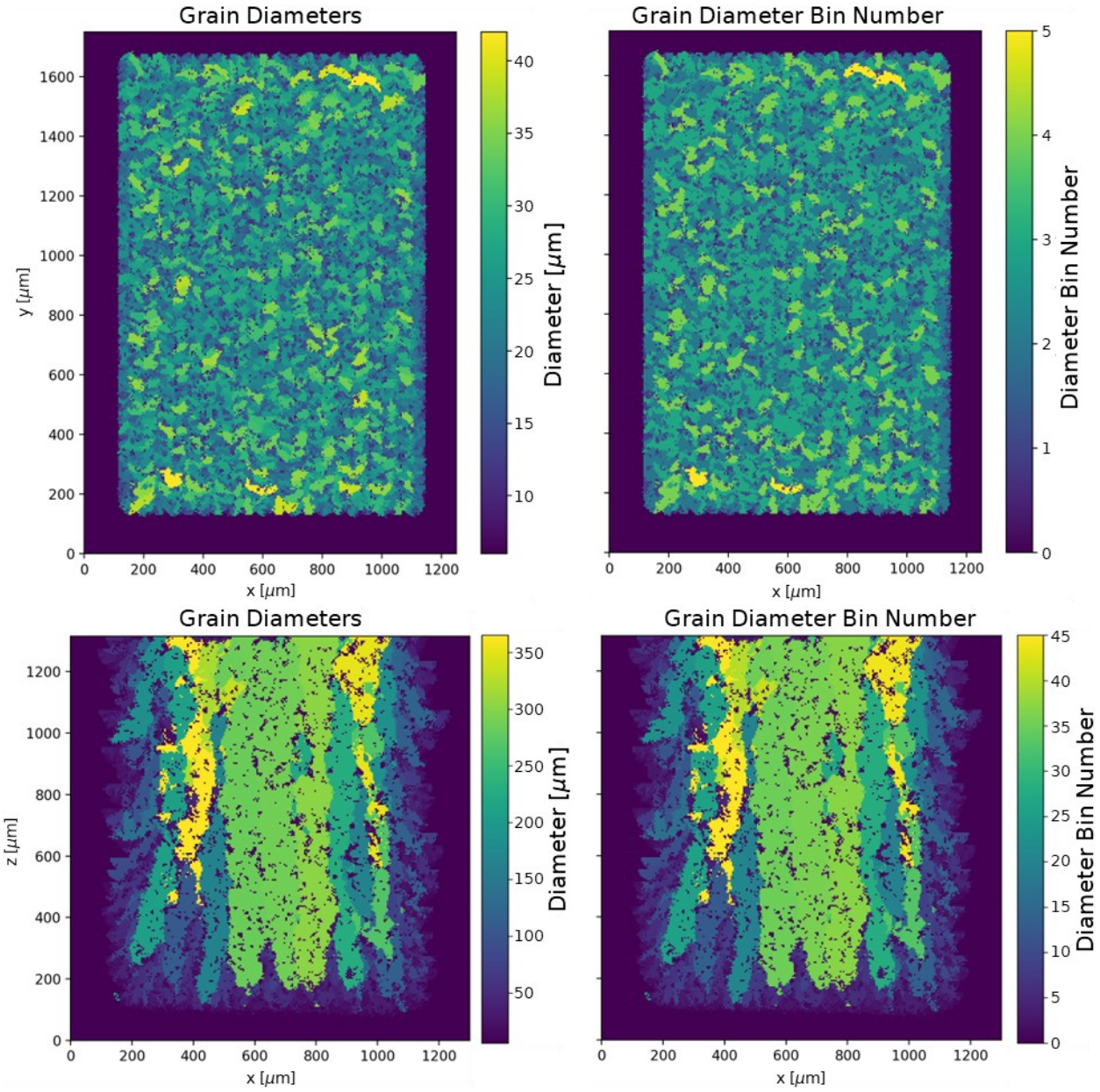

Figure 15 – Plots showing the diameters of each grain (left) and the corresponding bin number of each grain (right) for the top surface of a one-layer print (top) and the cross section of a 1mm cube(bottom).

As an aside, this way of visualizing the microstructures makes it abundantly clear that there are a large number of small grains entrapped within the larger grains, particularly within the 1mm cube. While it is not immediately clear how much of this phenomenon is an artifact of the simulation methods used by the SPPARKS code, this phenomenon can be observed within experimental EBSD results such as those by Roehling et al in Figure 16 [138]. This is not of particular interest to the research presented here, but has consistently drawn curiosity, so we felt it was best to note it here.

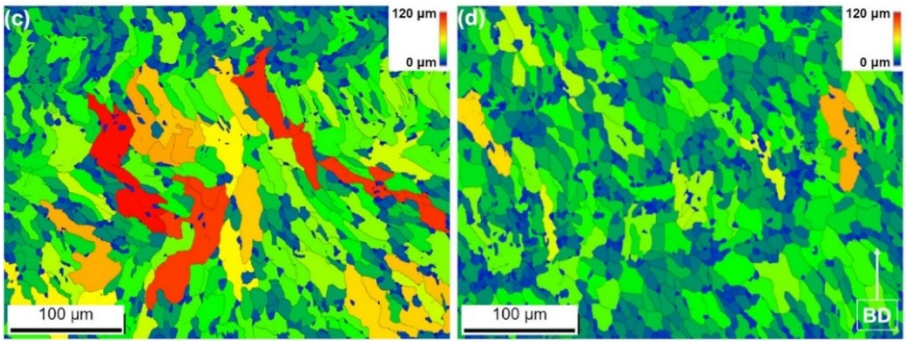

Figure 16 – EBSD images of LPBF microstructures by Roehling et al, which appear to show small grains entrapped in larger grains [138], reproduced under Creative Commons [139].

Unlike with the single simulation binning shown above in Figure 15, which are quite similar to a normal microstructure plot, the full ensemble microstructure statistics are quite different from the types of microstructure statistics normally used in materials science and the data itself is challenging to visualize. Traditionally microstructures are visualized via the same types of plots we have shown up to this point, where a slice of an individual microstructure can be measured and the statistics analyzed. In contrast, our method produces statistics for every voxel/pixel of the microstructure, in the form of a grain size probability distribution discretized into 51 probability values. The most straightforward way of visualizing this data is to pick a grain size bin of interest to plot, examples of which can be seen in Figure 17, which shows the probability that each pixel belongs to a grain of the specified size. Here we are plotting the same two prints shown in Figure 15, but instead showing the probability of each voxel belonging to a grain of a given size.

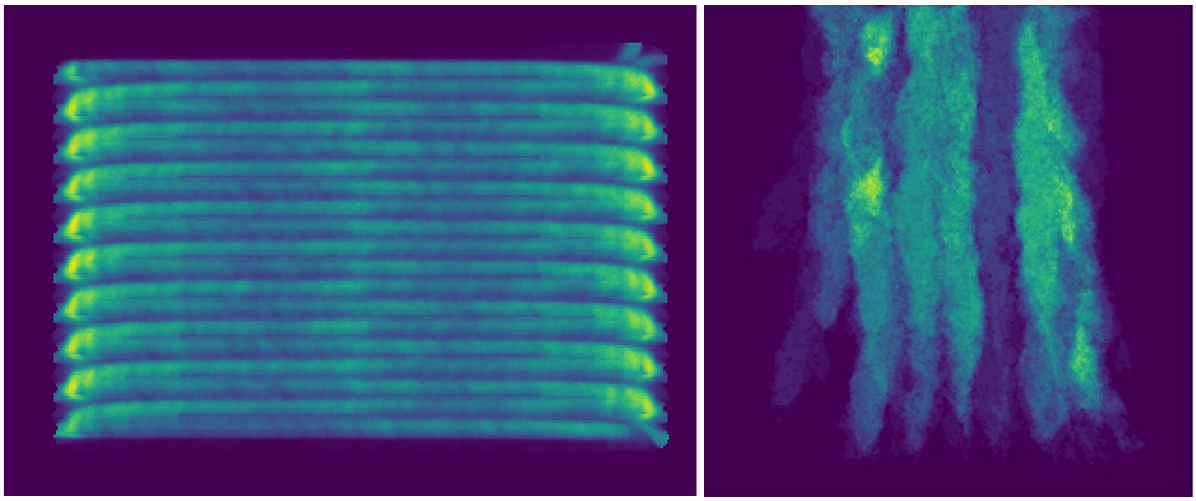

Figure 17 – Relative probabilities that each voxel belongs to a 32-40um grain on the top of a 1-layer raster scan (left), and a 240-248um grain on the cross section of a 1mm cube (right).

While contiguous regions of high probability may correspond to voxels belonging to the same grain some of the time, this is not necessarily true. Instead, these regions tell us that grains of the indicated size are likely to occur in this region, without telling us specifically which voxels a given grain occupies. As an example, a large region could hypothetically have a 100% probability of very small grains, indicating that the region is always full of some unknown arrangement of many small grains. By contrast a relatively small region could have a 100% probability of a grain larger than the region, indicating that this region always has one or more grains of the indicated size completely overlapping it. While the magnitudes in Figure 17 are much lower than 100%, the apparent striations in these plots are similar to the latter situation, where columnar grains consistently form in a given direction but at slightly different spots, resulting in regions which are commonly overlapped by these larger grains. Because these striations are caused by similar grains nucleating in slightly different locations, due to the stochastic nature of solidification, it is expected that they will decrease in apparent magnitude as the number of samples in an ensemble increases.

These emergent patterns can provide interesting insights into the microstructures created by the LPBF process, particularly when looking at trends as the grain size being plotted changes. For example, Figure 18 shows the probabilities of 64-72um, 160-168um, and 272-280um grains in the cross section of a 1mm cube, revealing that the outermost edges and bottom are likely to have small grains, with large grains in the center, and medium grains in between. It should be noted that this is only a representation of the microstructures predicted by SPPARKS, and that some of the increased probability of smaller grains near the edges is likely due in part to the random initialization of the grains in the powder bed and the ability for grains to nucleate from these small grains.

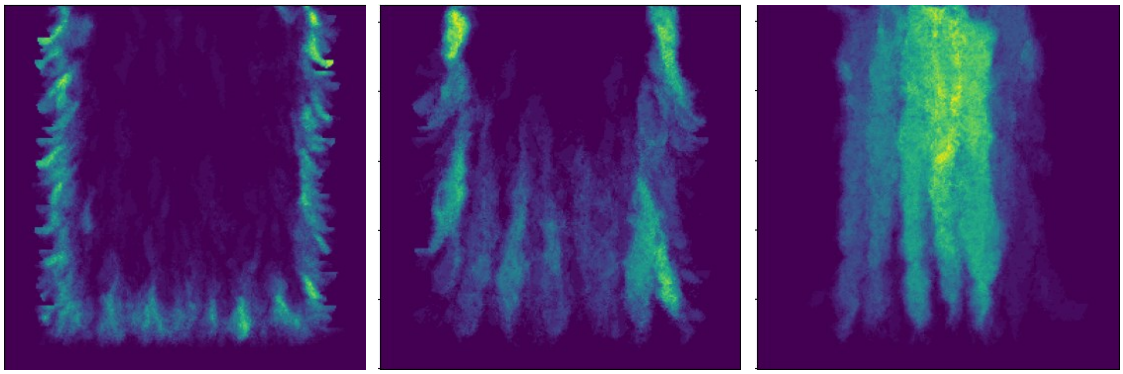

Figure 18 – Grain size probability fields of the cross section of a 1mm cube, showing the relative probability that each voxel belongs to a 64-72um, 160-168um, and 272-280um grain respectively.

Because we are only visualizing one bin at a time, these trends sometimes appear a bit inconsistent due in part to the stochastic nature of the underlying microstructures resulting in noisy distributions. However, the consistency of some of these trends, noisy as they may be, particularly for the large columnar grains which extend through the majority of the cube in Figure 18, point to there being some consistency in the nucleation points of certain groups of columnar grains, though the probabilities are low enough that they are not guaranteed. It is not immediately clear if these patterns are reflective of reality, or if they are an unknown artifact of the SPPARKS simulations. While it might be possible that the consistent heating and cooling patterns at these points across all the simulations of each ensemble are mimicking something akin to the carefully controlled conditions used for forming single crystal microstructures [98,99], much larger ensembles would be needed to rule out them being an artifact of our limited sample size, and more investigation would be needed before anything could be said definitively about these patterns.

In contrast to the splotchy and noisy patterns of the previous figures, Figure 19 shows that large ensembles significantly reduce the splotchiness of the statistics due to a decreased influence from the stochasticity of the underlying data, particularly for the simplest microstructures, such as those that occur with single layer prints. With this decreased stochasticity, the statistics appear to have a repetitive pattern that is consistent with the repetitiveness of the raster path taken by the laser, which is contradictory to our assumption that the proposed moving window approach will expose the model to a large variety of microstructure distributions. However, Figure 19 also demonstrates that the underlying distributions are not as repetitive as they first appear in the image-based plot. To show this, we take each column of the image and plot the magnitude of the probability on the left, and do the same with the rows on the bottom plot. While this does reveal that there are indeed patterns, the inconsistencies between the peaks on the left and the lack of identical lines on the bottom reveal that there are small changes in the microstructure distributions that are not readily apparent via the color-scale used in the image plots. If one considers that there are 50 other channels (i.e. bins) which we are not plotting here, though in this single-layer case only a few of the smallest grain bins have a non-zero probability, it is easy to see how these small variations might add up in a meaningful way.

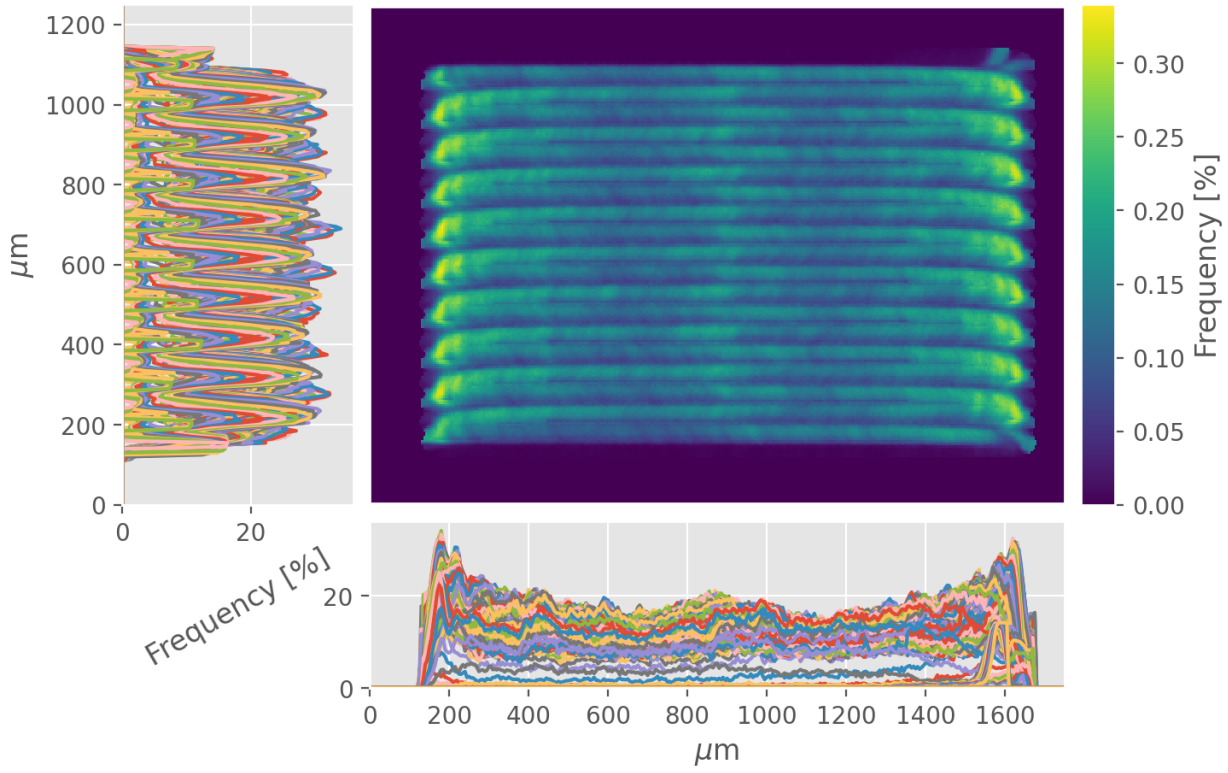

Figure 19 – Plot showing the apparent repetitive pattern of the frequency of occurrence of 32-40μm grains on the top surface of a 1-layer raster scan across two thousand simulations (top right). The plot on the left shows the actual magnitude of the frequency for each column of the image, with each line representing a column, and the plot on the bottom shows the same but with rows.

Though we use a large ensemble here to minimize the influence of the stochasticity of the underlying data, it is likely that some of this variation is still due to the nature of sampling a stochastic process with a finite number of samples. However, it is our assumption that a meaningful portion of these variations stem from the small changes in thermal characteristics of the solidification process that occur as the process progresses. Such changes are expected from the simulation domain accumulating heat, the melt pool getting closer and farther from the various boundaries of the simulation, and the complex dynamics of the LPBF process which allows for previously processed regions to influence future solidification behavior and future processing to influence grain growth. It is these variations in both the thermal characteristics and microstructure statistics, in addition to the stochastic sampling variations, which we hope to both capture and use to our advantage with the moving window approach proposed as part of our framework for the creation of the surrogate model.

Because these spatially defined statistics are an unusual way of representing microstructure information, new methods for predicting material behavior directly from these statistical representations

may need to be developed. While we do not explore the implications of this new representation of microstructures as it relates to material properties in this research, it may be possible to use the framework developed for the creation of this surrogate model to create a model capable of predicting spatially defined material properties from these microstructure statistic maps. Additionally, as research in the field of additive manufacturing progresses, this method of representing microstructures may be useful for understanding the uncertainties associated with the process.

2.4.2 A Better Way to Visualize Our Data

Because our data is high-dimensional, the method of visualizing the data used in the previous figures is limited in its ability to illustrate trends in the data. To better visualize the underlying trends, a method which uses the 3 color channels used by electronic displays – red, green, and blue (RGB) – to show three probabilities simultaneously was devised, where the magnitude of each individual color indicates the corresponding probability. For the purposes of consistency, and clarity for those familiar with RGB color, the smallest grain bins being plotted will always be represented by the red channel, and the largest by the blue channel, with green in-between. This method allows for the visualization of three grain size bins simultaneously or, more usefully, the visualization of broader trends in the grain size by grouping relevant bins into three super-bins to visualize simultaneously. An example of this method can be seen in Figure 20**,** which shows the three-color statistics from the full ensemble using grouped bins which span all the relevant grain sizes, along with a 5-simulation ensemble and a single microstructure for comparison purposes.

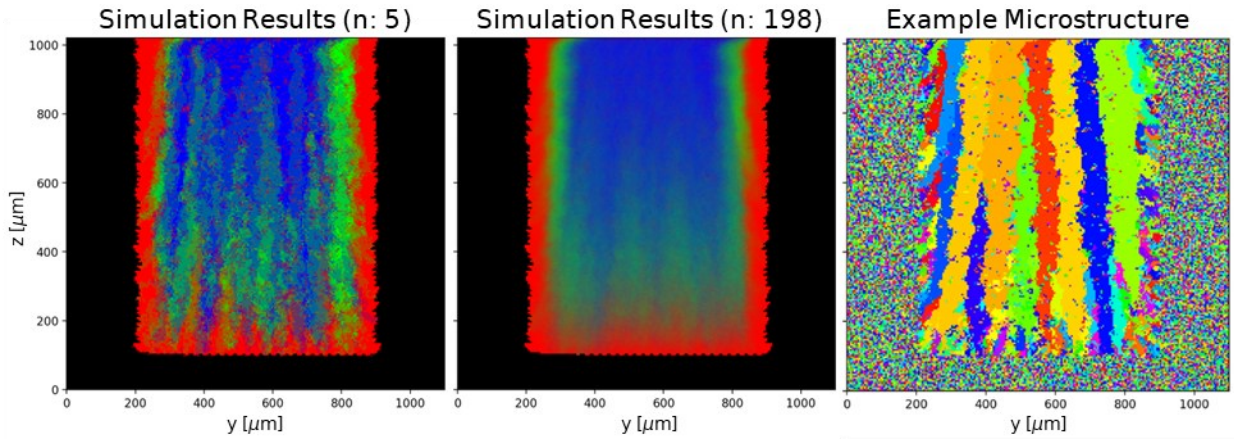

Figure 20 – Comparison between the cross section of a single microstructure and the equivalent statistics. Left) Statistics from an ensemble of 5 microstructure simulations. Center) Statistics from the full ensemble of 198 simulations. Right) Single simulated microstructure for comparison.

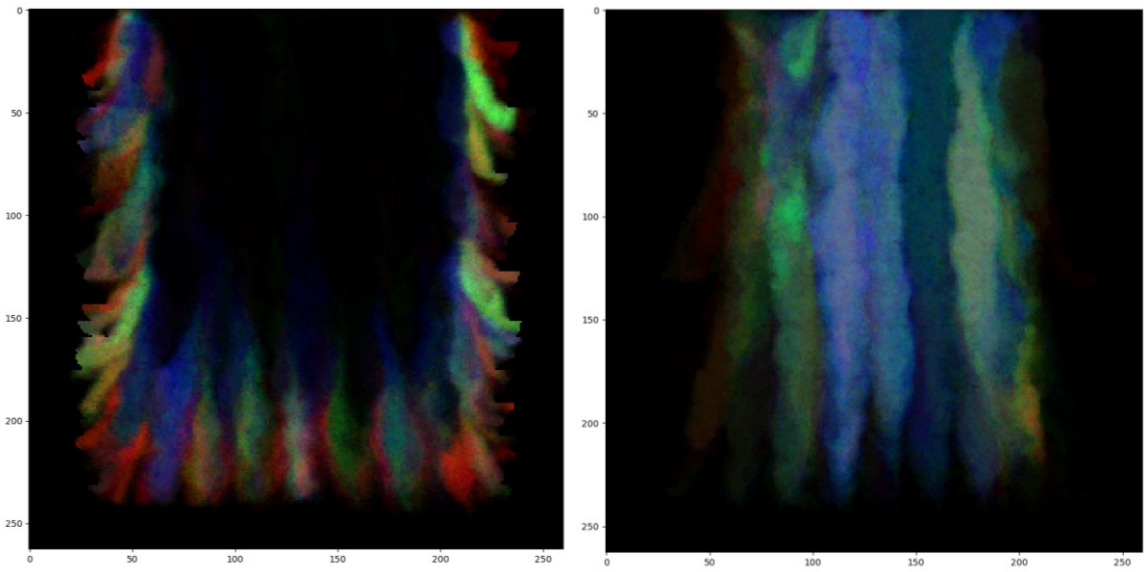

Figure 21 – Plots showing the probability of medium sized grains (left) and large grains (right) for the cross section of a many-layer print, where each of the 3 base-colors (RGB) represent adjacent groupings of 3 adjacent bins each. These smaller groupings of bins provide a more detailed look at the variation present across the part; however, for the purposes of simplicity, we do not explore such representations in this work.

For the purposes of visualizing the results of this work, we primarily use three equisized bins that collectively span all bins from the smallest up to the largest grain size bin which had an average probability greater than 0.5% across all voxels in the printed object. This method was chosen to capture the positively skewed nature of the distributions that was generally observed across our data while minimizing the influence of long low-probability tails on our visualizations. However, it was found that on occasion this would result in erroneous visualizations, particularly when visualizing inaccurate model predictions alongside equivalent ensemble-based data. In these cases, it is necessary to create a 2nd visualization which covers the first 99.5% of cases on average. When making comparisons between model predictions and target data, we find that this cutoff approach does a particularly good job of highlighting discrepancies in the data. An example of such a case where this is useful can be seen in Figure 22, where an erroneous cutoff determined using the model's prediction (not shown) results in a large region where all three bins have close to 0% probability, which is remedied with subsequent re-calculation of the cutoff point.

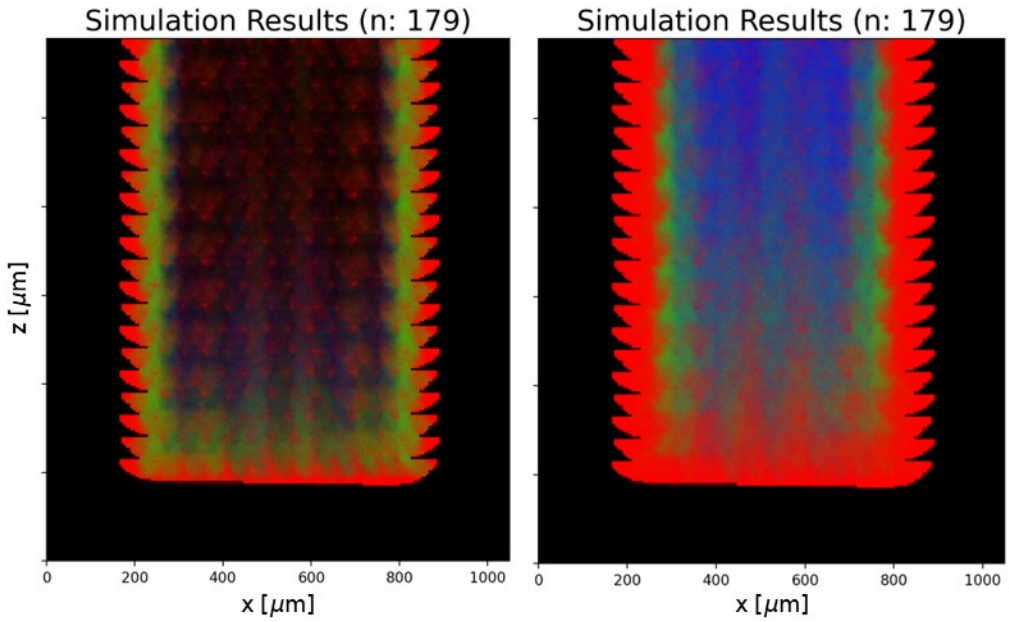

Figure 22 – Three color plot of the grain size probabilities for an ensemble of SPPARKS simulations, before (left) and after (right) adjusting the cutoff point to be 176μm larger using the 99.5th percentile.

Because these plots have 3 color values encoded into every pixel, a standard color-bar is not a sufficient scale for interpreting these plots. Instead, the color wheel in Figure 23 was designed to better show the complex relationships between the colors and values, though it is not possible to show all color combinations in a useful manner on a 2D plot. The color wheel is a combination of three plots, each representing the magnitude of one of the three color-channels, where the smallest bin corresponds to red, the 2nd to green, and the 3rd to blue.

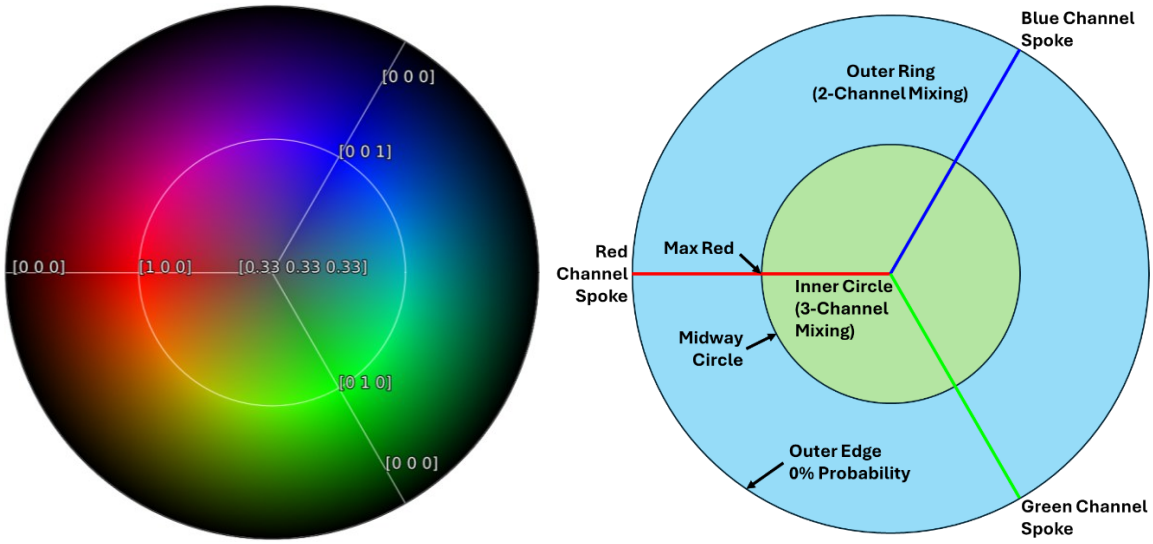

Figure 23 – Color wheel for reading statistics from tri-bin plot, where the 3-value labels indicate the probability associated with each of the three bins, along with a diagram indicating reference points on the color wheel.

This color-wheel can be thought of as 3 separate plots added together, with each showing a different base color, as illustrated in Figure 24. Here we can see that the maximum value of each individual color occurs midway along the corresponding spoke for each color, and represents a probability of 100% for the bin being represented. Between these midway marks and the outside edge of the color wheel, these spokes are purely the one corresponding color, at values between 0 and 1, representing that only the corresponding bin has a non-zero probability.

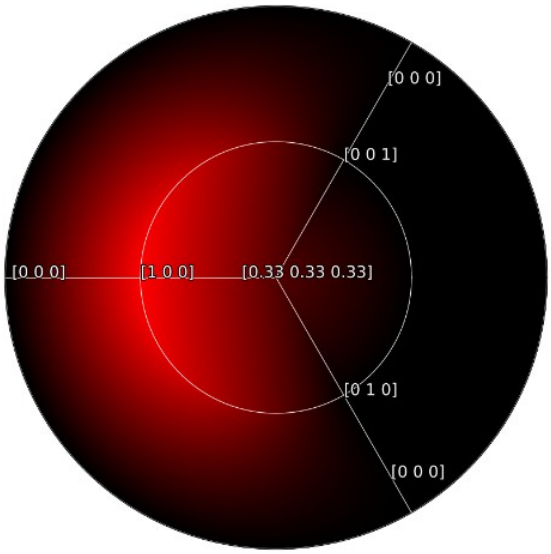

Figure 24 – Red component of the colorwheel.

Outside the circle which represents the midway mark, the value of each color decreases circumferentially from the corresponding spoke, where the value on the spoke at the equivalent radius is the maximum, and the value is always zero by the next spoke. For example, moving circumferentially away from a spoke along the outer edge will always be zero as that is the maximum, and moving circumferentially along the midway circle fades from the maximum of 1 at the spoke down to 0 at the next spoke.

Inside the circle which represents the midway mark, the value of each color has a gradient along the direction of the corresponding spoke, with values matching the circumferential gradient along the midway circle which we just discussed. This gradient decreases to a value of 0.33 or 33% at the center, and fades to zero before reaching the midway circle on the other side of the center mark.

When combined, these add up to the color wheel presented in Figure 23, with the inner circle (within the midway circle) representing the combination of all 3 bins having a non-zero probability, and

the outer ring (between the midway circle and the outer edge) representing both single bins and combinations of 2 bins having non-zero probabilities. As previously mentioned, there is no viable way of representing all possible combinations in a way that can be neatly interpreted, so this does not represent all possible combinations of the 3 probabilities, but should represent the majority that are relevant as probabilities must sum to 1. Additionally, while the color wheel is sparsely labeled, making it almost impossible to pinpoint an exact value, the corresponding RGB plots are intended for use in visualizing trends more than for examining the data in detail.

2.5 Framework Final Words

Now that we have detailed the framework designed to create the model, including the flow of data, the form the data takes, the simulations used to create the data, and the post processing needed to create the data from the simulations, as well as how we can visualize the data in a human readable manner, we can move on to the details of the surrogate model and the process used to create it.

Chapter 3 The Surrogate Model

3.1 Introduction

Now that we have established the tools and framework necessary for our surrogate microstructure model, we can explore the creation of the surrogate model itself. First, we will detail the architecture of the model, which was implemented using the PyTorch machine learning library in Python, along with the training process and the dataset selected to train it. After detailing these, we will explore the meaning of the metric used to measure our model's accuracy as it relates to our data. Finally, we will show how the architecture we based our model on, which was originally created for image classification, was tuned to better achieve our goal of a fast and accurate surrogate microstructure model.

3.2 Model Architecture

Here we will introduce the machine learning architecture selected for use in our surrogate model, and outline the major modifications needed to adapt it to our problem. Because the goal of this research is to find a way to replace conventional microstructure simulations, rather than to develop a new machine learning algorithm, many of these changes are not analyzed in detail. Instead, many of the changes were made on an as-needed basis for the purposes of creating a functional model, and the final resulting model was analyzed in detail in the remaining chapters.

As discussed in Section 2.2, the inspiration for the design of our surrogate model was image classification machine learning models, due to the form the data takes and the spatial relationships of the microstructure formation and LPBF processes. This left us with many architectures on which we could base our model, as there are a variety of image classification architectures which have been shown to perform well, though largely ruled out all machine learning models that are not neural networks. At the time of making this decision, the accuracy of state-of-the-art image classification networks had recently improved dramatically with the advent of the Vision Transformer (ViT) [140], and iterations on it such as the Swin Transformer [141]. The success of the techniques used in the ViT architecture had also inspired machine learning researchers to revisit other image classification architectures such as convolutional neural networks, with ConvNeXt [114], and multi-layer perceptrons, with MLP mixer [115]. While other options were available, these three architectures were the first considered as potential foundations on which we could build our surrogate model, due to their state-of-the-art accuracies.

While all three architectures achieve high accuracies for image classification tasks, this accuracy does not necessarily translate to our microstructure surrogate modeling task, and it is not obvious which will perform best. Transformer based architectures are generally slower than alternatives, making them a less desirable option, as speed is one of the primary objectives for our model. Additionally, while all neural networks can benefit from larger training datasets, transformer-based models often require more data to train than their counterparts, a drawback which could be hard to overcome given the computational costs of our data generation process. After ruling out transformer-based architectures for these reasons, the distinction between the ConvNeXt and the MLP Mixer architectures in their potential application to the surrogate modeling problem was less clear. Ultimately the ConvNeXt architecture was chosen because convolutions have a clearer relationship with spatial data, though it is not necessarily faster than MLP Mixers.

The ConvNeXt architecture consists of 4 stages, as illustrated in Figure 25, with each stage containing a down-sample convolution layer – reducing the size of the image while increasing the number of channels – and a variable number of the primary building blocks. Each of these blocks contains a convolution layer, two linear layers with an activation function in-between, and a layer-norm. For the purposes of our surrogate modeling problem, this architecture was not a plug-and-play solution and needed to be adapted to this use case with the changes discussed and illustrated below.

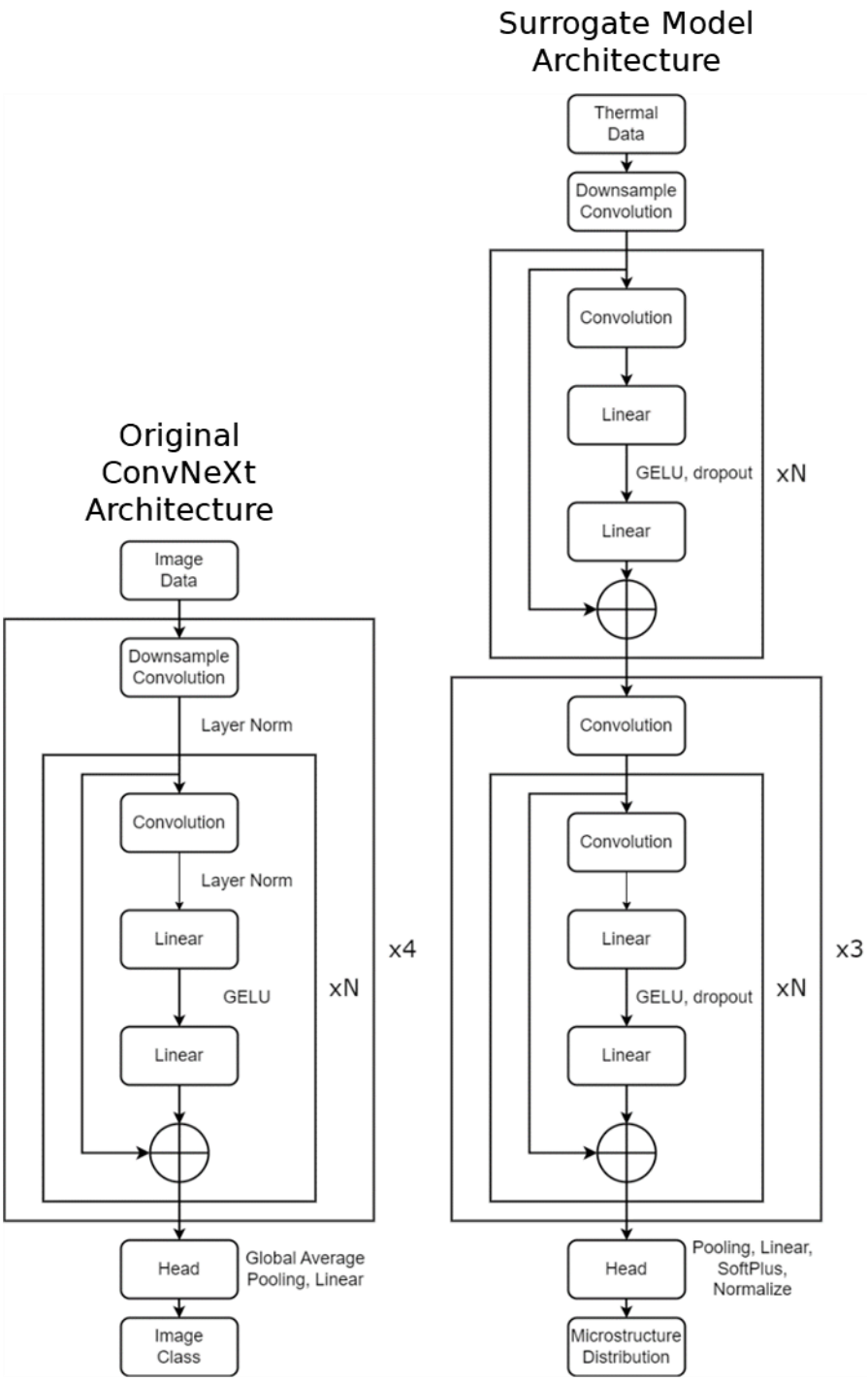

Figure 25 –Network diagrams showing original ConvNeXt architecture (left), and the surrogate model architecture (right). Where N indicates how many blocks occur in each stage of the model.

The most apparent difference between the ConvNeXt architecture and the neural network used in our surrogate model is that the ConvNeXt is designed for use with 2D images with 3 color-channels, whereas our surrogate model must work with 3D arrays with 9 thermal data channels. Most of the

changes required for this have limited impact on the architecture, requiring only the change of a few numbers and changing 2D convolutions to 3D convolutions. The one significant difference created by this is that the dimensions of the input array need to be much smaller than those of images, due to our 3D inputs scaling in size cubically instead of quadratically. This decrease in the dimensions of the input array in turn necessitated the removal of 3 of the 4 down-sampling layers, as the patch based down sampling would reduce the data to a single dimension after just 2 down-samples. This would have effectively removed any spatial information available to the last 3 blocks of the neural network. As the LPBF process is spatially complex, this was undesirable, so the remaining down-sample convolutions were modified to only alter the number of channels, keeping the spatial dimensions created by the first down-sample throughout the rest of the network.

Because we are trying to make predictions about a single voxel at a time, the dimensions of the input cube were made odd, so that the voxel of interest could be centered, to avoid directional bias. Additionally, because all 3 of the state-of-the-art architectures mentioned rely on “patchifying” – down-sampling the input using non-overlapping equisized patches and encoding them across the channels – to achieve their performance, it was necessary for the dimensions of the input to not be prime so that this could be maintained in our model. In order to maximize the speed of the model, and aware that the moving window approach results in overlaps where the data must be loaded and used for computations multiple times, including duplicate calculations, it was desirable to keep this window as small as possible while achieving reasonable model accuracy. Initial versions of the model used a cube 15 voxel (75µm) across successfully; however, the accuracy of the model began to drop as larger simulations with larger maximum grain sizes were incorporated into the data, indicating that more context was needed for accurate predictions. As a result, the input window was increased to 25 voxels (125µm) across, which resulted in higher accuracy for these cases and did not meaningfully slow the model.

While accommodating the dimensions of the data is the most obviously necessary change, one of the most significant changes is the removal of the layer normalizations. This was done because initial iterations of the model struggled to learn consistently until the layer normalizations were disabled. It is not known exactly why this improved the model’s performance, as normalization layers are generally considered to improve the training rate and generalization of neural networks [142,143]. One possible explanation is that the relationships between the microstructure distributions and thermal

characteristics are strongly dependent on the varying magnitudes of the individual thermal characteristic channels in a way that is impeded by contextually scaling them. However, no literature could be found which indicated that this was generally true for regression-based neural networks.

The head – the last block of layers, responsible for distilling the outputs of the model – was modified to use channel-by-channel pooling (averaging) instead of global average pooling, and a softplus function was added after the linear layer to ensure positive valued outputs, which were then normalized to sum to one, for a proper probability distribution. While other functions, such as softmax, accomplish the same thing in fewer steps, this was the first combination which was found to work and no reason to revisit it arose.

Another notable change made was the addition of dropout regularization – which randomly zeros out a portion of the values during each step of training – to the linear layers of the model. This is equivalent to removing the corresponding nodes, resulting in what is effectively a slightly different neural network being trained with each training step. While dropout has generally been shown to improve accuracy and prevent overfitting to training data [144], it was further hoped that the quasi-ensemble of neural networks which dropout creates during training would be more capable of learning from the stochastic microstructure data. Conversely, layer dropout, which randomly removes layers from the model and is included in the original ConvNeXt architecture, was not used because the original ConvNeXt code used for reference has it disabled by default and the model performed well without it.

3.3 Training Dataset

To train the surrogate model, a dataset was generated by simulating the LPBF process for 28 cases, with each case having a unique combination of process parameters and part dimensions. The process parameters which were varied most often between cases were the laser power and velocity, the combinations of which are plotted in Figure 26. When selecting the process parameters to use, the decision was made to focus primarily on the powers and velocities in the vicinity of those used for calibrating the thermal model on which ours was based, resulting in most of the data only covering a very small portion of the process maps proposed in other LPBF research. This is because, while the ultimate goal is to have a broadly applicable surrogate model, the computational cost of generating each dataset is high and the work presented here is still exploratory in nature.

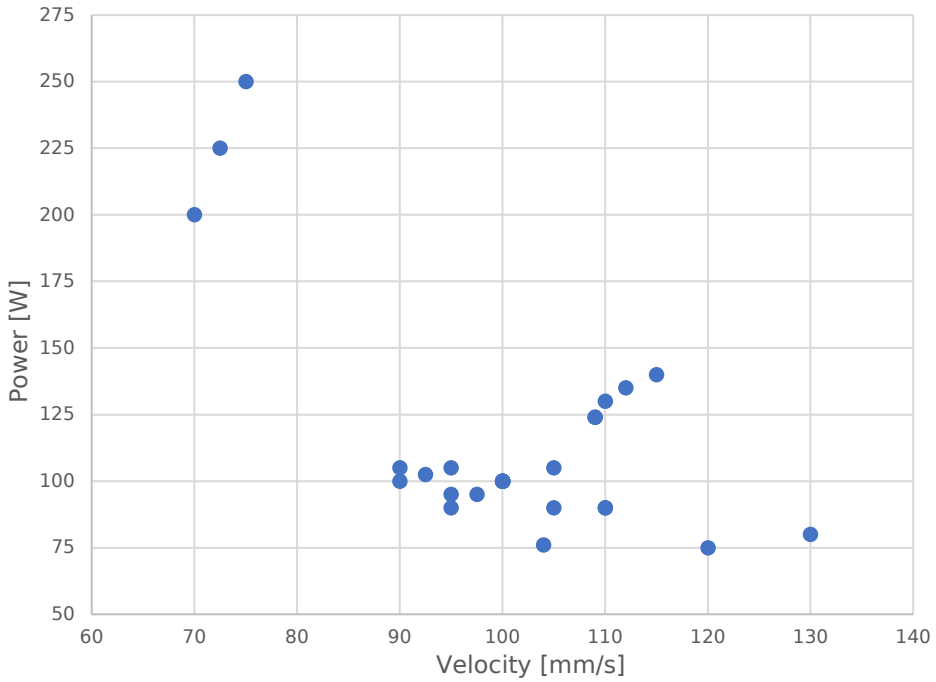

Figure 26 – Plot of the powers and velocities used for training data generation, other parameters such as the hatch spacing and layer thickness were varied independently.

In addition to the power and velocity, several geometry-based parameters were varied during the data creation process. These include the dimensions of the part being printed, as well as the layer thickness and hatch spacing, along with the dimensions of the simulation domain. The layer thickness and hatch spacing are obvious choices to vary, as they have a direct impact on the thermal history of the parts even when the part geometry remains the same. The part dimensions, which were on the order of 1mm, were varied in all three dimensions to reflect some of the diversity in part geometry that the model might encounter in practical applications, though all simulated LPBF prints were either rectangular or square profiles. While 1mm is quite small in comparison to the scale of typical printed parts, the scale is on the same order of magnitude as the simulations used when validating the SPPARKS model, and is large enough to capture all but the tallest grains from the original paper [93]. Additionally, while parts are generally much larger than this, some scan patterns such as islanding [49], and some geometric structures such as lattices [145], contain features that are similar in scale. The rectangular profiles of the parts were aligned with the simulation domain axes, and printed using a raster scanning pattern like those shown in Figure 4 of Section 1.2, with the laser turned off for turns. While the overall geometry was kept rectilinear, the direction of the laser scan path was rotated by 67° between each layer of most prints, though two cases using a 45° and a 90° rotation were also included, resulting in a

variety of thermal gradient, solidification, and grain growth directions. Contour passes – passes of the laser along the exterior edges of the part, which are commonly used to control the surface finish – were not included for any cases.

While it is standard practice in the field of modeling and simulation to vary the size of simulation domains in accordance with the size of the process being modeled, we also varied the width of the buffer zone between the edges of the scan path for the part and the boundaries of the simulation domain. This variation in the distances between the edge and bottom surfaces of the part to the boundaries of the simulation domain results in slight differences in the solidification processes increasing the diversity of the training data. While these changes to the buffer zone are not representative of specific changes to real world LPBF process parameters, as these boundary conditions are non-physical and real build volumes are typically much larger, every change in the process parameters will result in same variation in the distance between the edges of the melt pool and the simulation boundaries. By intentionally varying this, we expose the surrogate model to a broader range of the possible relationships that might occur during the LPBF process optimization process. Additionally, even though the non-physical nature of these changes could break the model's calibration, resulting in simulated thermal processes that are not accurate to the real process, the relationships between the resulting microstructure and thermal characteristics would still be valid. While the exact conditions such data captures may be unlikely to occur during real-world printing, learning from data that falls outside expected ranges may improve the model's accuracy within expected ranges.

Though there are many other process parameters which can be controlled, including the laser beam width, laser beam shape, and ambient temperature, the decision was made not to vary these parameters. The ambient temperature, which controls the initial temperature for all layers and the constant temperature boundary condition at the bottom of the simulation, was held constant at 300K for all cases in the dataset. Additionally, all cases used a laser beam diameter of 50µm with a rotationally symmetric Gaussian power distribution, where the beam width is defined as four times the deviation of the Gaussian. These parameters were chosen to be consistent with the calibrated thermal model on which ours is based, and are representative of a reasonable use case. While varying these process parameters would have created more variety in the data set, and been more representative of real-world process design, the laser power and velocity are generally considered the most influential process

parameters, and the beam width and shape are not controllable on all LPBF machines. Additionally, further increasing the dimensionality of the process parameter space would significantly increase the number of cases needed for equivalent coverage. For these reasons, and given that this work is exploratory in nature, it was deemed reasonable to exclude these and other process parameters from consideration in this study.

The cases in the initial dataset used powers and velocities selected via grid sampling, centered around a reference case of a 100-Watt laser moving at 1 meter per second, and all used a single-layer rectangular print with identical dimensions and simulation domains. These initial simulations were used when developing the model architecture, on the basis that simulating single layer cases is much less computationally expensive than simulating many-layer parts, allowing for a more diverse dataset to be created for initial feasibility tests. While this provided a diverse group of test cases in terms of process parameters, larger multi-layer prints provide more diversified training data due to the complexity of the process and the increased number of voxels on which to train. For this reason, and the fact that many-layer prints are much more representative of real-world use cases, multi-layer prints are better choices for creating training data.

To better diversify subsequent training data, we used a Gaussian process based Bayesian optimization method for experimental design to select process parameters and dimensions based on the microstructures of existing data. Because we wished to vary both the thermal characteristics and microstructure statistics in our training data, the optimization objective was finding the LPBF parameters which maximized the distance of the resulting grain size distribution from those created by the reference 100W-1m/s case. To increase the diversity of the resulting cases, the optimization algorithm was configured to slightly prioritize exploration of the parameter space over optimization. The result was a semi-random assortment of process parameters combinations to add to the training set, which we hoped would provide sufficient variations in both the microstructures and thermal characteristics for the model to learn the underlying relationships. Finally, a variety of parameter combinations were manually selected for use as test cases, for testing the predictive power of the surrogate model in a variety of scenarios. Though these test cases were selected for testing the surrogate model on cases for which it was not trained, presented in Section 4.5, they were included in the training data for the majority of the model analyses presented in this work due to the high computational cost of data generation.

For the purposes of referencing the cases in the dataset throughout this document, we will use the names listed in Table 3. These names are simplified representations of the process parameters, where the number after P is the power in Watts, the number after V is the velocity in centimeters per second, the number after LH is the layer thickness in 5µm voxels, the number after H is the hatch spacing in µm, the number after R is the rotation of the laser scan direction between layers in degrees. For all cases, the names are simplified down to just the process parameters that differentiate them, with the exception of the only 2 cases with different angles of rotation of the scan direction between layers. For cases that are not differentiated by these 5 parameters alone, we have appended a case number to the end. For a complete accounting of all process and geometric parameters used for these cases, refer to Appendix A. In total, the resulting dataset consists of just over 77 million voxels of grain size distributions, and their corresponding thermal characteristics.

Table 3 – List of cases in the full training dataset, with summarized process parameters.

Name	Power (W)	Velocity (mm/s)	Hatch Spacing (µm)	Layer Height (µm)	Rotation (deg)
P90V95	90	95	50	N/A	N/A
P90V105	90	105	50	N/A	N/A
P100V90	100	90	50	N/A	N/A
P100V100Case1	100	100	50	N/A	N/A
P100V100Case2	100	100	50	N/A	N/A
P95V95	95	95	50	N/A	N/A
P105V90	105	90	50	N/A	N/A
P105V95	105	95	50	N/A	N/A
P105V105	105	105	50	N/A	N/A
P124V109	124	109	65	N/A	N/A
P124V109LH20	124	109	65	20	67
P75V120	75	120	25	15	67
P90V110	90	110	50	25	67
P250V75	250	75	110	40	67
P200V70	200	70	100	40	67
P100V100LH15Case1	100	100	50	15	67
P100V100LH15Case2	100	100	50	15	67
P95V97.5	95	97.5	50	N/A	N/A

Name	Power (W)	Velocity (mm/s)	Hatch Spacing (µm)	Layer Height (µm)	Rotation (deg)
P102.5V92.5	102.5	92.5	50	N/A	N/A
P130V110	130	110	50	20	67
P135V112	135	112	50	20	67
P140V115	140	115	50	20	67
P100V100H45	100	100	45	N/A	N/A
P100V100H40	100	100	40	N/A	N/A
P225V72.5	225	72.5	105	35	67
P90V110R45	90	110	50	25	45
P76V104R90	76	104	50	25	90
P80V130	80	130	25	15	67

When generating the microstructure data for these LPBF parameter sets, the number of microstructure simulations in each ensemble ranged from 100 to 270 simulations, with the majority around 200. The exact number of simulations in each ensemble was varied so that the probabilities would not all be quantized multiples of 0.5 or 1%, exposing the model to a better representation of "real" probability distributions. This was done through both intentional variation of the ensemble size and in a more random fashion by not re-running simulations if a small number failed due to hardware issues with the cluster on which they were run.

In order to take full advantage of the expensive simulations used to create our training data, the multi-layer prints were treated as multiple distinct datasets by saving the simulations after each layer had been completed. Because the laser can re-melt lower layers or heat them enough for solid-state grain growth, and because some grains continue to grow through the new layers, the thermal characteristics and microstructure statistics for the lower layers continue to change as more layers are added. This means, for example, that a 30-layer print has 30 unique datasets associated with it, with each containing more, and more varied, data than the last. However, when this was first introduced, we found that the model struggled to learn as well as it had with just the final datasets. It was reasoned that this was due to the influence of the laser on the thermal characteristics of lower layers not being sufficiently large for the model to distinguish between intermediate layers given the limited context window of the input. As a result, the decision was made to limit the training datasets to only the first few

layers of the print, along with the full-sized print. The lower layers can be included because these primarily contain intermediate and small grain sizes that do not continue growing as more layers are added, and the context window is sufficiently large to capture the variable number of layers when there are only a few. With the full-size, final-layer data the large columnar grains are of a constant size and there is no need for the model to know the total number of layers when making predictions about the intermediate layers. While this may have artificial impacts on the model, and the relationships we are asking it to learn, the majority of the multi-layer prints were configured to be similar heights of about 1000µm to keep the model from associating specific thermal characteristics with specific heights of columnar grains as determined by the height of the print. It may be possible to minimize these limitations by increasing the size of the context window or by adding more thermal characteristics representing the time spent above lower temperatures, which may be indicative of grain growth occurring in subsequent layers not revealed by the context window.

Because the data used to train the model presented here covers a limited portion of the parameter space and does not cover the variety in part geometries expected in practice, future iterations of the model would need a larger and more diverse dataset for use in practical applications.

3.4 Data Regularization and Augmentation

As discussed in Chapter 1, it is common practice to apply regularization and augmentation techniques to the data used to train machine learning models, to help models better learn the desired relationships without overfitting to the training data. Because our surrogate model is still an exploratory effort, it was not evident which, if any, of the many common regularization and augmentation techniques would be applicable to our problem. Instead, we once again leveraged our knowledge of the process, along with the form of the data, to inform our choice of data augmentation. First, we will discuss the augmentation applied to the thermal data used as inputs during the training process, then those applied to the microstructure statistics used as training targets.

While our 3D thermal data was chosen because the growth of microstructures during solidification processes is highly dependent on the thermal gradient and the solidification behavior of nearby material, no outside directional spatial influences – such as gravity or magnetic fields [146,147] – are included in the SPPARKS simulations we use to create our data. Because of this, it was reasoned that

any possible orientation of the thermal data would be representative of a valid solidification process and would not be missing any information not captured by the thermal channels. Though LPBF printing upside-down seems highly unlikely to occur in practice, it was decided to incorporate these orientations into the model training process in the hope that it helps the model better learn the true influence of the spatial relationships in the thermal characteristics on the microstructure statistics. In theory, this should also eliminate any biases created by the consistent orientations of the scan paths across the many LPBF simulation datasets. As our moving context window is a cube, there are 48 possible orientations [148], created by rotating and flipping the cube, which we can sample from randomly and use to transform the data in our context window as we load it for training our model.

While these transformations are generally straightforward for most of the thermal characteristics because they are scalars, the thermal gradients require special treatment due to being vector components. This is because, in order for the gradient components to have a consistent meaning, the positive and negative values of each component must refer to consistent directions, something which is not maintained with a naïve rotation. To illustrate this problem, consider a 90-degree rotation around the z-axis as depicted in Figure 27, after which the initially positive x-component should now be pointing in the global negative y-direction, and the positive y-component in the global positive x-direction, as illustrated by the blue arrow. Because the rotation only affects the location of these gradient vectors, a naïve rotation will give the correct location of the thermal gradient, but the incorrect global direction as illustrated by the red arrow.

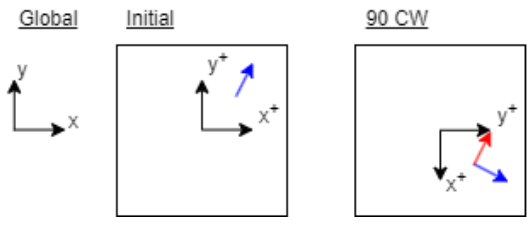

Figure 27 – Illustration depicting the rotation of the thermal gradients around the z-axis. The blue vector illustrates how the correct rotation changes the sign and magnitude of the components of the gradient with respect to the global axes. The red arrow depicts the incorrect global direction of the thermal gradient achieved with a naive rotation.

To account for this, the data-loader modifies and swaps the gradient components with each rotation and flip of the thermal data in accordance with these rules:

- If Flipping:
  - 1st. Flip the tensor.
  - 2nd. Swap the sign of the gradient component that is normal to the plane of the flip.
- If Rotating:
  - 1st. Rotate the tensor.
  - 2nd. Swap the directional thermal gradient components according to the following rules, as depicted in Figure 28.
    - i. The gradient component along the axis of rotation does not change.
    - ii. The remaining two gradient components swap.
    - iii. If the positive direction of a gradient component has been rotated to align with the negative direction of one of the global (original) axes, then use the negative of that component during the swap.

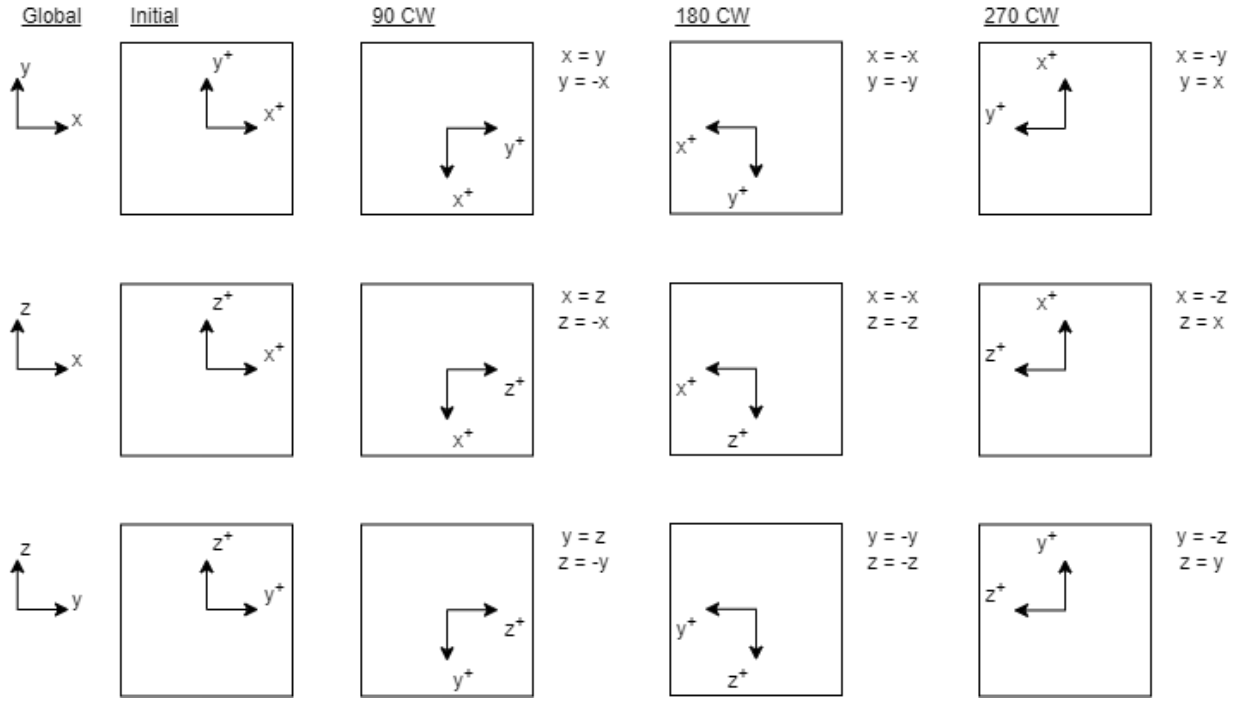

Figure 28 – Illustration of the rules applied to the thermal gradients during when rotated. The global axes depict the plane on which the rotation occurs and the directions of the global axes. The axes within the boxes depict the initial positive directions of these axes, and the orientation of these axes after their corresponding rotations, but before corrections. The equalities depict the swaps which need to be made after the tensor has been rotated, where the left side of the equality is the global axis, and the right is the data which needs to be moved to that axis.

Additionally, each thermal characteristic was divided by a constant normalizing factor to make the many different measures have similar maximum magnitudes on the order of 1. In contrast to the normalizations frequently applied during machine learning, this scaling does not attempt to scale the data to a specific distribution or shift the mean to zero, and is not done using learned parameters.

Instead, these normalizing factors were selected by manually inspecting the data rather than leaving them as a variable for the model to learn. This is done because the 9 thermal characteristics vary in magnitude significantly, which if left uncorrected, could cause instabilities when training the model due to the disparities in magnitudes both between the channels and with the model weights and target probabilities.

While it is common to apply other regularization techniques such as random noise to the inputs of image classification models [149,150], the decision was made not to use any here because unlike image data, our thermal data is not inherently noisy and is representative of the exact thermal history used by the microstructure simulations. As a result, any deviation from the computed thermal characteristics could negatively impact the ability of our model to learn the relationships by representing a slightly different solidification process.

In contrast to the exact thermal data, and unlike in image classification problems where there is a single correct answer, the target microstructure distributions are stochastic samples of the true distributions, and thus are not the exact correct distribution. We can leverage this inaccuracy in our data, to help avoid potential pitfalls that can occur while creating data-driven models, by resampling the target microstructures during the model training process, in effect adding noise. While this process, called bootstrapping, is generally useful for analyzing statistics sampled from a population, here the random variation should help prevent overfitting of our model, which would be problematic from both a generalization and accuracy perspective. In theory, this helps prevent overfitting by ensuring that the same samples are not observed multiple times during training, and that any correlations between neighboring voxels which are caused by outliers are minimized. In general, exactly reproducing non-deterministic datapoints used to train a machine learning model means that the model is overfit, and may have unpredictable behavior for new data, making this an undesirable characteristic. Additionally, because the microstructure statistics used for target distributions are themselves samples of the true underlying distribution, learning to exactly reproduce these distributions would inherently limit the accuracy of the model to that of the ensembles used in the training data, while appearing to have perfect accuracy.

While the sampling process is designed to create deviations from the data produced by the microstructure ensembles, those ensembles are the closest representation of the ground truth we have,

so the sampling process was made to use a large number of samples to avoid introducing new outliers. Additionally, this data resampling was only given a 50% chance of happening on any given distribution fed to the model for training. An additional small amount of noise was also added to each microstructure distribution so that the bins with zero probabilities would be a very small non-zero number, but the non-zero probabilities would not be affected. Because all the microstructures in the training data have distributions containing zero-probability bins, this was done in the hope that it would prevent these zero-valued bins from negatively influencing the model. However, the magnitude of this noise was very small, and no studies were performed to determine if this had any influence on the training outcome.

3.5 Model Training

Overall, we found that the surrogate model was very easy to train to our desired levels of accuracy, and very few changes to the training techniques presented in the ConvNeXt paper [114] were needed. For example, the model weights were randomly initialized using a truncated normal distribution, with bounds of [-2,2] and a standard deviation of 0.02, and the biases were initialized to zero, which is identical to the initialization used for ConvNeXt. As a result of this ease, and because the primary focus of this work is not on fine-tuning a new neural network architecture, very little experimentation of the training hyperparameters was performed, except for with the test-train split presented later in this section.

To train the model, we used the AdamW optimization algorithm [151,152] with a variable learning rate, adjusted every batch using the cosine annealing with warm restarts scheduler in PyTorch [153]. This learning rate scheduler decreases the learning rate in a sinusoidal fashion, before resetting to the initial learning rate after a controllable number of epochs. However, we found that the model consistently converged before the learning rate was reset, so we set the learning rate scheduler to reset after 4 epochs and trained the model for 4 epochs. The maximum learning rate used was 0.001, and the minimum 0.5E-5, along with the AdamW weight decay of 0.01. The model was trained on batches of both 128 and 256 samples at a time, with no obvious difference found in the accuracy of the resulting models. As such, the majority of the results presented in this work use models trained on 256 sample batches.

To measure the model's accuracy, and inform the training process, the mean squared error (MSE), Equation (6), was used as a loss function to calculate how far the model's predictions are from the target microstructure statistics in the training data.

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2 \qquad (6)$$

Y_i is the distribution being tested, $\hat{Y}_i$ is the ground truth distribution against which we are testing, and n is the number of bins used to discretize our distributions.

The MSE is a common metric for measuring the accuracy of multi-value regression machine learning models, primarily because it has nice mathematical properties. While it is mathematically elegant, the mean squared error does not capture discrepancies in the population statistics, such as the mean diameter, in a way that might be meaningful to users of the model. To try to better capture these discrepancies, alternatives to the MSE such as the Wasserstein metric (also called the earth movers distance (EMD)) were considered, but initial versions of the model failed to learn using these metrics. In light of this, the statistical experiments in Section 3.6 were carried out, to help us better understand the meaning of the MSE with respect to our data. The results of this analysis provide us with target ranges for our MSE which indicate that the model is performing well. With this information, it was not necessary to revisit our choice of loss function for the purposes of ensuring the model's accuracy. While this information makes the MSE useful for understanding the model's accuracy, it does not address the MSEs lack of representation of discrepancies in the population statistics. Despite this, we have chosen to use the MSE throughout this document for the purposes of consistency, though other metrics such as the EMD can easily be used for analysis of or optimization with already-trained surrogate models.

One of the few training hyperparameters which was varied and tested was the test-train data split, or the percentage of the total dataset which was used for training. We experimented with this hyperparameter not to improve our model, but instead to better understand the implications of our moving window approach for the testing accuracy of our model. While using higher proportions of the dataset for training the model is generally preferred, as the more examples the model has to train on the better it can learn, the fewer datapoints left for testing, the higher the likelihood of the testing set not being a representative sample. This risk is somewhat diminished for very large datasets, but the moving window approach we use, which results in overlapping data, combined with the spatial correlations of

microstructures, mean that there is a strong correlation between datapoints in our dataset. As a result, it was not known if our testing dataset was too correlated with our training dataset to give an accurate representation of the model's accuracy. To test this, we started at our standard 9:1 train test split, and trained models on decreasing proportions of the full dataset. The results of this test can be seen in Figure 29, which shows that there are no substantial decreases in the model's accuracy until less than 65% of the data was used for training. This represents a substantial decrease in the amount of data for the model to train on, which can significantly limit the accuracy of any neural network. As a result, and due to the results presented throughout the rest of this work, we do not believe that our 9:1 test train split is resulting in inaccurate accuracy measurements or overfitting of the model to the data.

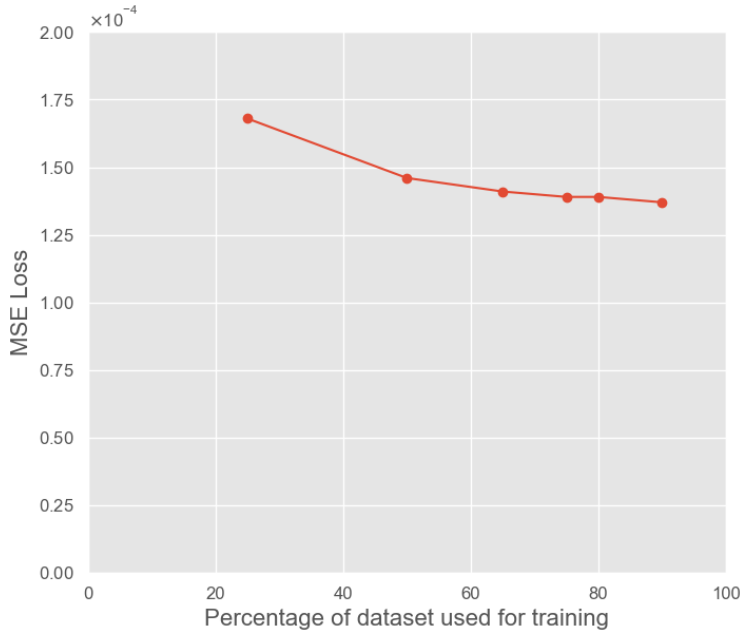

Figure 29 – Plot of the model's MSE against test data vs the percentage of the full dataset used for training the model.

Due to the unusual structure of our datasets, it was necessary to implement a unique data handling infrastructure for training the model. This is because, unlike with many machine learning models, the individual samples in the training set come not from many small files which can be loaded on the fly, but instead from sampling a small number of very large files tens of millions of times. To prevent unnecessary overhead from loading these files multiple times with every batch, the files were all pre-loaded to memory. However, PyTorch's default implementation of parallel data-loading using Python multiprocessing resulted in the dataset expanding to several terabytes in memory, as it needed to be duplicated for each worker process. This was ultimately solved by using the experimental PyTorch

TensorDict library, which does not load the files to memory, but instead uses memory mapped files stored on mass storage. These memory mapped files can then be accessed with minimal latency, without loading the full file, and can be shared between PyTorch's worker processes. While this method has increased latency over loading the files to memory, it allowed us to train the model with a relatively small memory footprint, faster than would have been possible loading data from the original files on the fly.

3.6 Understanding The Mean Squared Error

The mean squared error we are using is well understood in a mathematical sense, and a common choice of loss function for training machine learning models. However, it is not intuitive how the MSE relates to the accuracy of the grain size distributions our model predicts, for a variety of reasons. Here we discuss these reasons, then discuss how we used Monte Carlo sampling to better understand how to interpret the accuracy of our model as measured by the MSE.

The first reason for this opaqueness of the MSE is that, while a lower MSE always corresponds to a predicted distribution that is closer to the target distribution, our target distributions are themselves stochastically sampled from the underlying ground truth in the form of SPPARKS ensembles. Because of this, we know that our target distributions are themselves noisy, so exactly reproducing them with an MSE of zero would indicate that we have encoded the noise specific to our samples into any relationships the model has learned. This would be overfitting, and would likely mean that our model has not actually learned the physical relationships we hoped it would. While it is unlikely that our neural network would learn to exactly reproduce our data, partly because of finite training time and the huge number of parameters and datapoints, and partly because of the data augmentation used during training, this stochasticity means that we cannot simply aim for the lowest MSE possible and must determine what MSE corresponds to a reasonable accuracy.

The second reason is that the MSE is representative of the average accuracy of each bin of our predicted distributions but does not provide information about where in the distribution any discrepancies occur. For this reason, the MSE does not provide direct insight into the accuracy of the population statistics such as the mean and standard deviation of the grain size, or the impact that any

outliers might have[1]. Because the mean grain size has known relationships with physical properties [61], the accuracy of such statistics would likely be more important than the MSE to anyone using the model for LPBF process optimization. While there is no way to measure this directly using the MSE, we can be reasonably confident that a sufficiently accurate distribution, as measured by the MSE, will have accurate population statistics, but must determine what the corresponding MSE is.

With these limitations on the MSE in mind, our target MSE must be one which indicates both a sufficiently high accuracy for our population statistics to be the same, and a sufficiently low accuracy to indicate that the model is not overfitting to the stochastic data. To determine this target, we take advantage of the stochasticity of our data and perform a Monte Carlo sampling based statistical experiment as described below.

3.6.1 Method

Because our data is produced by ensembles of SPPARKS simulations, making it a representative sampling of the underlying ground truth, it is possible to create new distributions which represent the same ground truth by creating a new ensemble of data. Additionally, because both of these distributions represent the same ground truth, it can be argued that any discrepancies in the population statistics of these distributions are reasonable. Furthermore, because it is not possible to be more accurate than the ground truth, the lowest possible MSE value our model could have is the same as the MSE between the ground truth and our target distribution. However, because it is unreasonable to expect our model to perfectly predict the ground truth by learning from our noisy data, a more realistic target would be to predict a distribution which is representative of the ground truth. In this case, the expected MSE would be the same as that between two distributions sampled from the same ground truth. As such, we can say that an ideal target MSE value for our model is the same as the average MSE between distributions sampled from the same ground truth, and that on average it should never be lower than the average MSE between sampled distributions and their ground truths.

To determine these values, a computational experiment which calculates statistics by repeatedly sampling a ground truth and comparing the samples, as depicted in Figure 30, was carried out. As

[1] While a large discrepancy in any single bin would be reflected in the MSE, a small discrepancy in the probability of a bin that is far from the expected distribution, in terms of grain diameter, would appear identically to the same discrepancy occurring in a bin much closer to the bulk of the expected distribution.

described above, two distributions are sampled from the ground truth, then the MSE between them and each other, and them and the ground truth, is calculated and recorded. Because of the high computational cost of our SPPARKS ensembles, we do not actually run a new ensemble to get each distribution, but instead create a very large ensemble, representative of the ground truth, to sample from. Furthermore, to ensure the maximum variety in the distributions being sampled, we perform this experiment on each voxel of the part simulated in the ensemble. However, because of the high computational cost of the SPPARKS simulations, generating our ensemble with a large simulation and generating multiple large ensembles were not practical, so all distributions come from a single small print with a single set of LPBF process parameters.

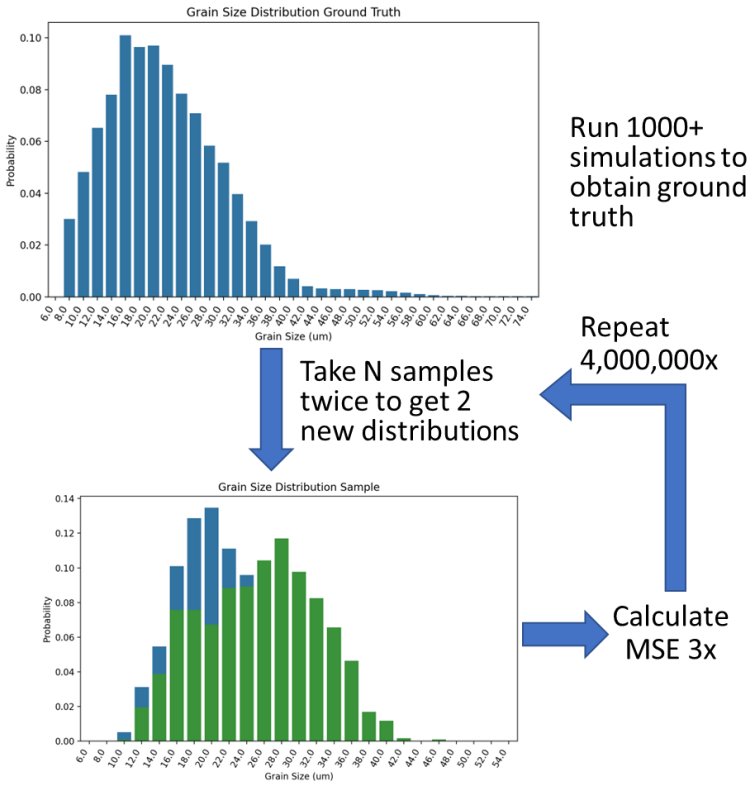

Figure 30 – Flowchart depicting the experimental procedure. We start with a distribution obtained from an ensemble, which we will call our ground truth. We then sample two distributions from the ground truth and calculate the MSE between all 3 pairings. We then repeat this process for the distribution corresponding to all voxels in the case.

3.6.2 Results

A plot summarizing the results of this numerical experiment can be seen in Figure 31, which shows a box and whisker plot of the mean squared error between the sampled distribution and the ground truth as the number of samples changes. The data shows that the mean squared error decreases

asymptotically towards zero as the number of samples increases, where an MSE of zero would indicate a perfect reproduction of the ground truth. This tells us the maximum reasonable MSE value for our model, but must be considered on a result-by-result basis as the size of the ensembles used to produce our target distributions is variable. While the MSE values shown here get quite close to zero at their minimum, such values are statistically unlikely, and the mean MSE of all predictions in a simulation should be compared to the boxes – which represent 50% of all measured MSE values – instead. In addition to indicating overfitting, these results represent the maximum measurable accuracy for the model against an ensemble of a given size.

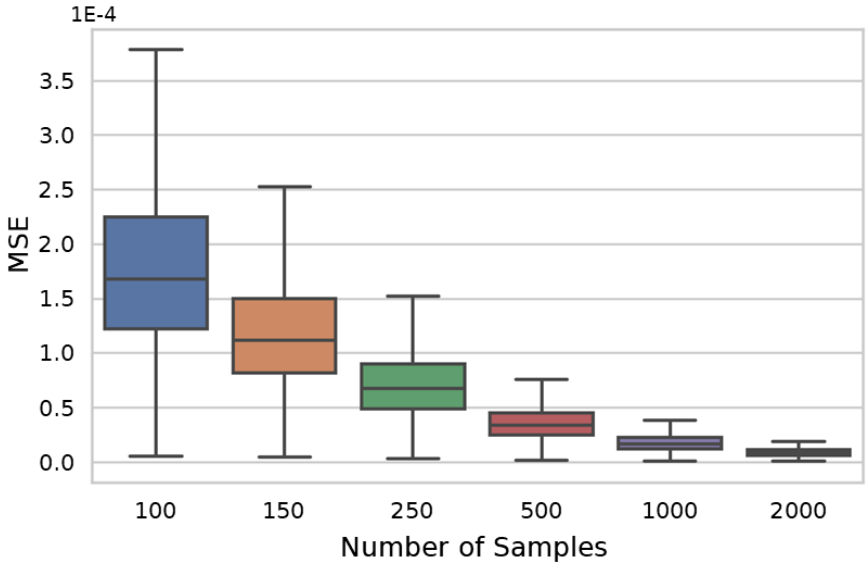

Figure 31 – Whisker plot showing the mean squared error between the sampled distributions and the full distribution. Shows the decrease in the mean squared error as the sample size increases.

In contrast to Figure 31, which represents the maximum theoretical accuracy, Figure 32 depicts a more reasonable expectation of what ideal accuracy values for our model would be. Here we are showing the MSE between the two distributions which were sampled from each ground truth as a function of the sample size. Once again, these values converge asymptotically towards zero, and the lowest values are quite close to zero compared to the MSE values when fewer samples are used. Knowing that the majority of our ensemble used for training have somewhere between 100 and slightly more than 200 simulations, we can say that our ideal accuracy target averaged across all data is around 1E-4.

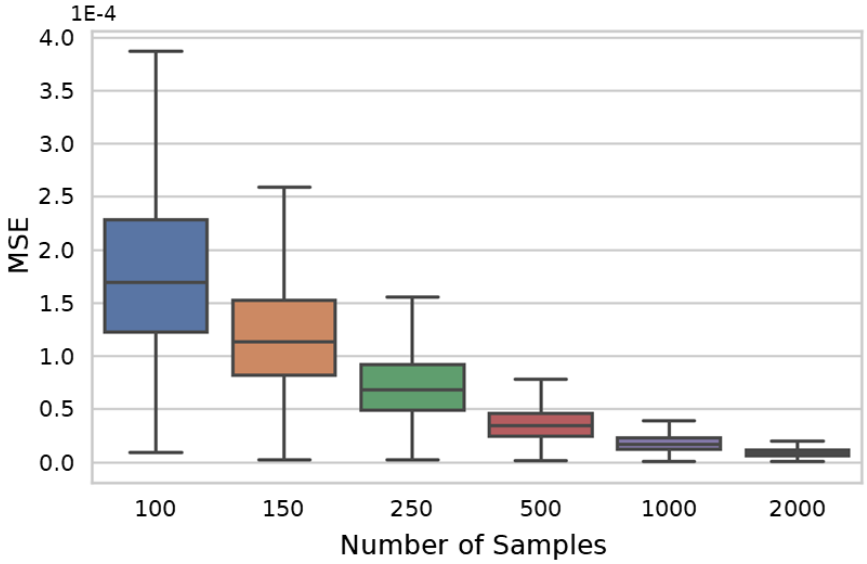

Figure 32 – Whisker plot showing the mean squared error between two sampled distributions. Shows the decrease in the mean squared error as the sample size increases.

Interestingly, the scale of the MSE for this data is not significantly different from the MSE of the previous data. This indicates that our expected MSE of a well-trained model vs our training and testing data is not significantly different from the MSE between the model and the unknown true distribution the training data represents. This does however make it harder to distinguish between an accurate model and an overfit model, indicating the need to avoid the lower tails of these ranges.

Because this experiment relied on a single ensemble of simulations of a single-layer print, the sampled distributions are likely much more similar to each other than distributions sampled from multiple ensembles produced by simulating prints with a variety of process parameters. The consequence of this is that it is hard to say for certain if these results hold true for all possible grain size distributions. However, the results obtained while training and using the model for larger simulations, which will be discussed throughout the remainder of this work, indicate that these results are at least consistent enough to serve as a reasonable reference point.

3.7 Model Optimization

After implementing the model as defined in Section 3.2, the accuracy of the model on the test data was on the order of 1E-4, well within the ideal MSE range given by the statistical experiments in Section 3.6 and at the limits of the maximum measurable accuracy given the size of our microstructure ensembles. While the model met our desired accuracy targets, one of the primary objectives for our

surrogate model is to make microstructure predictions as quickly as possible for the purposes of facilitating LPBF process optimization. Investigating the performance of our model with respect to speed, we found that the surrogate model was indeed much faster than the equivalent SPPARKS simulations. Not including the thermal simulation, the surrogate model can make predictions about the microstructure statistics of a print in minutes using a single NVIDIA A100 GPU. By contrast, a single equivalent SPPARKS simulation can take hours to days running on a high-performance computing node[2], and reproducing the statistics produced by our surrogate model requires an ensemble that takes days to weeks even with multiple nodes.

While these initial results were promising in terms of both speed and accuracy, the number of parameters in the neural network architecture were largely unchanged from the foundational ConvNeXt architecture beyond the changes caused by the dimensions of the thermal data used as the model input. Because each parameter corresponds to one or more operations, reducing the number of parameters in the model has the potential to further improve the observed speed-up. Additionally, while the model achieved the maximum measurable accuracy with the original number of parameters, it is possible that fewer parameters or a different configuration of parameters could result in a model which performs better across a wider range of LPBF parameters.

To reduce the number of parameters in the model, we used multi-objective optimization to find the model hyperparameters which achieved the highest throughput and best accuracy. While we did not expect this to improve the model's accuracy, the accuracy was included as an optimization target to avoid over-pruning the model to the point that it could not make accurate predictions. To carry out this optimization, we used Bayesian optimization, as implemented in Optuna [154] on top of BoTorch [155], and targeted 16 model hyperparameters to optimize, as shown in Table 4. A complete breakdown of the parameters and the ranges used for optimization can be found in Appendix B.

[2] The SPPARKS simulations used for training data generation were run on nodes with 2 AMD EPYC 7532 processors for a total of 64 cores with multi-threading disabled.

Table 4 – Model hyperparameters before and after optimization.

Parameter	Initial value	Result
Layer 1 depth	3	1
Layer 2 depth	3	1
Layer 3 depth	9	3
Layer 4 depth	3	0
Layer 1 Kernel Size	7	3
Layer 2 Kernel Size	7	3
Layer 3 Kernel Size	7	4
Layer 4 Kernel Size	7	N/A
Layer 1 Number of Channels	96	64
Layer 2 Number of Channels	192	96
Layer 3 Number of Channels	384	304
Layer 4 Number of Channels	768	N/A
Layer 1 Scale	4	7
Layer 2 Scale	4	8
Layer 3 Scale	4	1
Layer 4 Scale	4	N/A

This process was able to successfully reduce the number of parameters in the model from 38.9 million to 5.7 million, a reduction of over 85%, increasing the model throughput by approximately 2.5 times while maintaining the accuracy of the model within our ideal accuracy range. A plot showing the accuracy and throughput of all the optimization trials can be seen in Figure 33**,** with the red points indicating the Pareto front, the curve which defines the best combinations of accuracy and throughput. With the exception of a few trials, all resulting models had very similar accuracies, and fell within our target accuracy range, so the model throughput was the primary deciding factor when selecting the optimal configuration. As such, the trial with the hyperparameters reported in Table 4 was selected, as it is the trial with the highest throughput and did not have unreasonably high or low accuracies. While this is the best performing trial observed during the process, the small number of trials in the immediate vicinity and the relative lateness of these optimal trials in the optimization process indicate that there may be other model configurations which perform as well or better; but the computational cost of the process limits the number of trials which can be performed in a reasonable amount of time.

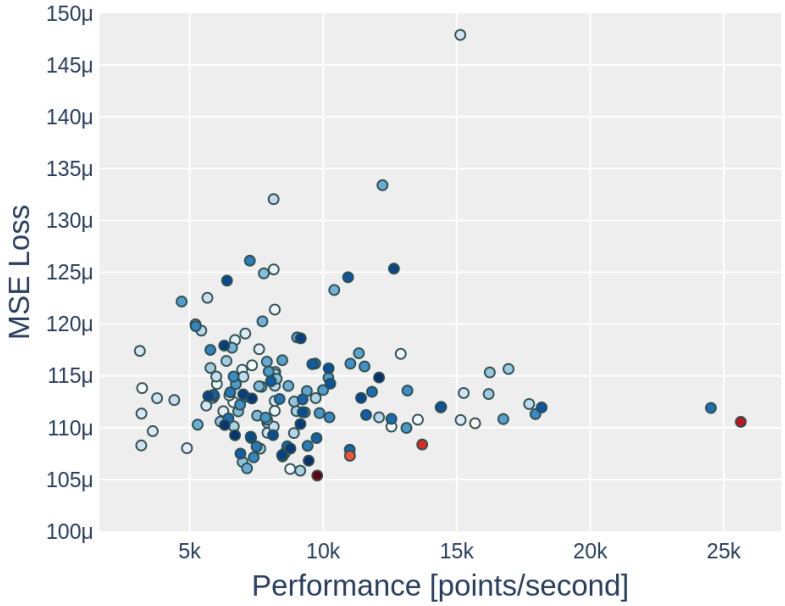

Figure 33 – Plot showing results of all the trials of the network optimization process, where the objective is to minimize the loss and maximize the performance. The best trials are shown in red, and the shade of the points corresponds to the trial number.

Between this optimization and the prior modifications needed to make the model function, the resulting neural network is significantly different than the ConvNeXt architecture on which it is based, despite many of the main principles remaining unchanged. The most significant changes resulting from the optimization process are caused by the removal of the last stage from the model. Not only does this change eliminate several convolution and linear layers, but it effectively removes the 1:1:3:1 compute ratio of the stages by removing the last group to make the ratio 1:1:3. In addition to altering this compute ratio, the optimization process resulted in a significant decrease in the total number of blocks, as each stage has 3 times more in the original ConvNeXt. Also of note, the substantial ratio introduced to the width of the linear layers was surprising due to the lack of any ratio in the original architecture. In contrast, the now smaller convolution kernels make intuitive sense due to the smaller dimensions of the data.

While this process resulted in significant improvements in speed, though no accuracy gains were possible, it was undertaken relatively early in the research. As a result, this optimization process was performed using a limited sample of the total training data, as some of it had yet to be generated, and was done using the initial 15 voxel across input window rather than the current 25. As a result, and because the number of iterations was limited by the computational cost, it is likely that a slightly different set of model hyperparameters could perform better than the ones presented here.

In all, the resulting surrogate model, including the thermal process model, has been observed to be approximately 600 times faster than running a comparable ensemble of 100 SPPARKS simulations on a single compute node, and up to 6 times faster than a single microstructure simulation[3]. Because the thermal model is GPU accelerated and SPPARKS is not, some portion of this speedup is hardware acceleration based rather than inherent to the surrogate model. As such, until hardware acceleration is implemented in SPPARKS, it is hard to say definitively that the surrogate model is faster than a single SPPARKS simulation. However, the surrogate microstructure model provides statistics that would not be available from running a single SPPARKS simulation, so the comparison between the surrogate model and the SPPARKS ensemble is more appropriate. Additionally, while the thermal simulations used here are nominally part of the microstructure surrogate model, it is likely that anyone attempting to optimize LPBF process parameters for a part via simulations would also be modeling other process outcomes that require thermal modeling, such as residual stresses. If we take this into account, and exclude the thermal model runtime from our comparison, the surrogate microstructure model alone is approximately 146 times faster than a single SPPARKS simulation.

3.8 Conclusions

Here we have introduced the model architecture, the data used to train it, the regularization and augmentation methods used during training, and the methods used to train the model. We then analyzed the meaning of the mean squared error with respect to our data so that we could have a reference point when discussing the accuracy of the model. Finally, we showed that the number of parameters in the surrogate model could be significantly reduced while maintaining our accuracy target, resulting in a surrogate model that is significantly faster than the equivalent kinetic Monte Carlo based SPPARKS simulations.

[3] Subsequent optimizations to the thermal characteristics calculations have resulted in an improvement to the thermal process model of up to 80% less runtime on an RTX 3090, meaning that the true number is likely higher.

Chapter 4 Surrogate Model Accuracy

4.1 Introduction

In Chapter 3 we showed that the microstructure surrogate model created using the proposed framework achieves our desired level of accuracy. More specifically, we showed that the MSE achieved by the surrogate model is equivalent to generating new ensembles of SPPARKS simulations equivalent in size to those used for training. Furthermore, we showed that it does so much faster, or with much fewer computational resources, than creating said ensembles.

While these metrics indicate that the surrogate model has nominally met our desired outcomes for this exploratory research, they do not provide sufficient insight into the model's predictions to be confident in its use for process optimization. To better understand the usefulness of the surrogate model for making new predictions, here we will incorporate a more qualitative approach to analyzing the results by visually inspecting the model's predictions for a variety of situations. In doing so we discover that the surrogate model outperforms our expectations in a way that is not measurable by the MSEs obtained while training and testing our model.

To do this, we will first compare the model's predictions for the cases included in the training data to the training data itself. We then train multiple iterations of the model to understand the influence of the size of the ensembles used for training data on the accuracy of the model. Next, we explore the accuracy of the model for a variety of cases on which it has not been trained, to better understand how it will perform in its intended real-world use case. Finally, we examine the implications of using small ensembles to fill in gaps in the model's training data.

4.2 Initial Model Results

In the previous chapter (Sections 3.6 and 3.7) we showed that the model achieves a level of accuracy, as measured by the MSE, equivalent to sampling new distributions from the same ground truth. However, this single number metric alone does not provide satisfactory insight into the accuracy of the surrogate model. Here we qualitatively compare the microstructure predictions made by early iterations of the model to the SPPARKS simulation-based ensemble data on which they were trained by visualizing the statistics. In doing so, we uncover one of the most interesting behaviors of the surrogate model. Finally, we examine a sampling of the microstructure statistics predicted by the final version of

the surrogate model to provide insight into the measured accuracies. This provides a baseline against which to compare the results presented throughout the rest of this chapter.

The initial results presented here come from an early version of the model architecture, which served as a proof of concept and had not yet been rigorously tuned for this use case. These results are worth discussing because they show an interesting phenomenon, which exists in all versions of the model, particularly well. Because the initial training data consisted primarily of single-layer cases to minimize the computational cost of generating the data needed for a proof of concept, the maximum size of the grains in the initial data was much smaller. As a result, the bins used to digitize the grain sizes covered a much smaller range of grain sizes than in final iterations of the model architecture. This, in turn, amplified the influence of the stochasticity on the microstructure statistics, because even small deviations in the size of a given grain could result in a different binning.

After training this early iteration of the model, it was used to predict the microstructure statistics for the full P100V100case1 case, which was included in the training and testing data. This case is a single layer print, printed with a 100-Watt laser moving at 1 meter per second. When measured against the 144-simulation ensemble-based statistics used for training and testing, the model achieved an average MSE of 1.4 E-4. According to the results of Section 3.6, this value is in line with the MSE that we would expect if we were to measure the accuracy of the ensemble against the underlying ground truth, indicating that this is the lowest MSE we can expect to measure using this ensemble.

The microstructure statistics predicted by the model can be seen in Figure 34, which shows the probability that each voxel in the top surface of the print belongs to a 30-32µm diameter grain as predicted by both the model and 144-simulation ensemble. While the overall patterns present in the model's prediction match that of the ensemble, the ensemble-based training data has more variation in magnitude and pronounced striated features that are not present in the model's predictions. Because these features are due to the nature of randomly sampling the stochastic process, and therefore present in all of the training data, this discrepancy was an unexpected result.

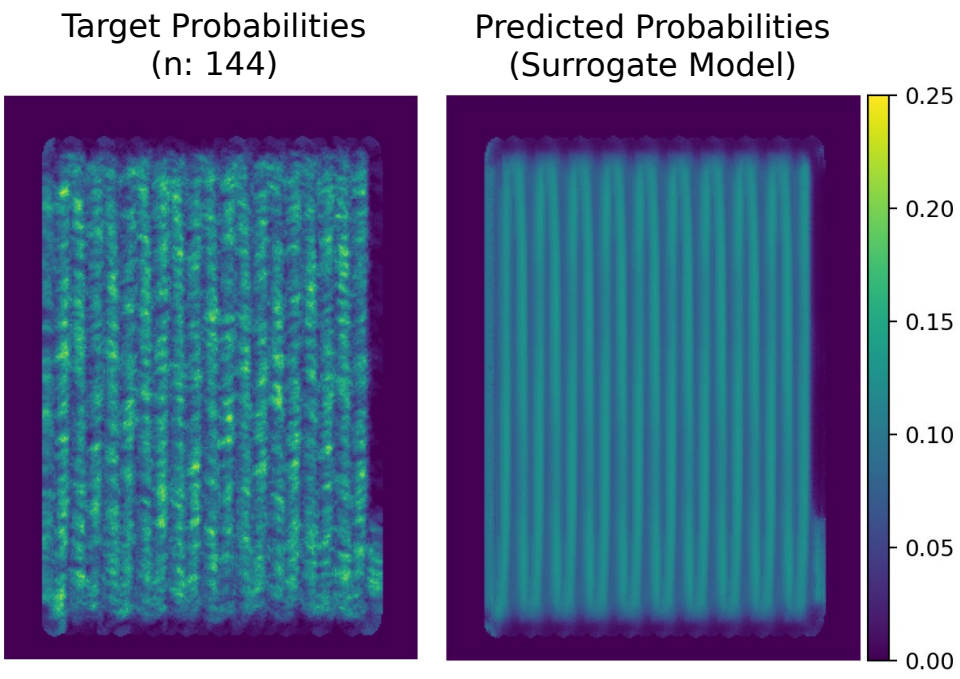

Figure 34 – Probability that each voxel in the top surface of a 1-layer raster scan printed with a 100W laser moving at 1m/s (P100V100Case1) belongs to a 30-32µm diameter grain, with results of the 144 simulation SPPARKS ensemble used for training on the left and the surrogate model's predictions on the right. The irregular striated patterns in the ensemble data are due to the stochastic nature of the SPPARKS model.

These results indicate that the model is learning to average out the random variation present in in the ensemble-based training statistics, and predicting the mean behavior instead. Because increasing the number of samples when measuring a stochastic process decreases variation and better approximates the mean behavior, it was hypothesized that the model's smooth predictions would be closer to the statistics of a larger ensemble than they are to the training data, indicating the model is more representative of the underlying microstructure probabilities than the data on which it was trained. To test this, a much larger ensemble of 1188 SPPARKS simulations was created for this case, decreasing the stochasticity of the resulting statistics and making them a better representation of the ground truth. By measuring the model's predictions against this larger ensemble, we obtain a lower MSE of approximately 2E-5. This lower MSE is consistent with that expected when comparing two ensembles of this size, as determined in Section 3.6, once again indicating that the model is achieving the maximum accuracy we can measure with this ensemble. Additionally, in Figure 35 the model's predictions qualitatively appear to be more accurate compared against this larger ensemble than they did against the smaller ensemble used for training. This confirms that the surrogate model is learning to average out

the behavior of the stochastic noise in the training data when making predictions, in a way that makes the predictions more representative of the underlying statistics than the training data. Due to the high computational costs of continuing to increase ensemble sizes, it is not practical for us to determine the true accuracy of the surrogate model, or how large of an ensemble would be needed to match it[1].

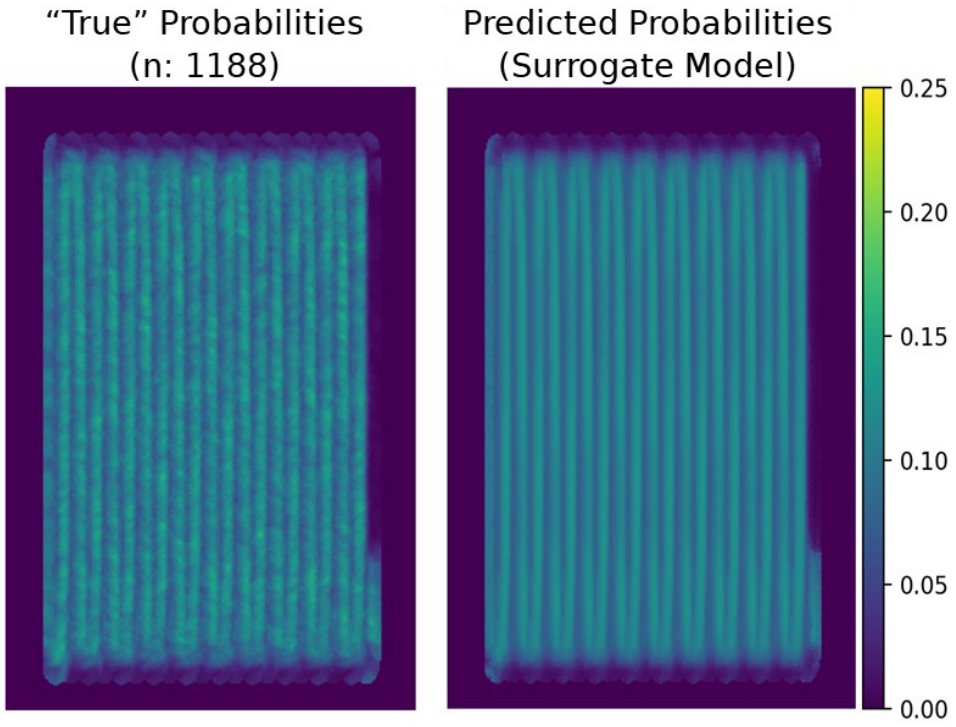

Figure 35 – Probability that each voxel in the top surface of a 1-layer raster scan printed with a 100W laser moving at 1m/s (P100V100Case1) belongs to a 30-32μm diameter grain, with results of the larger 1188 simulation SPPARKS ensemble used for testing on the left and the surrogate model's predictions on the right. Note the significant decrease in appearance of the irregular striated patterns with the larger ensemble, which more closely matches the model's predictions.

While it's not clear exactly how the surrogate model is learning to remove the stochastic noise from our data when making predictions, the ability of neural networks to denoise data is well known. In fact, denoising is the basic principle behind the diffusion models used by popular image generators such as DALL-E, Midjourney, and Stable Diffusion [156]. However, as we do not have a noise-free version of our data, like is used when training such models, the mechanism by which our model learns a less noisy interpretation of the data may be closer to other models such as NOISE2NOISE [157] which is trained

[1] We showed in Figure 19 that there is still a meaningful amount of stochastic noise at 2000 simulations, even with the larger bins used there.

using noisy inputs and noisy targets, and NOISE2VOID [158] and the model by Laine et al [159] which use only noisy input data for training.

4.3 Final Model Results

While this smoothing effect was particularly obvious with the smaller grain size bins and single-bin plotting used for early iterations of the model, these effects continue to be visible with the final model architecture. In Figure 36 we show the prediction from the final version of the model using the 3-color plotting introduced in Section 2.4.2, with a comparison to the ensemble based statistics used for training, with the number of simulations in the ensemble given, for the same P100V100Case1 case shown in Figure 34 and Figure 35 (rotated by 90°). In addition to this direct visual comparison, we also include a plot showing the MSE of each voxel on the surface, where the values are scaled according to the highest MSE shown, and with the average MSE across the entire print given. In this case, the MSE of 7E-5 continues to be in line with what we would expect from a model which is at least as accurate as the 200-simulation training ensemble against which we are measuring. While there are voxels with higher MSEs, they are generally individual voxels or clusters around the extremities of the part, which do not dramatically alter the overall grain size statistics and are not significant enough to be obvious in the visual probability comparison. The lack of visual differences at these lower accuracy clusters indicates that the highest probabilities in these regions belong to the correct color's super-bin, which in this case span just 16µm.

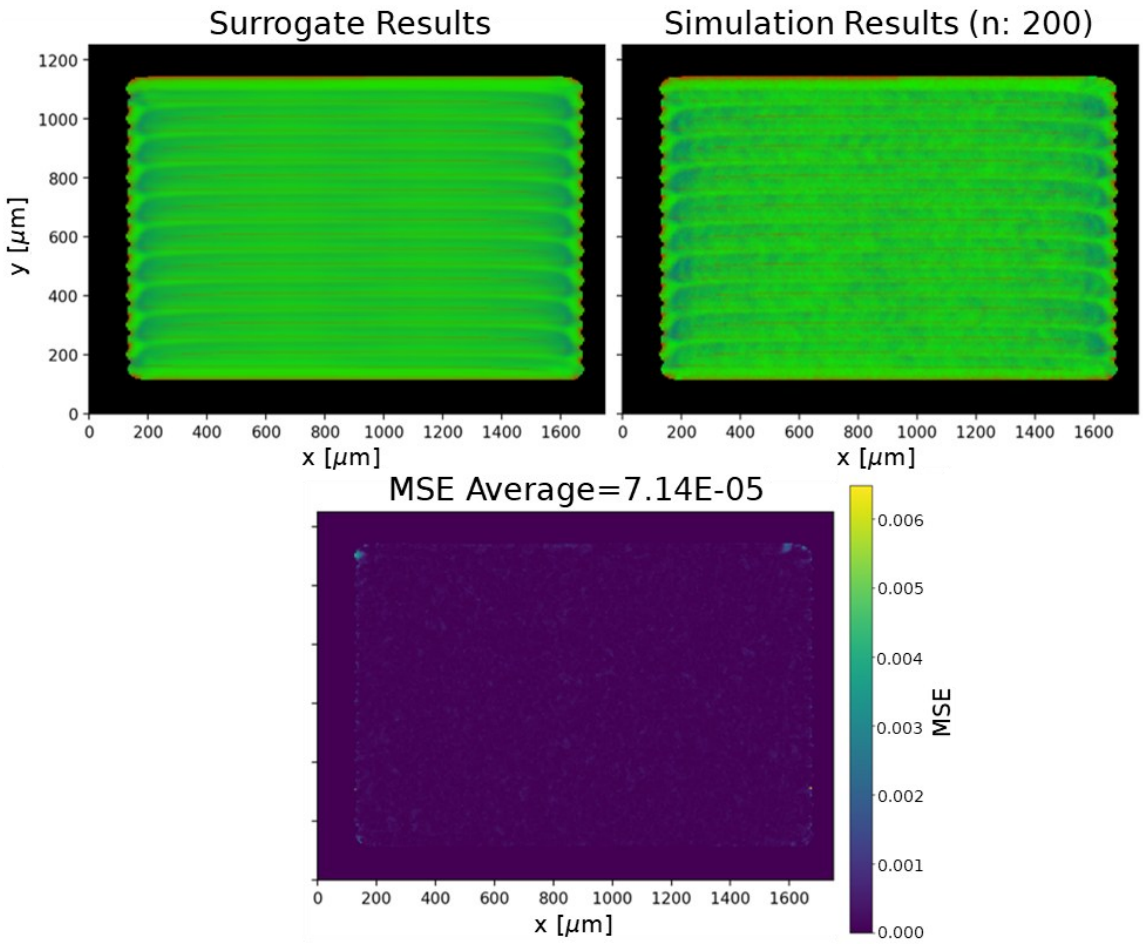

Figure 36 – Prediction from the final version of the model, for the same case presented in the previous figure(P100V100Case1), using the 3-color plotting method presented in Section 2.4.2. Red:0-16μm, Green:16-32μm, Blue:32-48μm. We continue to see that the model's predicted probability maps are smoother than the training data, and achieve a very high accuracy.

These results continue to show that the model's predictions are "smoother" than the training ensemble, indicating that the model is continuing to learn some sort of averaging behavior. Comparing this prediction to a larger ensemble in Figure 37 once again demonstrates that the model's averaging behavior results in its' predictions being more accurate to the underlying ground truth than the training data on which it was trained. For this comparison we use an even larger ensemble, than was used to test the predictions by the initial iteration of the model, of 2000 simulations. At this ensemble size, the results of Section 3.6 indicate that the minimum measurable MSE (i.e. the maximum measurable accuracy) is in the range of 1E-5 to 2E-5. The measured MSE of 1.43E-5 is in this maximum accuracy range, indicating that the model is at least as accurate to the underlying distributions as the 2000-simulation ensemble, and that it continues to be more accurate than the training data for these simple cases.

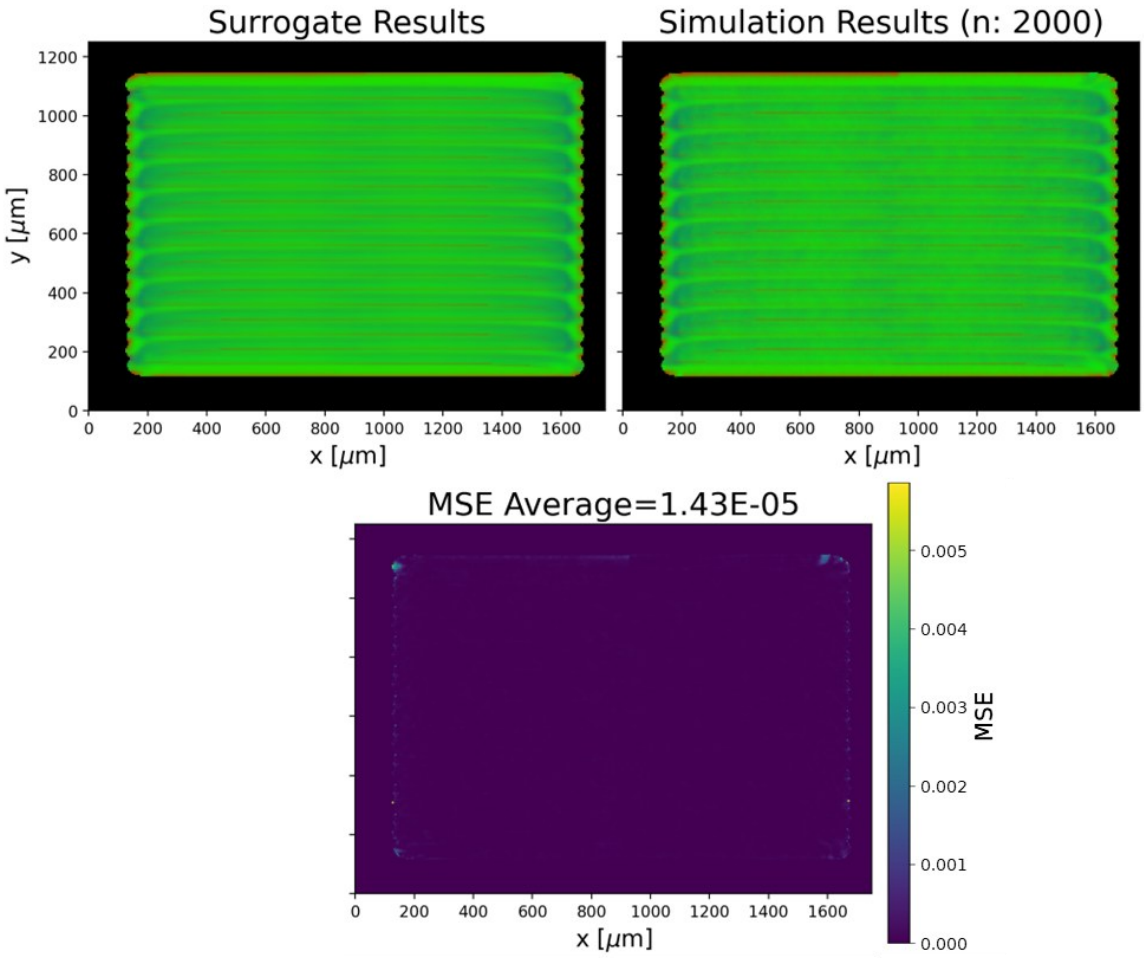

Figure 37 – Prediction from the final version of the model, for the same case presented in the previous figure (P100V100Case1), compared against a very large ensemble of 2000 simulations. Red:0-16μm, Green:16-32μm, Blue:32-48μm. We once again see that the measured accuracy of the model's predictions improves as we increase the size of the ensemble against which we measure.

In addition to performing well for the simple single layer case, we find that the model continues to make reasonable predictions for multi-layer prints that are much more representative of real-world scenarios. As seen in Figure 38, which depicts the top surface and cross-section of the 15-layer P100V100case2, the average MSE for these cases is generally higher than the ideal found in Section 3.6. However, the model captures many of the complex patterns which appear in this statistical representation, with a low MSE across much of the part. While these visualizations show only small slices of the full print, they illustrate that there are once again lower accuracy voxel clusters around the extremities of the part which are skewing the average despite not significantly contributing to the overall statistics.

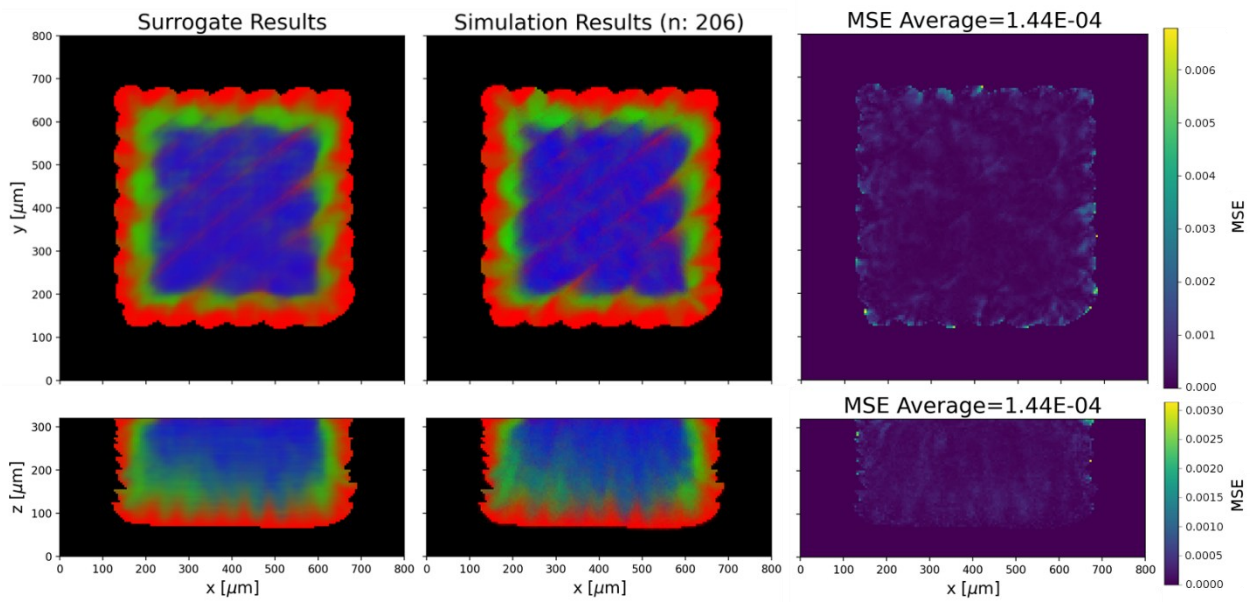

Figure 38 – The top surface and cross section of a 15-layer print, P100V100Case2, showing good agreement between the model's predictions and the corresponding ensemble data. Red:0-40μm, Green:40-80μm, Blue:80-128μm.

While these larger MSEs do not always correspond to visually identifiable differences in the overall statistics, for the larger and more complicated cases there are often small but visually identifiable non-physical artifacts created by the model. This behavior can be seen in Figure 39, which depicts the cross-sections of two different 30-layer cases, where we see visible horizontal banding, particularly in the lower layers. These bands primarily occur near the tops of the individual layers of the prints, indicating that there is something specific to the tops of these layers that the model is struggling to capture despite performing well overall. This behavior was observed to be more prominent in earlier iterations of the model, which used a context window 15 voxels across, when data from all intermediate layers of the print was added to the training set. As such, it may be possible to further reduce this artifacting by further increasing the context window size or eliminating intermediate layers from the training data, in addition to more training and/or a larger dataset.

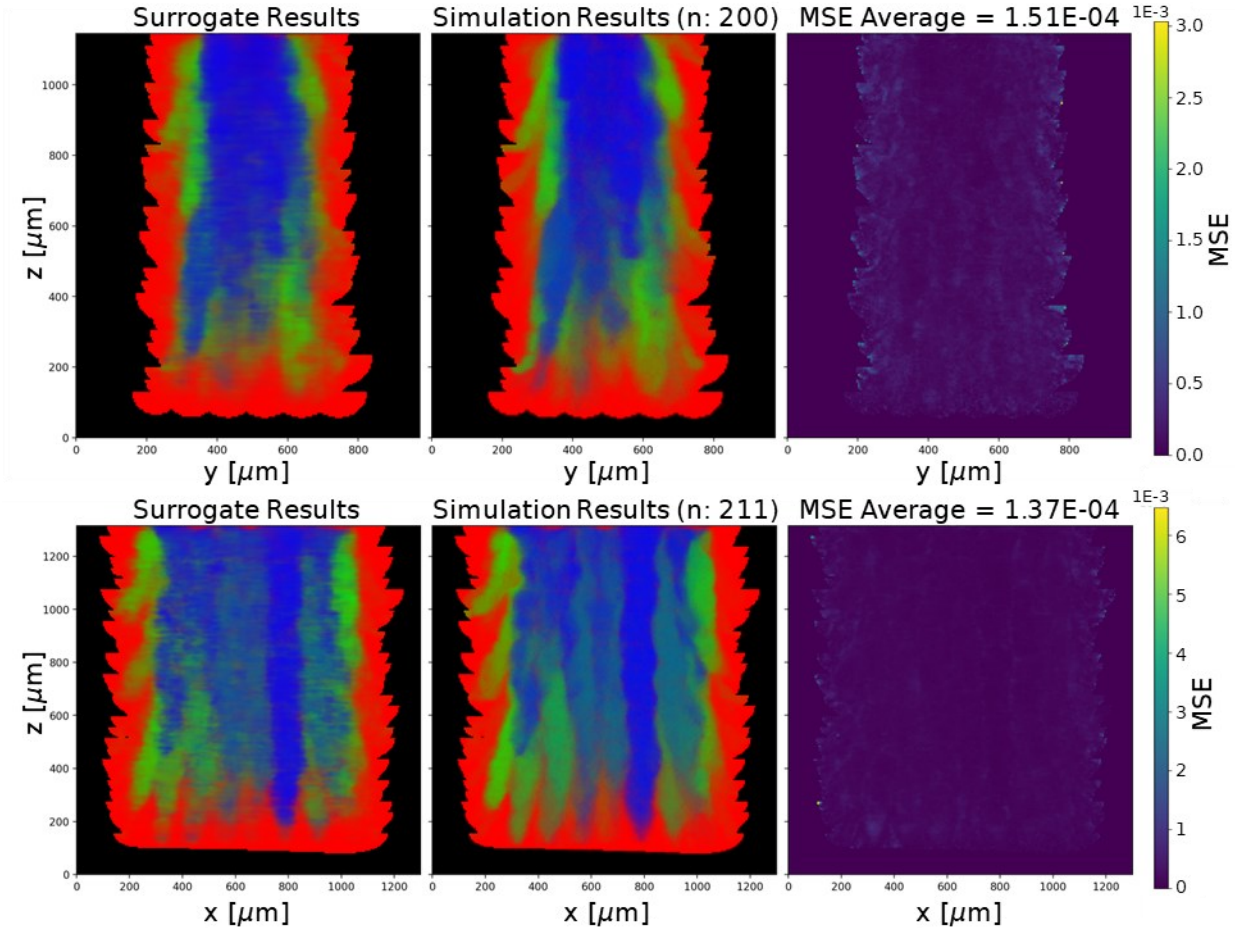

Figure 39 –Cross sections from two different 30-layer cases, both of which exhibit banding artifacts in the model's predictions, though there is no significant increase in the MSE around these artifacts. (Top) P225V72 Red:0-96µm, Green:96-192µm, Blue:192-296µm. (Bottom) P250V75 Red:0-128µm, Green:128-256µm, Blue:256-400µm.)

The only case included in the training data where the model's prediction was notably different across the whole print, despite not having a higher average MSE than the cases already discussed, was the single case where the direction of the scan path was changed by 90 degrees between layers. While there was also a case with 45° of rotation between layers, the majority of the cases included in the training data used an angle of rotation of 67°. For this 90° case, shown in Figure 40, the model appears to underpredict how far the regions of increased probability for small and medium grains (i.e. the red and green) extend from the bottom surface, while overpredicting the magnitude of probability for small grains scattered throughout the full height of the central region. Despite these discrepancies, the model's predictions remain largely correct, with no significant discrepancies between the 99.5% cutoff point of the model's prediction and the ensemble-based statistics. In both cases, the 99.5% cutoff results in the same upper bound of 320µm, though the model predicts that the print averages 0.44% of voxels

in larger grains, versus the 0.34% from the ensemble. This type of small discrepancy in the remaining percentile is present in all cases presented in this work, including those already shown, and is largely inconsequential for these cases, where the model has accurate predictions.

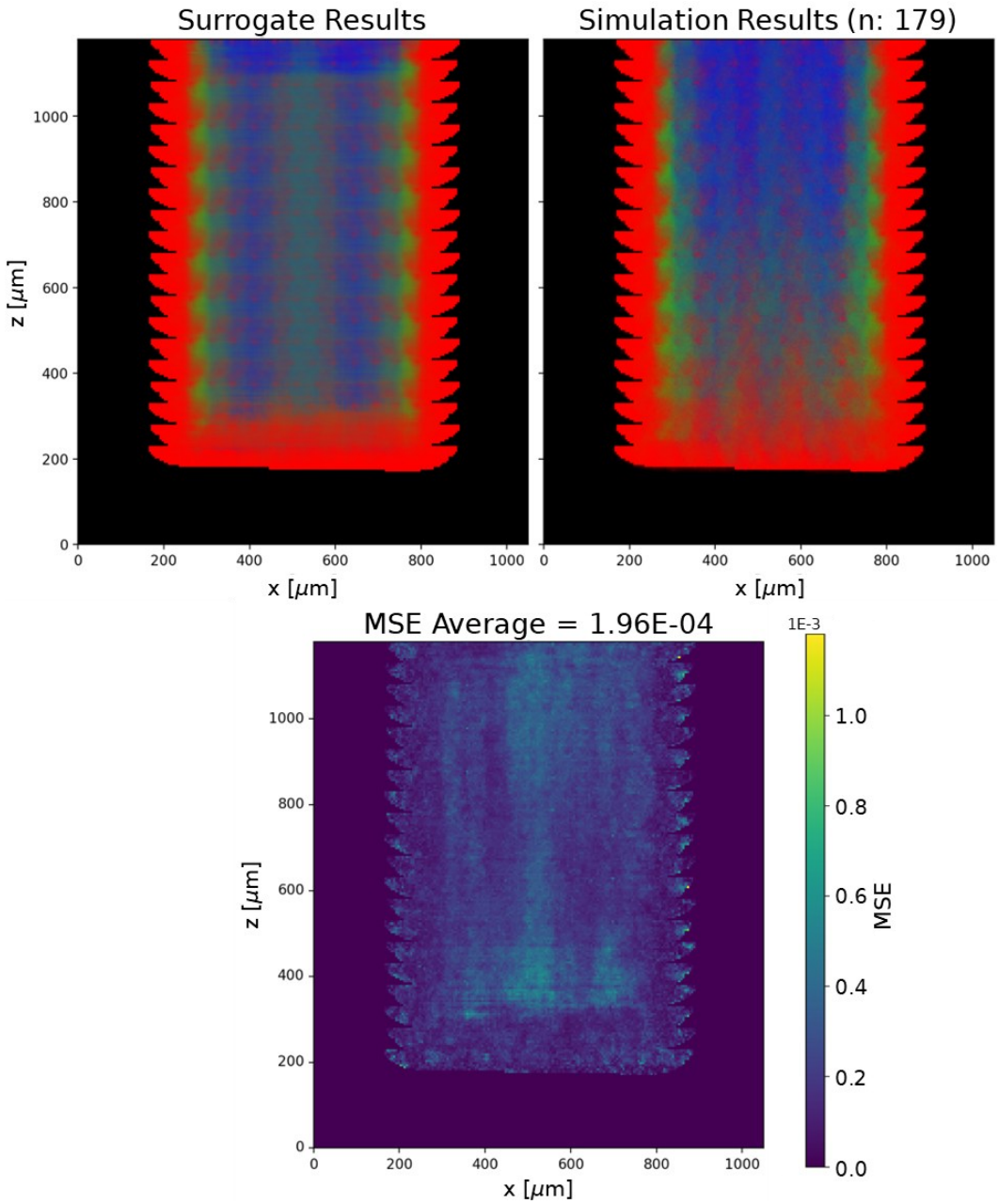

Figure 40 – Cross section from the single case included in our dataset which used a 90-degree rotation of the laser scan path between layers. Red:0-104μm, Green:104-208μm, Blue:208-320μm.

Overall, we have shown here that the surrogate model generally achieves good agreement with equivalent SPPARKS simulation ensembles for the LPBF parameter sets included in the training data. This

includes qualitative measures like predicting the patterns present in the statistics, and quantitatively with reasonable MSEs across much of the print volumes. While these results are from cases included in the training data, 10% of the voxels in each case are new to the model. As these new voxels are randomly selected, poor model performance for these new voxels would show up as randomly distributed voxels of lower accuracy, which we do not consistently observe. Instead, there are more often small clusters of low accuracy voxels, indicating particular relationships the model has not fully reproduced. Though there are often some voxels with lower-than-ideal accuracy, and some artifacts introduced by the model, these low accuracy clusters are not significant enough to skew the predicted grain size ranges in these results. In addition to showing that the model performs well for most of the cases included in the training set, we have also shown that the model generally learns to average the observed relationships in a way that mimics the accuracy of larger ensembles than those used in the training data.

4.4 Influence of Ensemble Size on Model Accuracy

When developing the framework for our surrogate model, the assumption was made that the training data must meet some threshold for minimum ensemble size to capture the stochastic nature of the process in a statistically meaningful way. While it was not clear what this threshold should be, it was reasoned that relatively large ensemble sizes, as compared to the number of bins, would be needed to avoid binning artifacts and minimize stochastic error, and that a statistical resolution of 1% or better would be ideal. For these reasons, the ensembles used to create training data were generally on the order of 200 simulations. However, with the initial results shown in the previous section, we found that the model appears to be capable of making predictions equivalent to much larger ensembles than those used in the training data, indicating that such large ensembles may not be necessary. With this knowledge, we set out to explore in detail the influence of the ensemble size on the model's accuracy, in the hopes that much smaller ensembles could be used to create a model with similar levels of accuracy. This would allow for training data to cover a much larger portion of the LPBF parameter space using the same amount of compute resources for training data generation, which is the current bottleneck of the model creation process.

4.4.1 Method

To analyze this influence, the raw SPPARKS simulations used to create the full training data were re-processed at a variety of ensemble sizes; specifically, statistics were calculated using 5, 10, 25, 50, 75, and 100 of the existing simulations. This method of re-processing the data does introduce some correlation into the data sets, as each increasingly large ensemble contains the previous, but this is not expected to have a meaningful influence on the overall trends. These six new variable sized ensembles were then used to train six new models from scratch, at which point we measured the testing loss of each model against the largest ensemble available for each case in the dataset. To compare these results to our previous observation that the model's predictions appear to more accurately reflect the underlying statistics than the training data, the accuracy of each of the variable sized training ensembles was also measured against the corresponding largest available ensembles. After exploring the implications of these results, we examine how each of the models' predictions for the P225V72 case compare to the full ensemble statistics, both quantitatively and qualitatively.

4.4.2 Results

The results of this analysis are summarized in Figure 41, which shows how the accuracy of the model, with respect to the largest available ensembles, increases as the size of the ensembles in the training data increases. Here we see that the measured accuracy of the model quickly increases as the ensemble size grows, up until around 50-75 simulations, at which point the model's accuracy starts plateauing. Additionally, we see that up until 75 simulations, where the accuracy of the training data meets that of the model, the accuracy of the surrogate model predictions is significantly higher than that of the training data.

Figure 41 – Accuracy of the model vs the largest available SPPARKS ensembles as the size of the SPPARKS ensembles used for training changes. The largest ensemble varies case to case but is generally around 200 simulations. The accuracy of the training ensembles vs the large ensembles is shown for comparison; shows that the model outperforms the data it is trained on. At an ensemble size of 75, the model and training data have similar accuracies vs the largest ensembles.

We have chosen not to measure the accuracy of the training data beyond the 75-simulation ensemble data because there are two cases included in the dataset which have a maximum ensemble size of 100 simulations. Measuring the accuracy of the 100 simulation ensemble training data against the identical 100 simulation target ensembles for these two cases would result in an MSE of zero, artificially increasing the apparent accuracy of the training data. While this does also skew the measured accuracy of the model slightly, it does so by decreasing the maximum measurable accuracy, rather than by artificially improving the apparent accuracy.

These two cases may also be contributing to the measured accuracy of the training data surpassing the measured accuracy of the model at the 75-simulation ensemble mark. When measuring the accuracy of the training data, the 100-simulation ensembles against which we are comparing these cases only contain 33% more samples vs the more than 100% for many of the other cases, meaning they will generally have a lower measured MSE despite having the same number of samples, and by extension the same accuracy versus the ground truth.

A plateau in the model's accuracy as the accuracy of the training data continues to improve, as seen here, is indicative of one of two things. Either the model has some limitations that are keeping it

from improving in accuracy as the accuracy of the data improves, or the model is reaching the maximum accuracy that we can reliably measure using the largest ensembles we have available. To differentiate between these options, we would need to create much larger ensembles against which to measure our model; a decrease in the measured MSE would indicate that the plateau is due to the latter option. Due to the computational expense of this process, this was not a viable option. However, if we take a closer look at the results, Figure 42, we find that the apparent plateau in the model's accuracy is not as definitive as it first appeared.

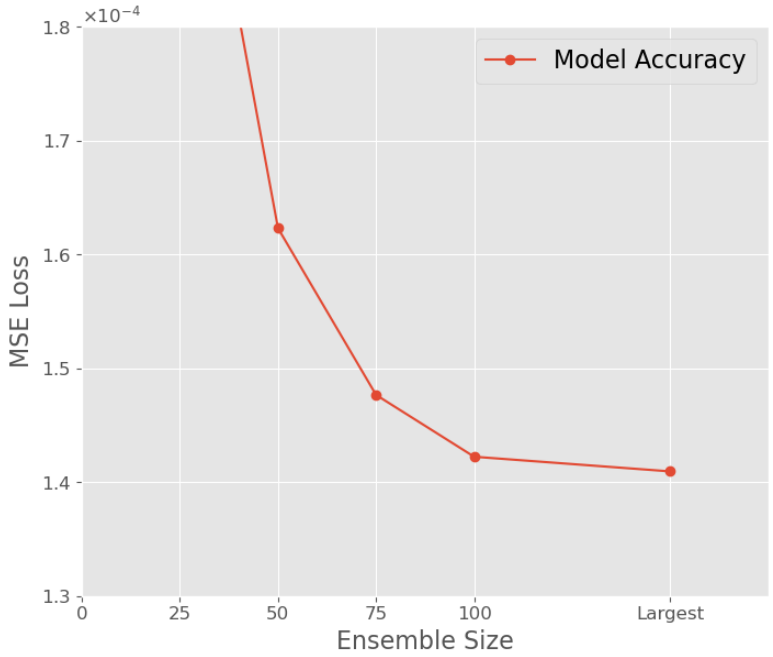

Figure 42 – Accuracy of the model vs the largest available ensembles, as the training ensemble size grows, zoomed in on the region in which the accuracy appeared to plateau. While the rate of improvement does decrease significantly, the model's accuracy does continue to improve in this region.

This behavior is more interesting, as it shows the model is continuing to improve as the training data increases in accuracy, but at a much slower rate. While we cannot say definitively what this indicates without larger ensembles to test against, this seems to indicate that the model quickly approaches the maximum measurable accuracy then continues to make small improvements as the accuracy of the training data improves. Because we know that the model's accuracy is much higher than the training data, but cannot surpass the maximum measurable accuracy, it makes sense that each incremental improvement in the model's accuracy would be much smaller than those observed in the training data itself. While these improvements are small enough that it appears as though the accuracy is

plateauing, this behavior is consistent with a model that is continuing to improve but approaching the maximum measurable accuracy.

To better understand these results, we can visually inspect how the model's predictions compare to the ground truth ensembles. Here we will look at the predictions from each version of the model for the P225V72 case, which contains complicated statistical structures and is in a region of the LPBF parameter space that is sparsely covered by the training data. While the models' accuracies and behaviors varied across the cases in the dataset, we found that the behavior of the model was largely consistent across all cases for each training ensemble size.

4.4.2.1 Five-Simulation Training Ensembles

Starting with the scenario where the model was trained on 5-simulation ensembles, as shown in Figure 43, the model's accuracy is low, with an average MSE of 9.4E-4 for the P225V72 case. This MSE is higher than the average of 6.66E-4 across all cases which was shown in Figure 42, indicating that this is a good test case as it is representative of a worst-case scenario. Despite the high MSE, the model qualitatively predicts the overall patterns present in the statistics to a reasonable degree of accuracy. However, the model does slightly underpredict the maximum likely grain size by 16µm, though this is a minority of all possible outcomes. The ensemble predicts only a 0.29% average probability of larger grains than the 296µm cutoff shown here occurring, which is only marginally higher than the model's 0.24%.

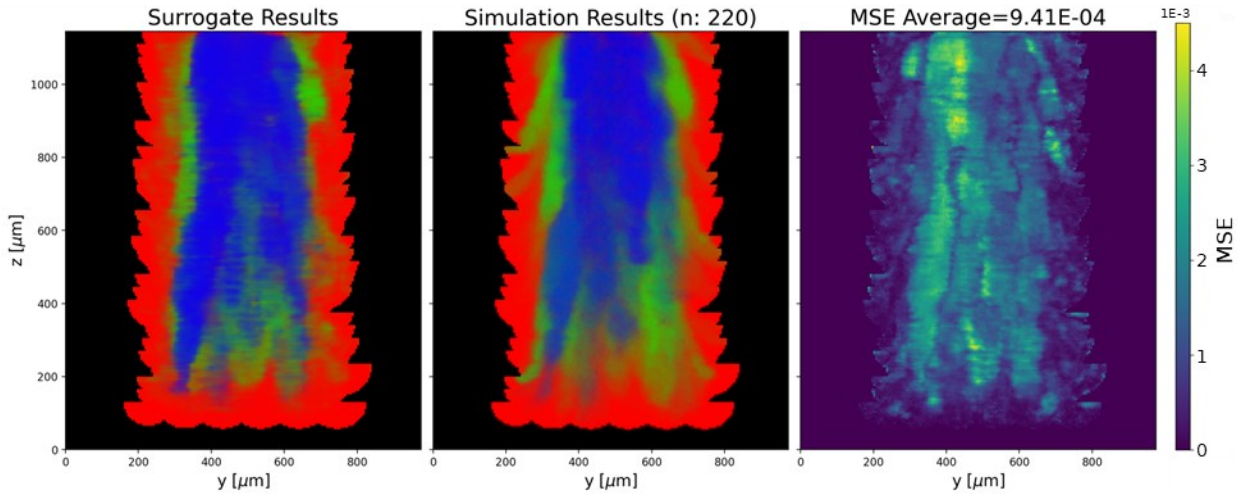

Figure 43 – Model's prediction for the P225V72 case when trained on 5 simulation ensembles, compared against the full ensemble of 220 simulations. Red:0-96µm, Green:96-192µm, Blue:192-296µm.

4.4.2.2 Ten-Simulation Training Ensembles

Increasing the training ensembles to 10 simulations, we see in Figure 44 that the model's predictions are significantly improved with an MSE of 4.75 E-4, and an average MSE of 3.61E-4 across all cases. In addition to no-longer having any large regions of high MSE in the bulk of the print, the model does a better job of capturing some of the patterns present in the statistics. Additionally, the model does a better job at capturing the nuance present in the statistics with fewer sharp transitions between grain size regions. Here the model once again underpredicts the maximum likely grain size by 16µm, but with a similar margin of probability.

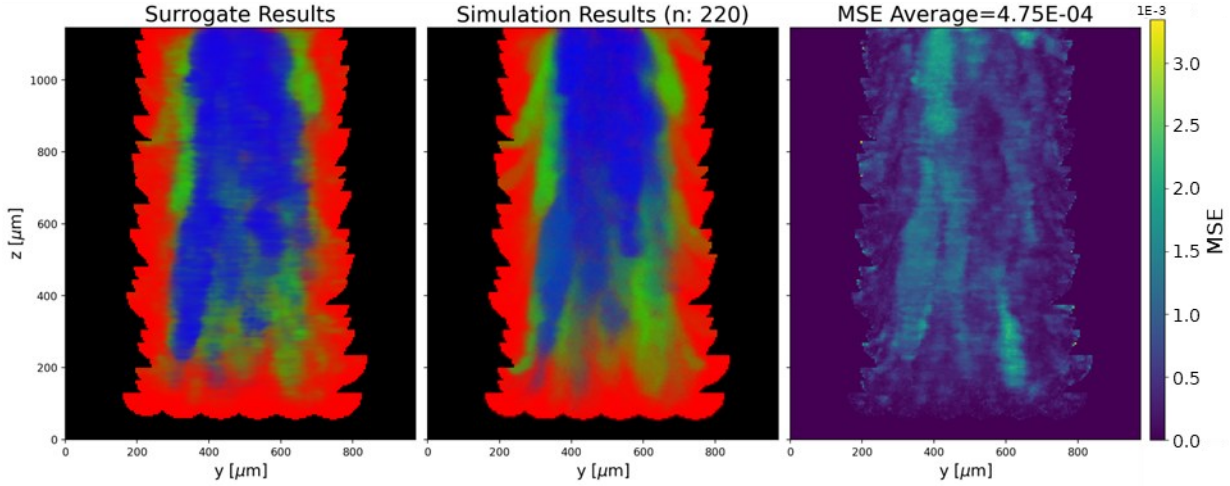

Figure 44 – Model's prediction for the P225V72 case when trained on 10 simulation ensembles, compared against the full ensemble of 220 simulations. Red:0-96µm, Green:96-192µm, Blue:192-296µm.

4.4.2.3 Twenty Five-Simulation Training Ensembles

Increasing the training ensembles to 25 simulations, in Figure 45 we see that the model's predictions are again improved with an average MSE of 2.74E-4, not much behind the average of 2.1E-4 across all cases. This time there are not any significant changes visible in the prediction of the statistical patterns, though it appears much smoother overall, with less pronounced but still present banding artifacts. At this point, the model and ensemble are in agreement about the maximum size of likely grains.

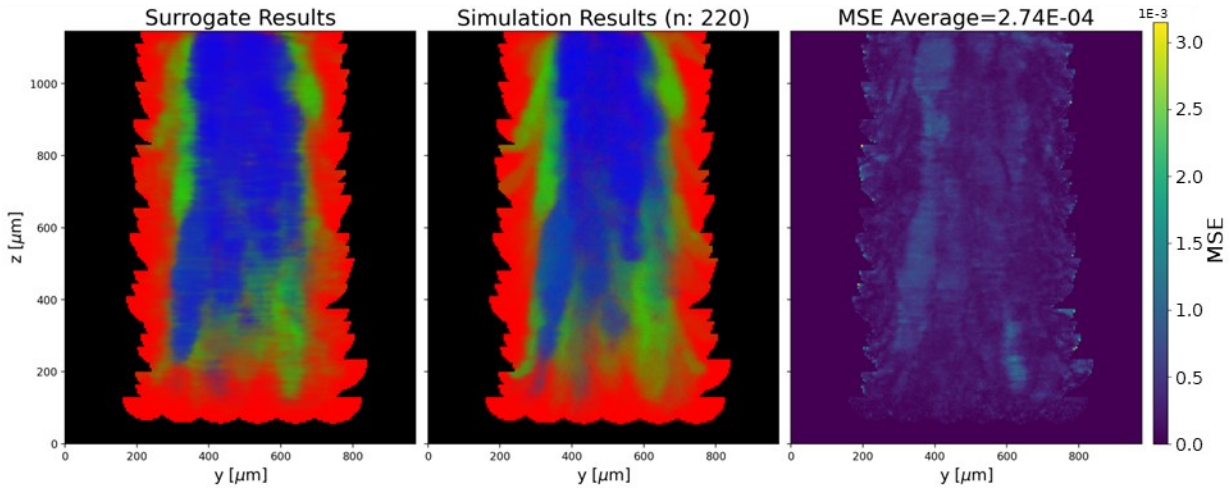

Figure 45 – Model's prediction for the P225V72 case when trained on 25 simulation ensembles, compared against the full ensemble of 220 simulations. Red:0-96μm, Green:96-192μm, Blue:192-296μm.

4.4.2.4 Fifty to One Hundred-Simulation Training Ensembles

The improvements when trained on ensembles of 50, 75, and 100 simulations, Figure 46, are visible but minor, with very little difference between the three. While the accuracy of the model as measured by the MSE continues to improve as the ensemble size increases, the prediction of the patterns present in the statistics does not improve significantly beyond the 25-simulation ensemble-based model.

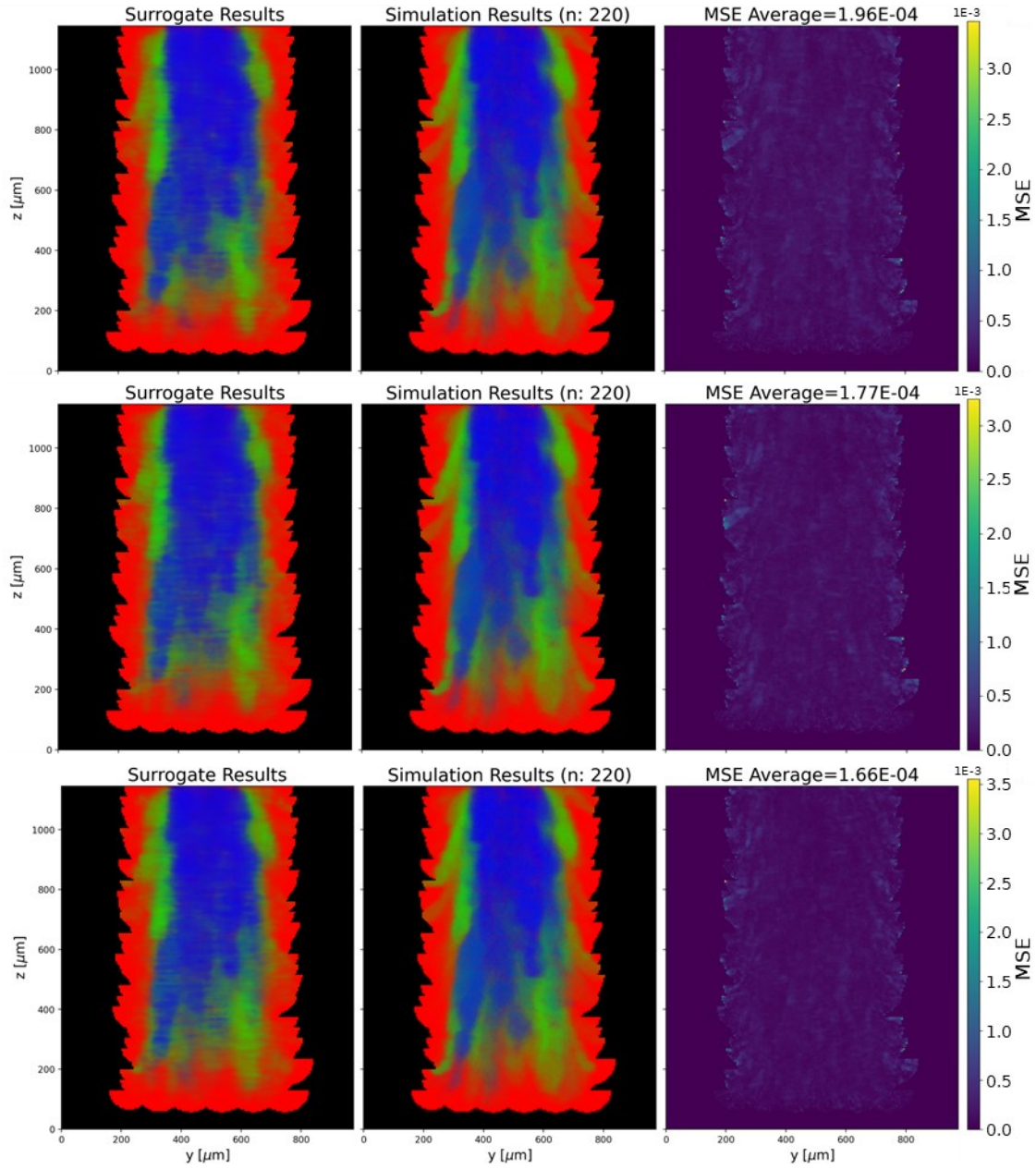

Figure 46 – Model's prediction for the P225V72 case when trained on 50 (top), 75 (middle) and 100 (bottom) simulation ensembles, compared against the full ensemble of 220 simulations. Red:0-96μm, Green:96-192μm, Blue:192-296μm.

These results support the conclusion that the model learns to largely predict the correct statistics with relatively small training ensembles, but continues to make small improvements as the accuracy of the training data increases. However, as shown here and in the previous section, the model continues to produce artifacts in complex cases even when trained on the largest training ensembles, indicating that there are limits to the accuracy of the current iteration of the model architecture.

While we showed that the accuracy of the model does not truly plateau as the ensemble size grows, and the results shown here do not support one, we also cannot rule out any artificial limitations in the measured accuracy due to limitations in the test data. In addition to having already shown that there is an artificial limitation on the measured accuracy of some cases in Section 4.2, we cannot say definitively that none of the statistical patterns shown here would continue to change as the size of the test ensemble increases. Though many of the patterns the model has failed to learn in these results appear more consistent than the striated patterns that disappeared with very large ensembles, it is possible that some would appear closer to the model's predictions given a large enough ensemble size.

4.5 Predictive Power Analysis

While we have shown that the model performs well when making predictions about cases included in the training data, the intended use for this model – allowing for the rapid iteration of LPBF process parameters – makes the accuracy of predictions about cases not included in the training data more important. To better understand how the surrogate model might perform in such scenarios, here we analyze the predictive power of the model for cases not included in the test data. In doing so we will gain some understanding of how the relationship between the process parameters included in the training data and the parameters in the predicted case influence the accuracy of the model.

4.5.1 Methods

Rather than creating entirely new cases on which to test our model, a computationally expensive process, subsets of the full dataset used for training the model were categorized according to specific scenarios we wished to test. These scenarios were then tested one-by-one by removing the corresponding cases from the training data, training a new model from scratch, and using the newly trained model to make predictions about the excluded cases. Because attempts were made to maximize variation in the training data, the cases excluded in these scenarios often varied by several process and geometric parameters beyond those use for assigning them to a scenario. However, for the most part, the scenarios are:

1. Interpolation: Cases with powers and velocities central to the main cluster of cases
2. Hatch Spacing: Identical powers and velocities to other cases but with different hatch spacings

3. Extrapolation: Cases that are increasingly far from the main cluster of cases
4. Sparse Interpolation: Only two neighboring cases, which are far apart
5. Rotation: Cases with powers and velocities in the extrapolation and sparse interpolation regimes, with new angles of rotation for the scan direction between layers
6. Far Extrapolation: A case with a velocity higher than any other

These cases are summarized by the power-velocity plot in Figure 47 and by Table 5.

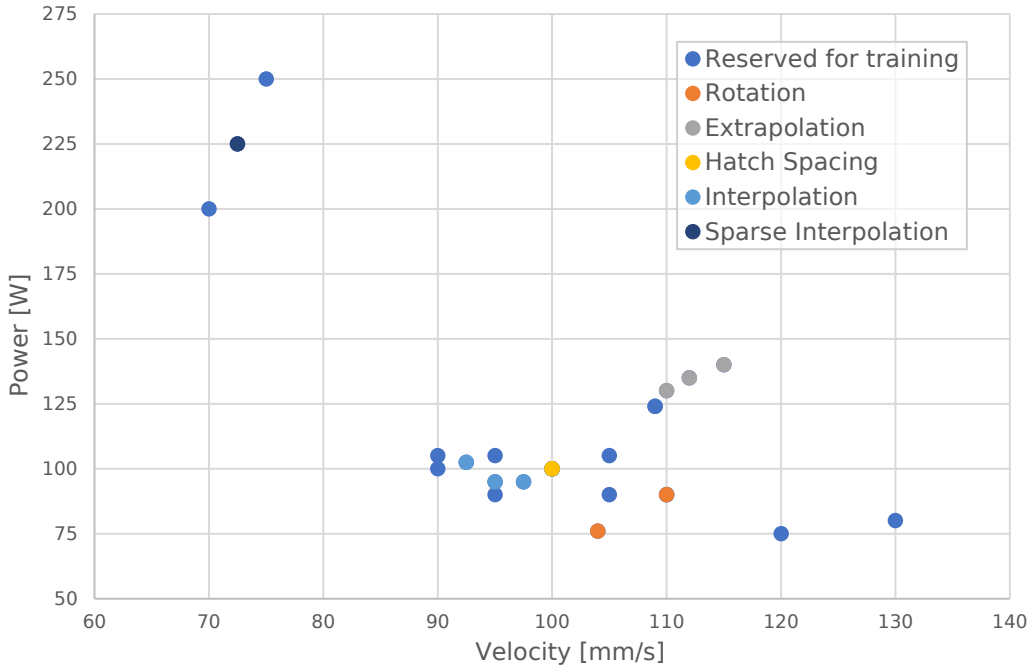

Figure 47 – Plot showing the process parameters used for both training and testing the model. For a given scenario, all data except the corresponding testing set was used for training the model.

Table 5 – Summary of the cases excluded for testing the various predictive power scenarios.

Name	Test Case Type	Power (W)	Velocity (mm/s)	Hatch Spacing (µm)	Layer Height (µm)	Rotation (deg)
P95V95	Interpolation	95	95	50	N/A	N/A
P95V97.5	Interpolation	95	97.5	50	N/A	N/A
P102.5V92.5	Interpolation	102.5	92.5	50	N/A	N/A
P100V100H45	Hatch Spacing	100	100	45	N/A	N/A
P100V100H40	Hatch Spacing	100	100	40	N/A	N/A
P130V110	Extrapolation	130	110	50	20	67
P135V112	Extrapolation	135	112	50	20	67
P140V115	Extrapolation	140	115	50	20	67
P225V72.5	Sparse Interpolation	225	72.5	105	35	67
P90V110R45	45° Rotation	90	110	50	25	45
P76V104R90	90° Rotation	76	104	50	25	90
P80V130	Far Extrapolation	80	130	25	15	67

4.5.2 Limitations

For the purposes of this analysis, we have chosen the cases for each scenario using a limited subset of the simulation and LPBF process parameters, without focusing on any other differences that may be present. In addition to not fully capturing the differences that may be present due to parameters which were not accounted for, the complexity of the relationships between process-parameters and printing outcome makes grouping cases based only on process parameters inherently reductionistic. While the Power-Velocity plot in Figure 47 illustrates how each case can be classified as being in an interpolation or extrapolation regime, the data the model actually sees is much more complex. This is illustrated in Figure 48, which shows the distributions of each of the nine thermal characteristics for all the voxels in a subset of the cases, where each color corresponds to a different case. This method of presenting the thermal characteristics is a simplified view, as in reality the datapoints in each distribution are paired with datapoints from each other characteristic, and the model itself is exposed to groups of more than 15 thousand such pairings at a time. Despite this simplification, this illustrates that establishing any given case as existing in an extrapolation or interpolation regime using the actual thermal data would be difficult to impossible, as there is much overlap between the cases in the coverage of each thermal characteristic.

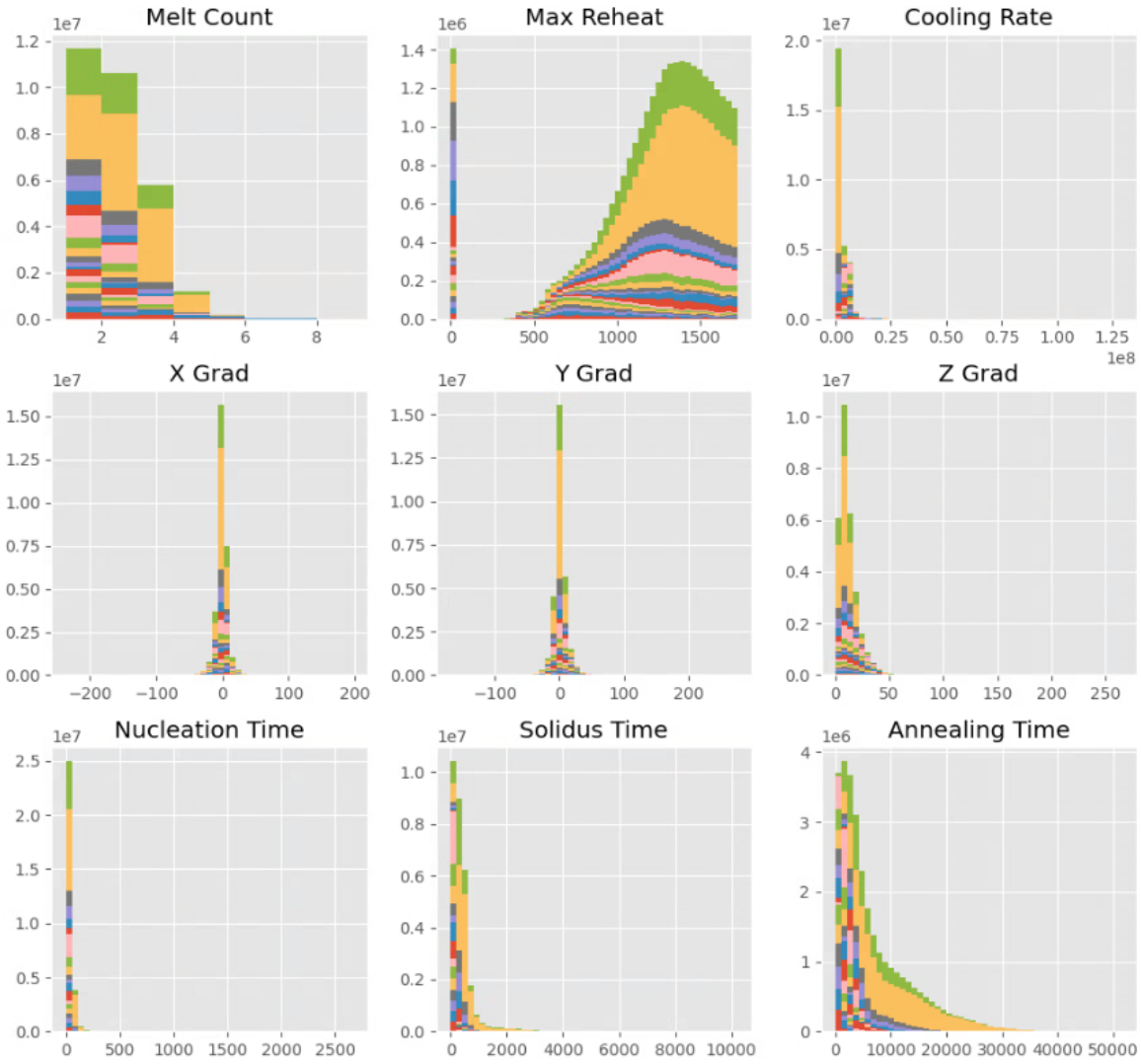

Figure 48 – Stacked histograms showing the distributions of the thermal characteristics across all voxels in several cases. Each case is represented by a different color. This is a simplified view that does not account for the fact that each voxel has a specific pairing of the nine thermal characteristics.

Despite this simplification, we believe that the analysis presented here is insightful for the purposes of understanding how the model behaves when making predictions about new process-parameter cases. Additionally, while we do not attempt here to understand the relationships between the model's performance and the thermal characteristics it uses to make predictions, in Section 5.3 we further explore the relationships that exist between the thermal characteristics of the various process-parameter cases.

4.5.3 Results

4.5.3.1 Interpolation: Cases with powers and velocities central to the main cluster of cases

Starting with the simplest scenario, where the cases being predicted are simple single layer cases that have many nearby training cases surrounding them in the process parameter space. Because these are single-layer cases and therefore have a very limited maximum grain size, these cases are not particularly useful for understanding the predictive power of the model for real world scenarios However, the simplicity of these cases, and their similarity to other cases which were not excluded from the training set, makes them useful for understanding the best case scenario for the model's predictive power given small variations. This is important because, while the model's ability to accurately make predictions for complex cases where there is little available data is arguably more useful from a process exploration point of view, predicting the influence of small changes to the process parameters is important for any use as part of an optimization cycle/pipeline.

The resulting predictions from the model, shown in Figure 49, are quite accurate for all three cases, with average MSEs of 8.71E-5 or lower, which is in line with the maximum measurable accuracies for the ensemble sizes against which we are measuring. While there are some voxels with higher errors, particularly in the P95V97.5 case with a peak over 1.2E-2, these low accuracy voxels are confined to small clusters at the very edges of the print and don't have a significant overall impact. The most insightful feature of these results is how well the model predicts the shape and intensity of the blue-green medium-to-large grain patterns in the vicinity of every turning point for the laser. While the patterns for the small grain (red) regions are fairly consistent along the edges of all single-layer cases, these blue-green regions vary more visibly, making them one of the few indicators of model accuracy for these cases.

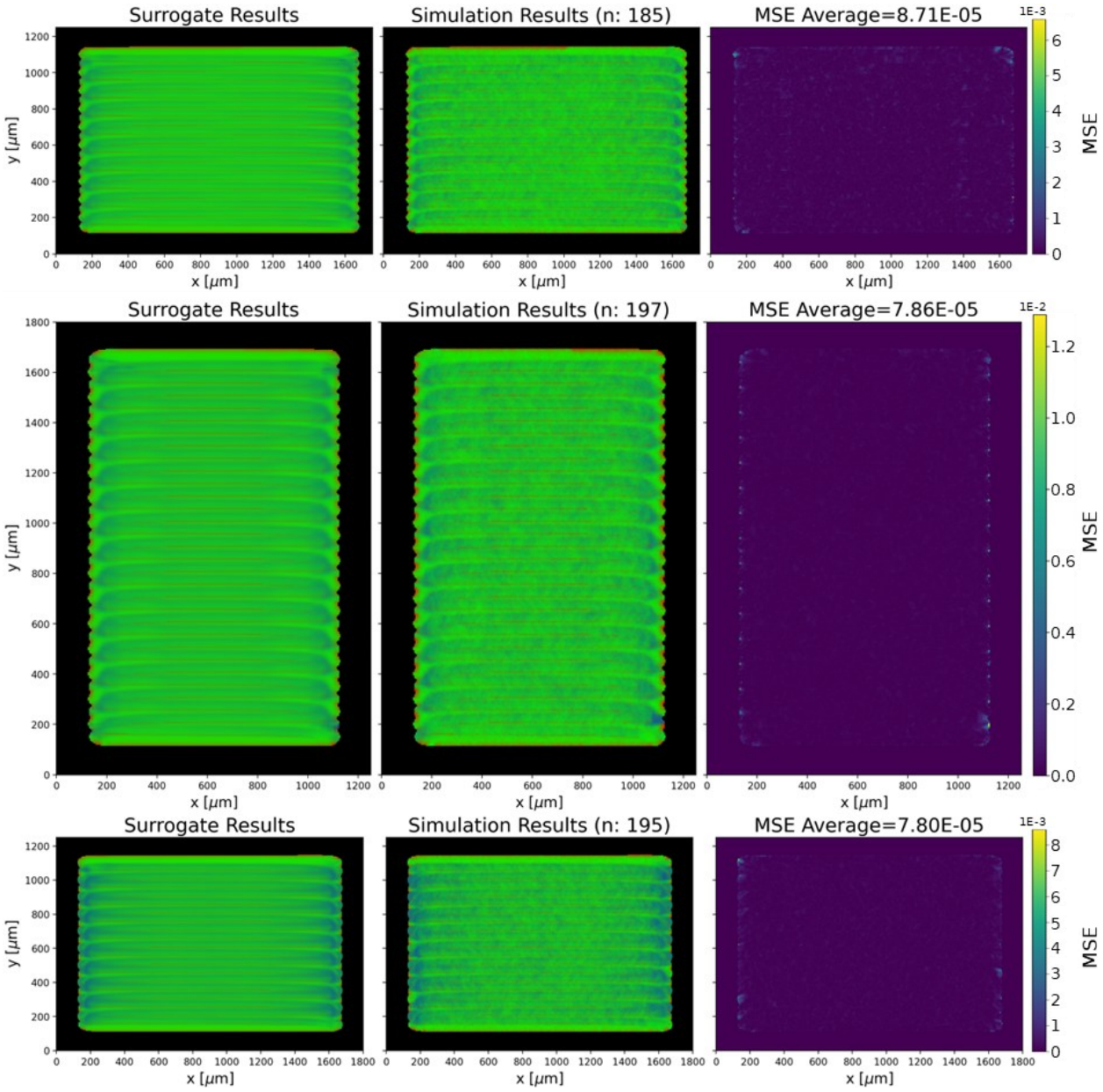

Figure 49 – The model's predictions for the top surface of all three interpolation cases: Top) P95V95 Middle) P95V97.5 Bottom) P102.5V92.5. All three predictions are in good agreement with the results from the simulation-based ensembles and have a low MSE. Red:0-16μm, Green:16-32μm, Blue:32-48μm.

4.5.3.2 Hatch Spacing: Identical powers and velocities to other cases but with different hatch spacings

The hatch spacing cases were selected to test the influence of just the hatch spacing on the model's predictive power. These cases use identical laser parameters to other cases in the training dataset and are simple single-layer cases, which we have already demonstrated the model performs well on. However, with the exception of a single case with a hatch spacing of 25μm, the majority of the cases in the dataset use hatch spacings of 50μm or hatch spacings greater than 65μm. By decreasing the hatch

spacing by small steps down from 50µm, we can directly explore the influence of a parameter that is sparsely explored.

Starting with the 45µm case P100V100H45, with a hatch spacing just 5µm away from the closest training data and identical powers and velocities, we see in Figure 50 that the model's predictions align quite well with the test ensemble data. The most obvious difference between the model's prediction and our ground truth is a slight underprediction in the prevalence of small grains along the edges. Additionally, the model also seems to slightly underpredict the prevalence of the largest grains – in fact this is the largest source of error. This is most visible in the region of the 4th turning feature working up the right side, where our ground truth data indicates that it is very likely for the largest grains to extend all the way to the edge. While these "large grains" are quite small on the scale of all grain sizes found in our data, this behavior is somewhat unusual in the dataset, so it is not surprising that the model's prediction is not in perfect agreement here.

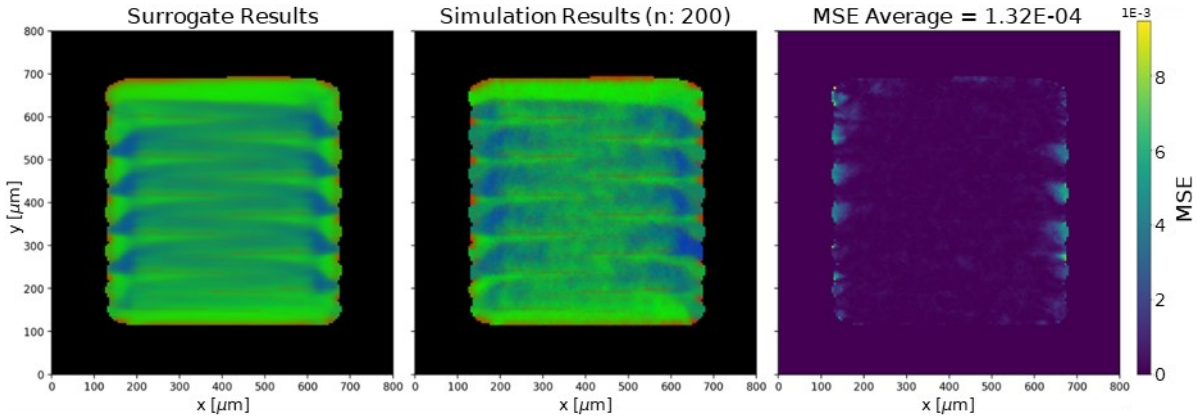

Figure 50 – Model's predicted probabilities for the top surface of the P100V100H45 case, a 1-layer raster scan with a hatch spacing of 45µm. The model's predictions are reasonably accurate when compared to our test data, but it does not capture some of the features around the turns. Red:0-16µm, Green:16-32µm, Blue:32-48µm.

Moving on to the 40µm hatch spacing case P100V100H40, in Figure 51, we find that the model's accuracy is decreased slightly from the 45µm case. Interestingly, in this case, the test data indicates slightly lower probabilities for the largest grains to extend to the edges of the turns, so the error in these regions is slightly less obvious, but not significantly lower. While the MSE plot does appear to indicate less error in these regions, it is primarily due to a much higher maximum error at the tail end of the print (the top left in this view).

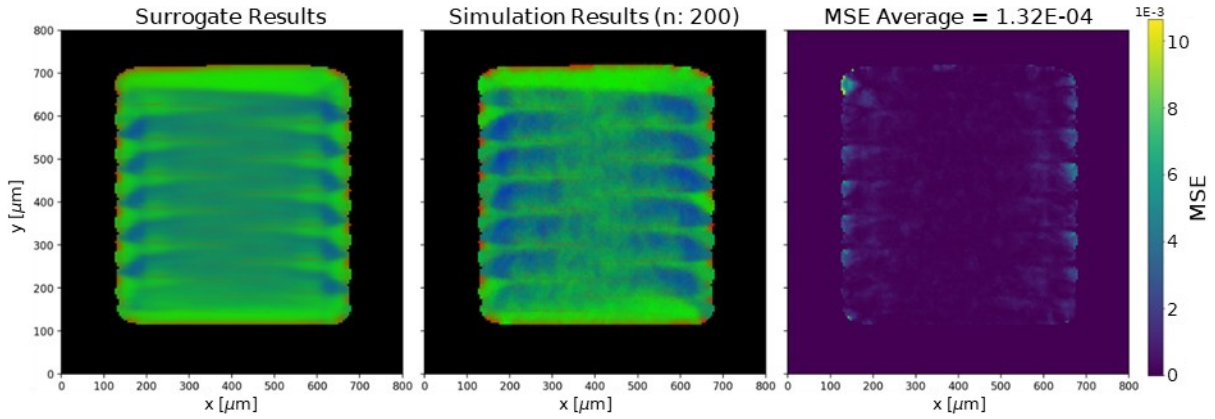

Figure 51 – Model's predicted probabilities for the top surface of the P100V100H40 case, a 1-layer raster scan with a hatch spacing of 40μm. The model's predictions are slightly less accurate than for the 45μm case, but the discrepancies in the vicinity of the turns are less obvious. Red:0-16μm, Green:16-32μm, Blue:32-48μm.

These results indicate that at some level the model was able to learn the influence of the physical distances between features of the thermal characteristics on the microstructure and apply those influences when making predictions about cases that were otherwise nearly identical to training data. As the training data does not expose the model to any variations in the hatch spacing this small, any relationships it was able to learn came from much larger differences in hatch spacing, combined with other indirect influences on the distances between features. Overall, while this case is not indicative of the model's predictive power for particularly unique cases, it does show that the model appears to be able to apply the relationships it's learning to adapt cases learned from the training data.

4.5.3.3 Extrapolation: Cases that are increasingly far from the main cluster of cases

Moving on to the extrapolation scenario, where the process parameters of the test cases get increasingly far from the main cluster of training cases, we find that the model performs reasonably well for the initial layers of these prints. This can be seen in the Figure 52, Figure 53, and Figure 54, which show the top surface of the 3rd layer for all three cases, starting with P130V110 and ending with P140V115. We find that the average MSE is higher than ideal for all three cases, and all slightly overpredict the abundance of the largest grains. However, the maximum expected grain sizes are largely in line with that of the ensembles, overpredicting the largest grain size by 8μm in all three cases using the 99.5% cutoff. Though, somewhat counterintuitively, in all three cases the ensembles have a slightly higher remaining percentage in larger bin sizes (e.g. in the P135V112 case the ensemble has an average of 0.29% remaining vs 0.23% from the model).

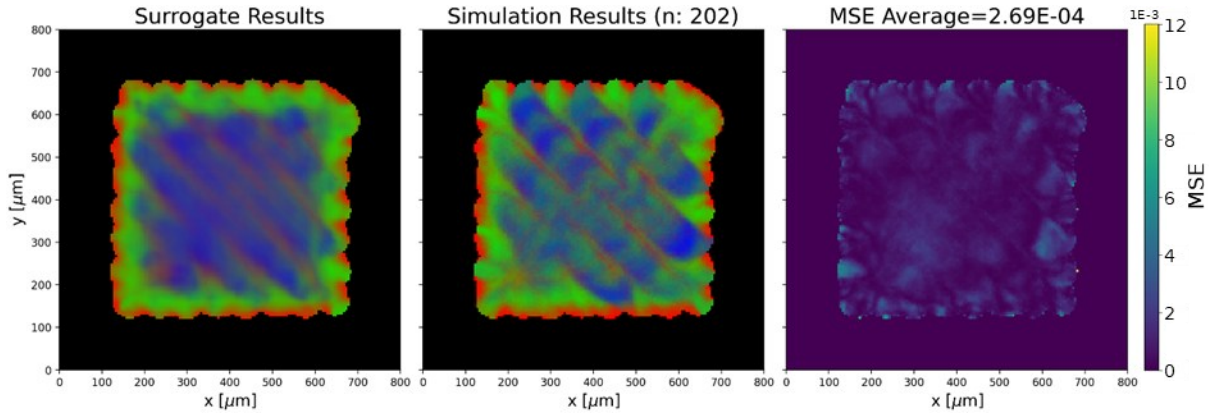

Figure 52 – Top surface of the 3rd layer of the extrapolation case P130V110. Red:0-24μm, Green:24-48μm, Blue:48-80μm.

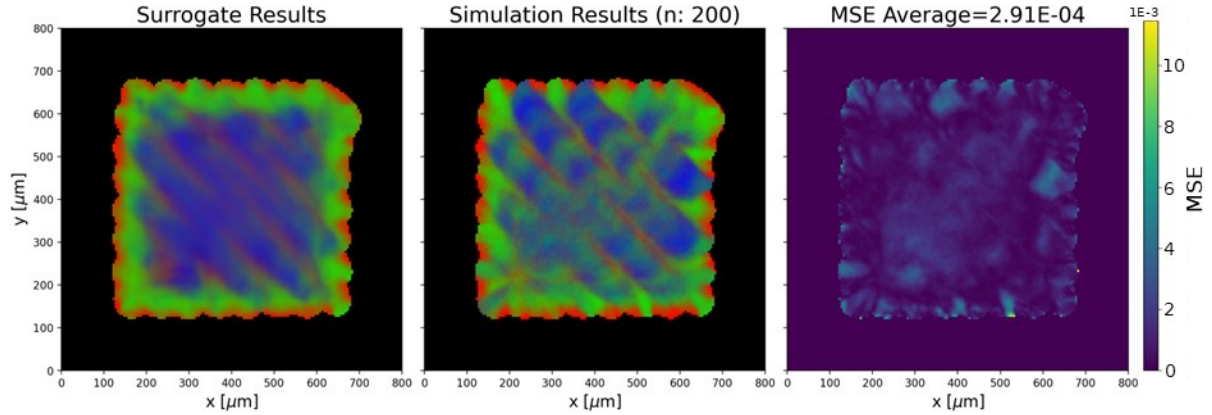

Figure 53 Top surface of the 3rd layer of the extrapolation case P135V112. Red:0-24μm, Green:24-48μm, Blue:48-80μm.

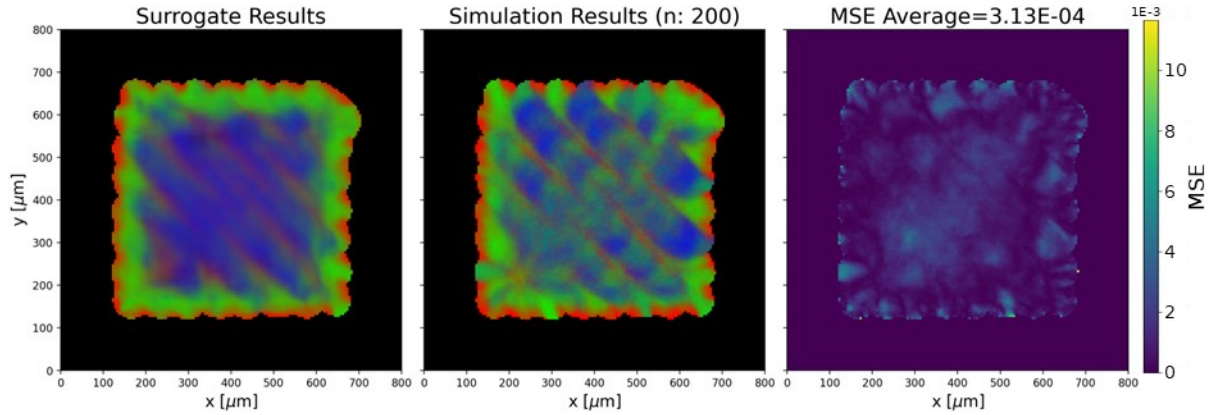

Figure 54 – Top surface of the 3rd layer of the extrapolation case P140V115. Red:0-24μm, Green:24-48μm, Blue:48-80μm.

Moving on to the 4th and 5th layers of the prints, we find that the accuracy of the model dramatically decreases, with a significant overprediction of the size of the largest likely grains, as shown in Figure 55, Figure 56, and Figure 57. These discrepancies were once again fairly consistent across all three cases with the P135V112 and P140V115 overpredicting the largest likely grain size by 48μm in the 5-layer print, and the P130V110 by 40μm. In retrospect, this finding makes sense, as there is only one

other case in the dataset which stops at 5-layers, all other 5-layer cases the model has seen also include subsequent layers in the training data. Given that the model does not have a large enough context window to see the entire height of these 5-layer cases and has not been trained that these specific thermal characteristics correspond to smaller grains, it is not surprising that the model might be interpreting these thermal characteristics as belonging to taller parts which would have taller grains. This is in line with our observations in the decreased accuracy of the model when we first added intermediate layers to the training data, which led to the implementation of a larger context window and removal of most of the intermediate layers from the training data. These results suggest that while those changes remedied the overall accuracy of the model for the cases in the test/train set, they were not sufficient to improve the model's accuracy for new few-layer cases.

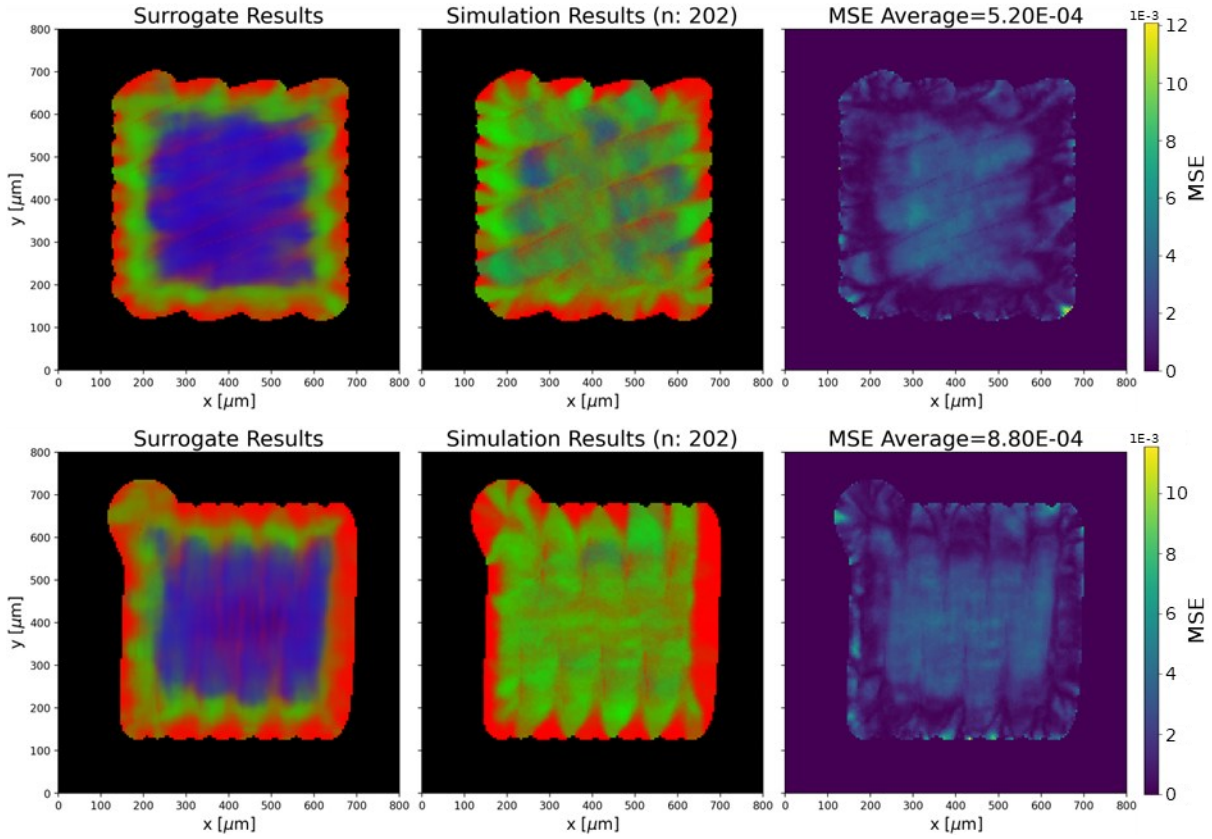

Figure 55 – Top surface of the 4th (top) and 5th (bottom) layers of the extrapolation case P130V110. Top) Red:0-32μm, Green:32-64μm, Blue:64-112μm. Bottom) Red:0-40μm, Green:40-80μm, Blue:80-128μm.

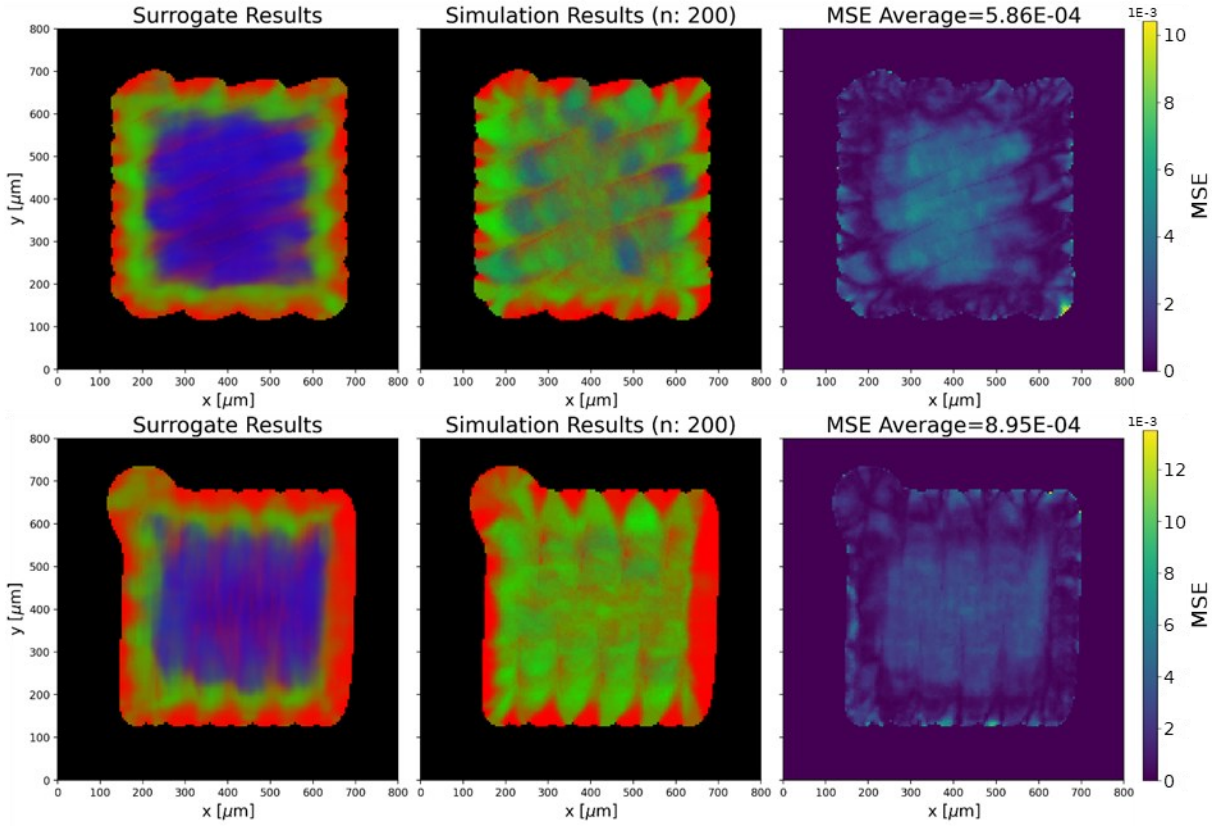

Figure 56 – Top surface of the 4th (top) and 5th (bottom) layers of the extrapolation case P135V112. Top) Red:0-32µm, Green:32-64µm, Blue:64-112µm. Bottom) Red:0-40µm, Green:40-80µm, Blue:80-136µm.

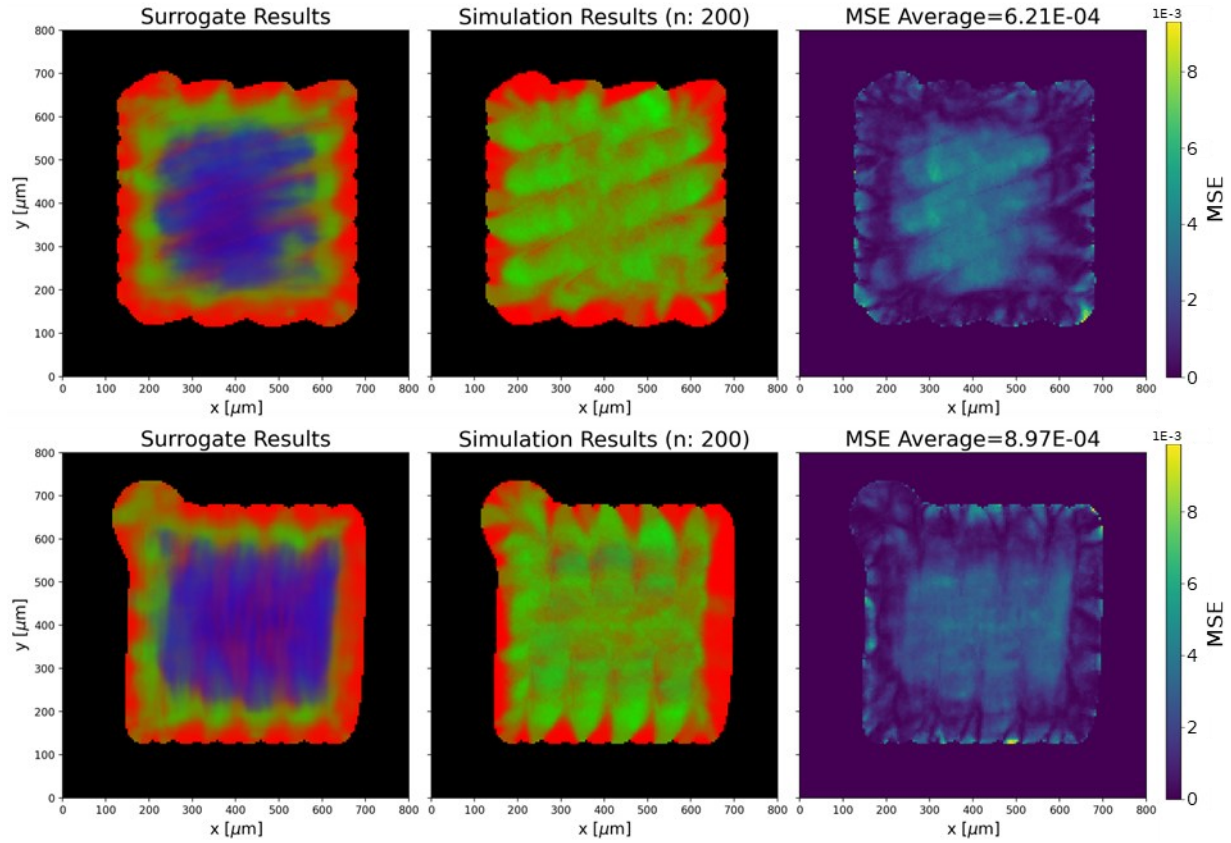

Figure 57 – Top surface of the 4th (top) and 5th (bottom) layers of the extrapolation case P140V115. Top) Red:0-32µm, Green:32-64µm, Blue:64-112µm. Bottom) Red:0-40µm, Green:40-80µm, Blue:80-136µm.

As we do not have taller versions of these cases against which to test, we cannot say for certain how the model would perform in these cases for the approximately 1mm tall print size we use for many layer prints. It does stand to reason that overpredicting the grain size in the context of short prints does not necessarily indicate poor performance for taller prints, where such grains would be present. However, as using data-based models for extrapolation of complex phenomena is not generally recommended, the next scenario (sparse interpolation) provides more useful insight into worst case scenarios for real world use.

4.5.3.4 Sparse Interpolation: Only two neighboring cases which are far apart

The sparse interpolation scenario is representative of a potential real world use case where large parts of the LPBF process space have been covered in the training data using only a handful of cases. Here we have a potential worst-case version of such a scenario, where only two such training cases are in the vicinity of the sparse interpolation case (P225V72.5) we wish to make predictions for. Starting with the first layers of the print, in Figure 58, which shows the top surface of layers 1 and 2, we find that the

model's predictions start off reasonable. While the MSEs are a higher than ideal, and the model slightly overpredicts the maximum likely grain size by 16µm and 8µm respectively, the model largely reproduces the correct patterns.

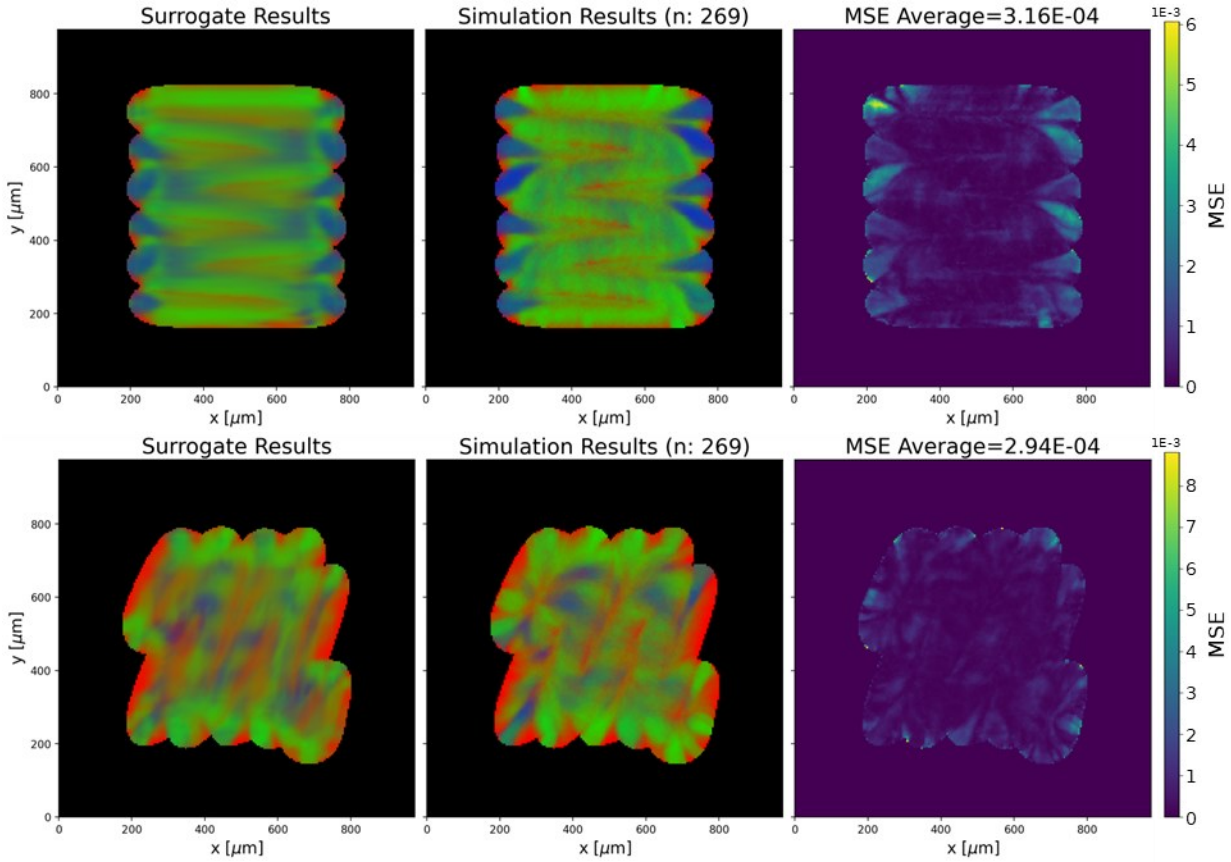

Figure 58 – Top surface of the 1st (top) and 2nd (bottom) layers of the P225V72.5 sparse interpolation case. Top) Red:0-24µm, Green:24-48µm, Blue:48-80µm. Bottom) Red:0-32µm, Green:32-64µm, Blue:64-96µm.

Starting with the 3rd layer in Figure 59 we get a dramatic drop-off in the model's accuracy, with an overprediction of the size of the large likely grains by 248µm according to the 99.5% cutoff, resulting in every voxel of the ensemble results belonging to the smallest (red) super-bin. We can resolve this visualization problem by determining the cutoff point using the last bin which crosses below the 0.5% threshold from one bin to the next[2], the results of which are shown in Figure 60. With this new, much lower cutoff point, we find that the majority of the model's prediction actually does a reasonable job of predicting the patterns present in the statistics, though there are regions with low chances of any of the displayed grain sizes occurring due to being above the cutoff. This lower cutoff is only 16µm larger than

[2] See Section 2.4.2 for a refresher on this distinction.

the same cutoff determined using the ensemble data, a significant improvement in agreement between the model and ground truth. Additionally, we find that the average percentage of all distributions which are above this cutoff point is just 2.55%. However, as the number of voxels which indicate larger grain sizes are a small fraction of all voxels, the actual remaining percentage for these voxels is much higher.

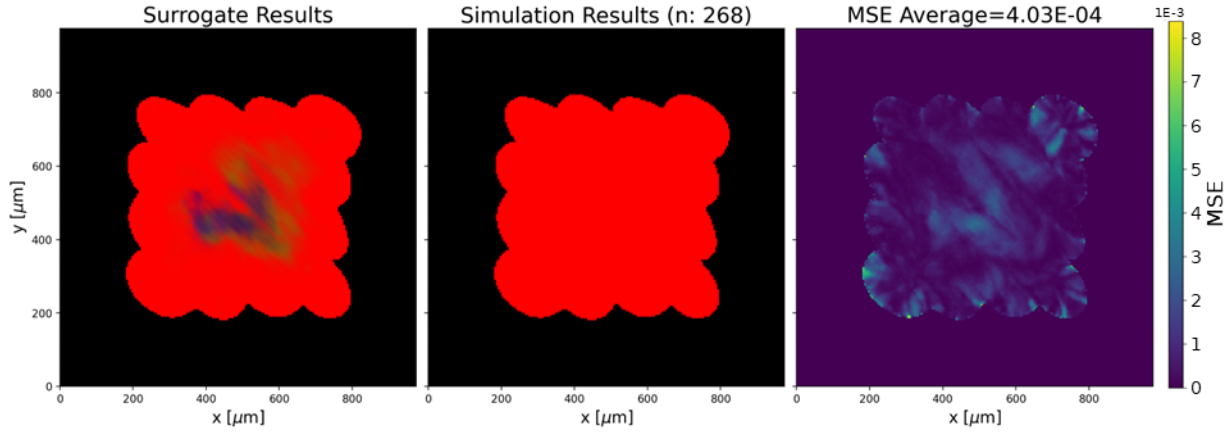

Figure 59 – Top surface of the 3rd layer of the P225V72.5 sparse interpolation case. The model significantly overpredicts the size of the largest likely grains, by 248μm, resulting in the majority of every distribution falling in the smallest (red) super-bin. Red:0-112μm, Green:112-224μm, Blue:224-352μm.

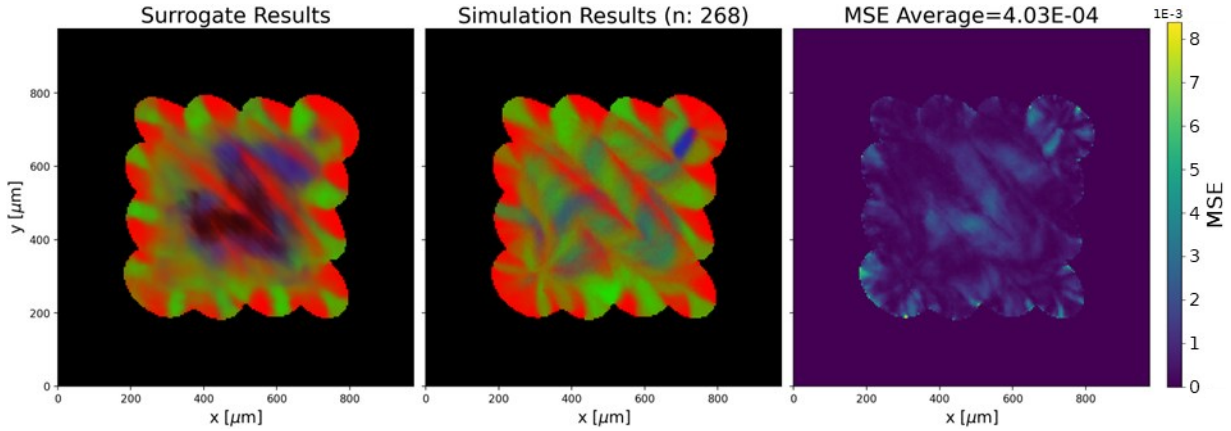

Figure 60 – Top surface of the 3rd layer of the P225V72.5 sparse interpolation case. Determining the upper cutoff using the 0.5% method results in a much more reasonable representation, with the model overpredicting the grain size by just 16μm, though some distributions are cut off. Red:0-40μm, Green:40-80μm, Blue:80-120μm.

Interestingly, the model's predictions for the 4th and 5th layers improve slightly, overpredicting likely grain sizes by 96μm and 72μm respectively, as can be seen in Figure 61. Surprisingly, the overprediction of the likely grain sizes in the 5-layer case is slightly lower in part due to the model predicting a smaller maximum likely grain size than it did for the 4-layer case. In both cases, using the 0.5% method to determine the upper grain size cutoff for visualization, as seen in Figure 62, once again results in an improved comparison to the ensemble data. However, unlike with previous cases this lower

cutoff does not introduce any large regions where data is visibly getting cut off (resulting in black patches), indicating that there are not generally regions with a high probability of being wrong, but instead large regions with a low probability of being wrong.

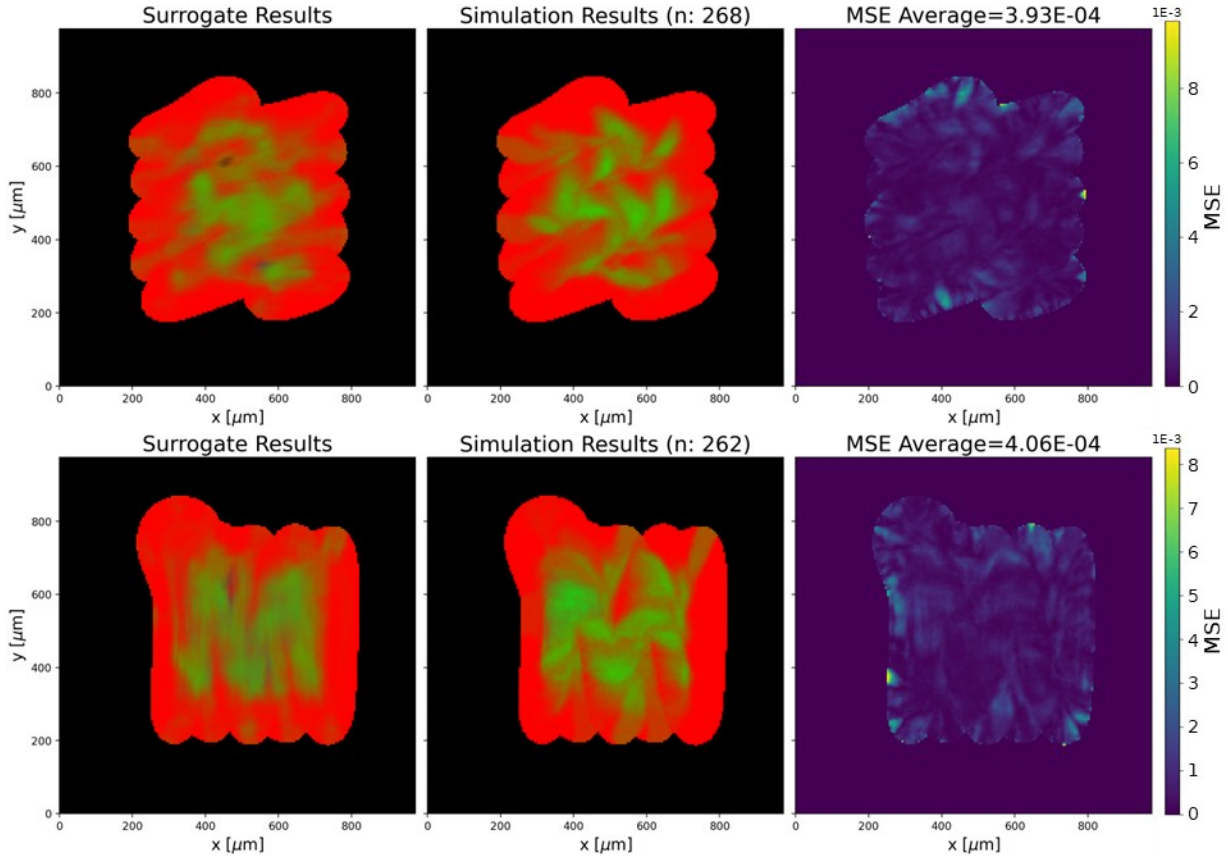

Figure 61 – Top surface of the 4th (top) and 5th (bottom) layers of the P225V72.5 sparse interpolation case, with the upper cutoff for visualization calculated using the 99.5% method. The model overpredicts the maximum size of likely grains by 96μm and 72μ respectively. Top) Red:0-64μm, Green:64-128μm, Blue:128-208μm. Bottom) Red:0-64μm, Green:64-128μm, Blue:128-200μm.

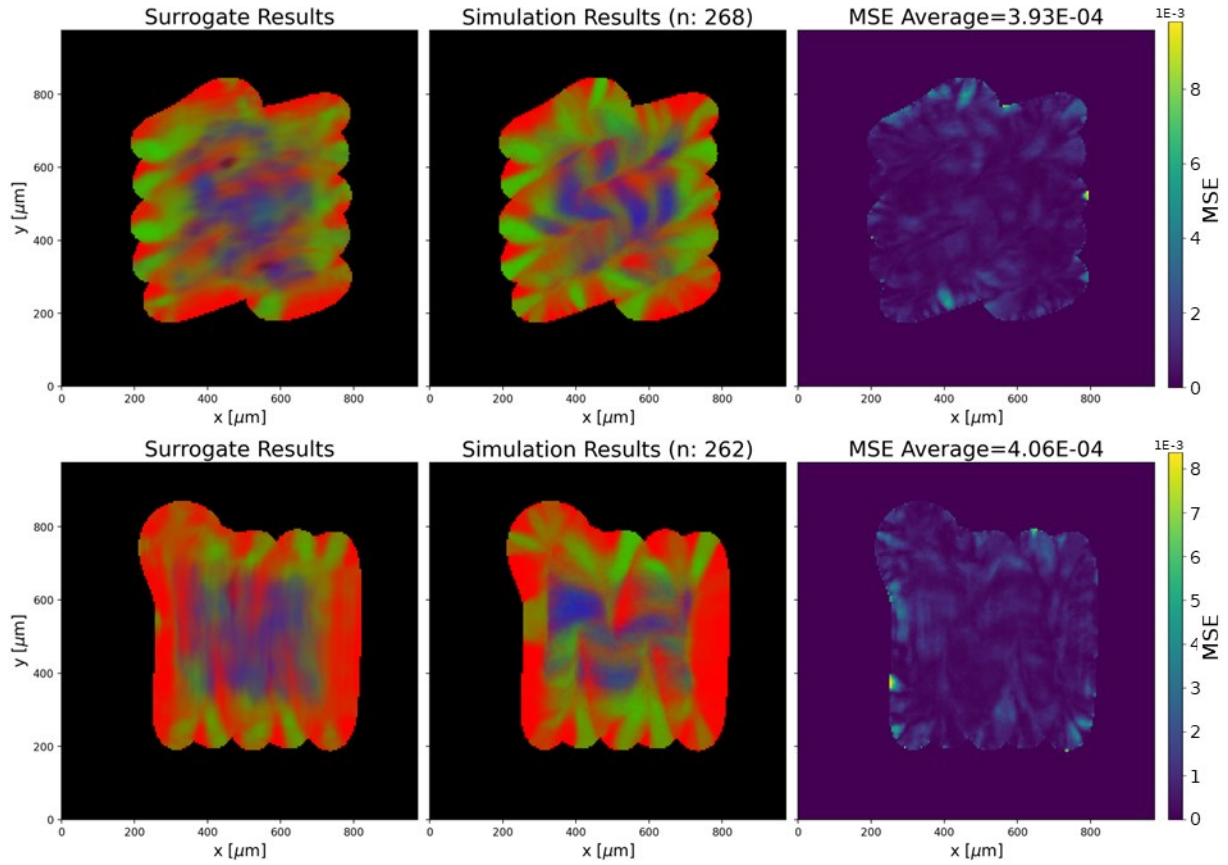

Figure 62 – Top surface of the 4th (top) and 5th (bottom) layers of the P225V72.5 sparse interpolation case, with the upper cutoff for visualization calculated using the 0.5% method. The model overpredicts the maximum size of likely grains by 16μm for both cases. Top) Red:0-40μm, Green:40-80μm, Blue:80-128μm. Bottom) Red:0-48μm, Green:48-96μm, Blue:96-144μm.

Moving on to the full completed 30-layer print in Figure 63, we find that the model continues to overpredict the maximum grain size by 96μm using the 99.5% cutoff. Interestingly, using the 0.5% cutoff in Figure 64, we find that the model's predictions and simulation-based ensemble appear more similar than they did using the 99.5% cutoff. This indicates that much of the probability in the central region of the ensemble statistics belongs to grains less than 32μm smaller than the largest (blue) super bin in the first figure. Also of note is that in both figures, the model's prediction appears to have some semblance of the statistical patterns present in the red to green transition of the ensemble-based results. While these resemblances are rough at best, it is a good sign that the model is picking up on some relationships in the data. However, these results are not particularly accurate, indicating that this level of sparsity should be avoided.

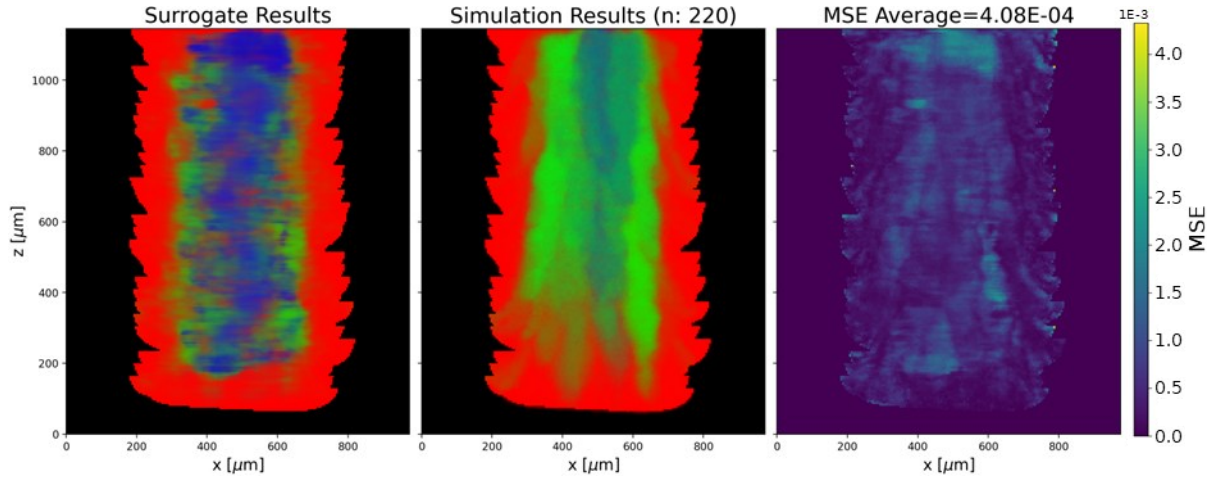

Figure 63 – Cross section of the full 30-layer P225V72.5 sparse interpolation case. The model overpredicts the maximum likely grain size by 96μm when the 99.5% cutoff is used. Red:0-128μm, Green:128-256μm, Blue:256-392μm.

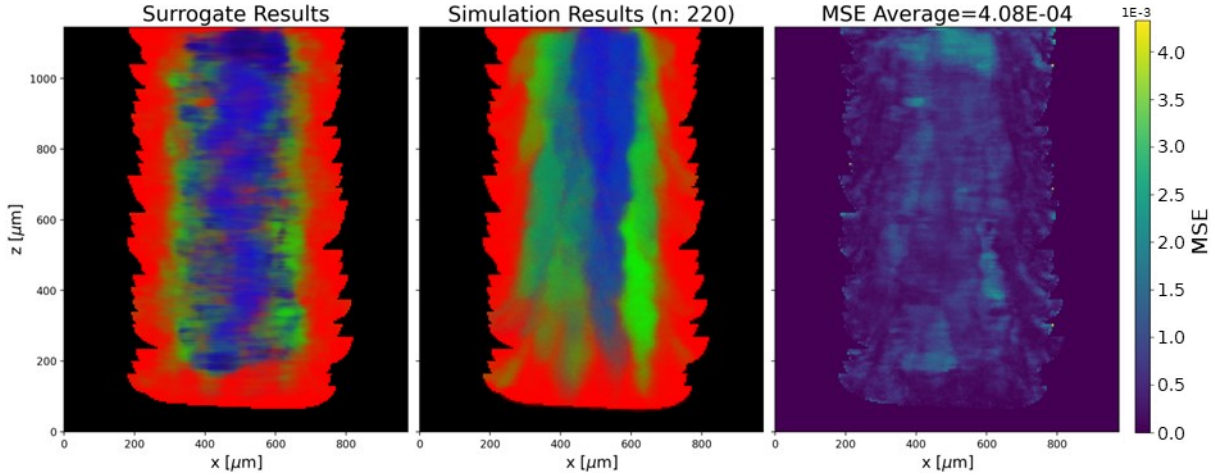

Figure 64 – Cross section of the full 30-layer P225V72.5 sparse interpolation case. Using the 0.5% cutoff results in the model's prediction looking more reasonable, though it still overpredicts the maximum likely grain size by 72μm. Red:0-112μm, Green:112-224μm, Blue:224-352μm.

Notably, despite this prediction significantly overpredicting the maximum likely grain size, the MSE of these results is actually lower than the MSE of the prediction for the same case by the model in Section 4.4.2 which had been trained using 5 and 10-simulation ensembles for all cases. This indicates that having training data for a specific case does not necessarily mean that the model will make more accurate predictions than a model trained on higher quality data. This will be explored more in Section 4.6.

4.5.3.5 Rotation: Cases with powers and velocities in the extrapolation and sparse interpolation regimes, with new angles of rotation for the scan direction between layers

With the rotation scenarios, we target the lack of diversity of laser scan rotation angles included in our dataset. Though many rotation angles are prominent in LPBF literature, the rotation angle of 67 degrees is particularly common. To better focus our exploration of the parameter space when generating training data, we chose to make all but two of the cases in our dataset use the 67° rotation angle. This angle is particularly convenient for the purposes of generating varied data because it does not divide evenly into 360°, resulting in it taking many layers before the pattern repeats. However, as other angles are also used, it is important that a surrogate model be able to make predictions about arbitrary rotation angles. As only two cases with different angles were included in the dataset, the P76V104R90 case with a 90° rotation and the P90V110R45 case with 45° rotations, this was treated as two separate scenarios where each case was individually excluded from the training data. As these angles fall on opposite sides of the 67° standard, both scenarios are in the extrapolation regime.

Starting with the 90-degree case in Figure 65, we find that the model performs quite poorly overall with an average MSE of 9.25E-4, similar to that of the model trained on 5-simulation ensembles when making predictions about the P225V72 case. These results are particularly interesting, as they show the model significantly underpredicting the size of the largest likely grains, with the 99.5% cutoff of the model's prediction being 160µm smaller than that of the ensemble data. This is particularly evident if we switch to using the 0.5% cutoff method for visualization, which results in a large mostly black central region of the ensemble data, indicating that the majority of the probable outcomes for these voxels fall outside of the range of the displayed red, green, and blue super-bins. Interestingly, the lack of any significant blue regions in this plot indicates that the ensemble predicts almost no grains with a diameter of 96-144µm, a bimodal pattern that is not present to such an extent in any other case.

Despite this discrepancy between the model prediction and ensemble statistics, the model qualitatively does an acceptable job at predicting the prevalence of small and medium grain sizes along the outer extremities of the part.

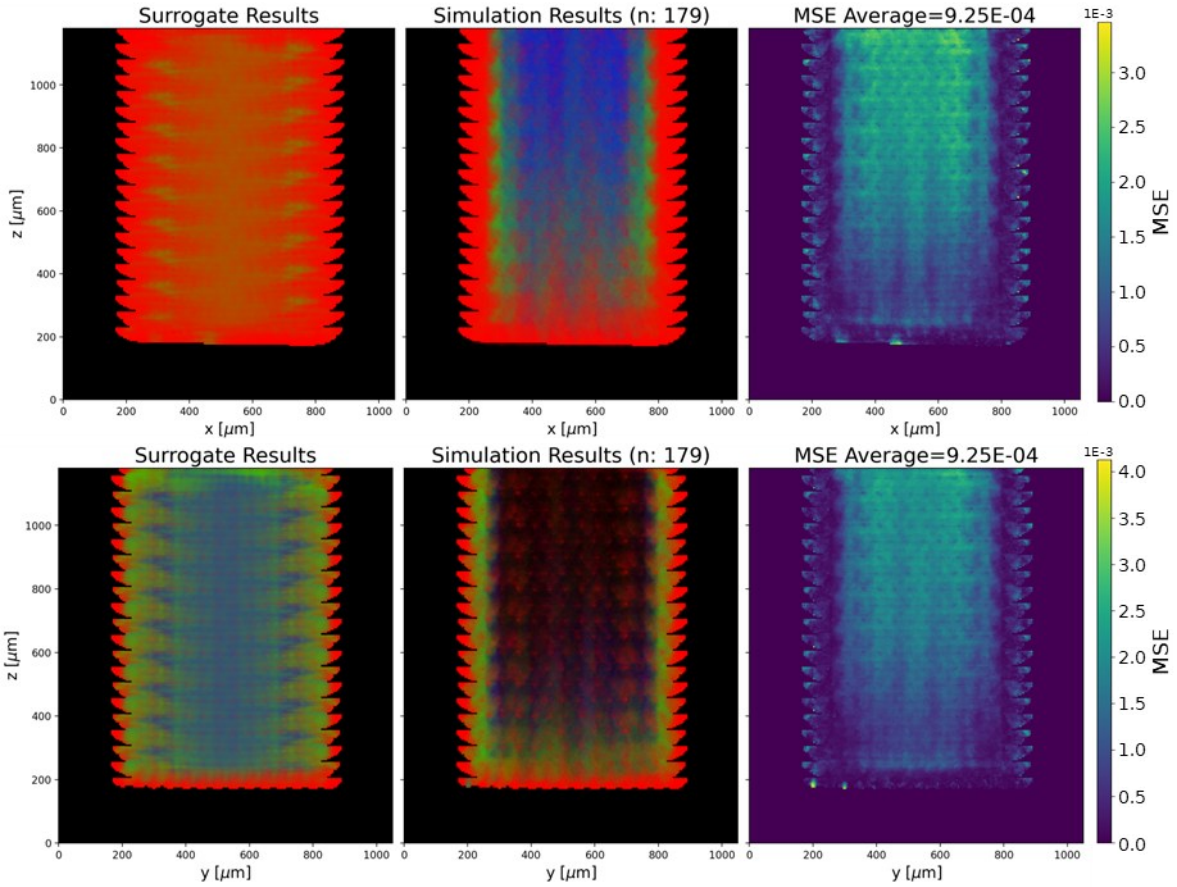

Figure 65 – Cross section of the P76V104R90 90-degree rotation case, using the 99.5% cutoff (top) and the 0.5% cutoff (bottom). The model significantly underpredicts the size of the largest likely grains, which results in a large black area in the ensemble predictions when the 0.5% cutoff is used. Top) Red:0-104μm, Green:104-208μm, Blue:208-320μm. Bottom) Red:0-48μm, Green:48-96μm, Blue:96-144μm.

Examining the top surface of this print in Figure 66, we find that the model does an okay job at predicting some of the statistical patterns. However, the model fails to reproduce some of the patterns that do not lie along either of the two axes and seems to instead be overemphasizing the influence of the previous layer's scanning direction.

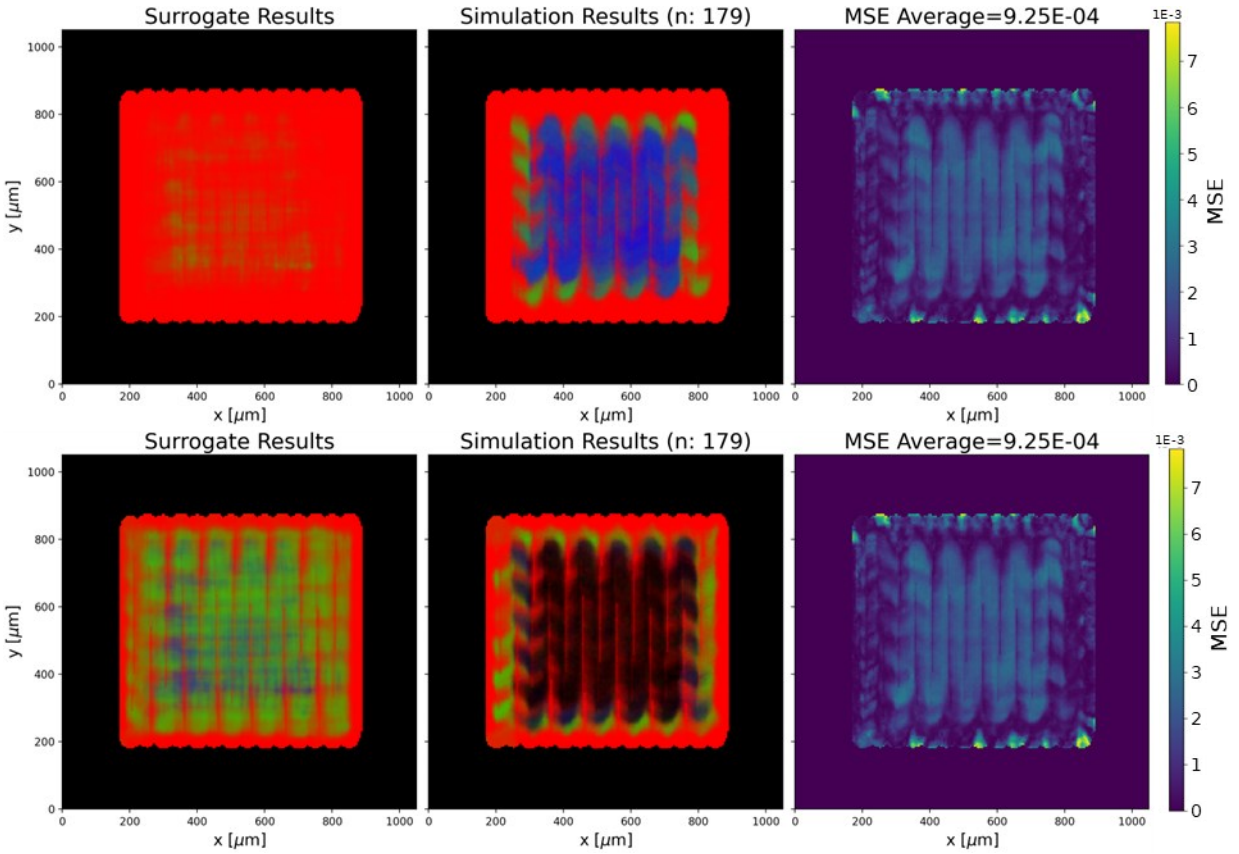

Figure 66 – Top surface of the P76V104R90 90-degree rotation case, using the 99.5% cutoff (top) and the 0.5% cutoff (bottom). The model fails to accurately predict the direction of some of the patterns, seemingly overemphasizing the importance of the previous layer's scan direction. Top) Red:0-104μm, Green:104-208μm, Blue:208-320μm. Bottom) Red:0-48μm, Green:48-96μm, Blue:96-144μm.

Moving on to the 45-degree P90V110R45 scenario, we find that the discrepancies between the model's prediction and the ensemble data is much lower, with an average MSE of just 3.16E-4 for the full 10-layer print. While this is still higher than ideal, in Figure 67 the majority of the error seems to be concentrated to a few clusters, and the largest likely grain size predicted by the model is only 8μm too large. Additionally, inspecting the model's predictions, we see that the results are qualitatively quite similar to the ensemble data, though it fails to predict some of the complexity in the pattern on the top surface.

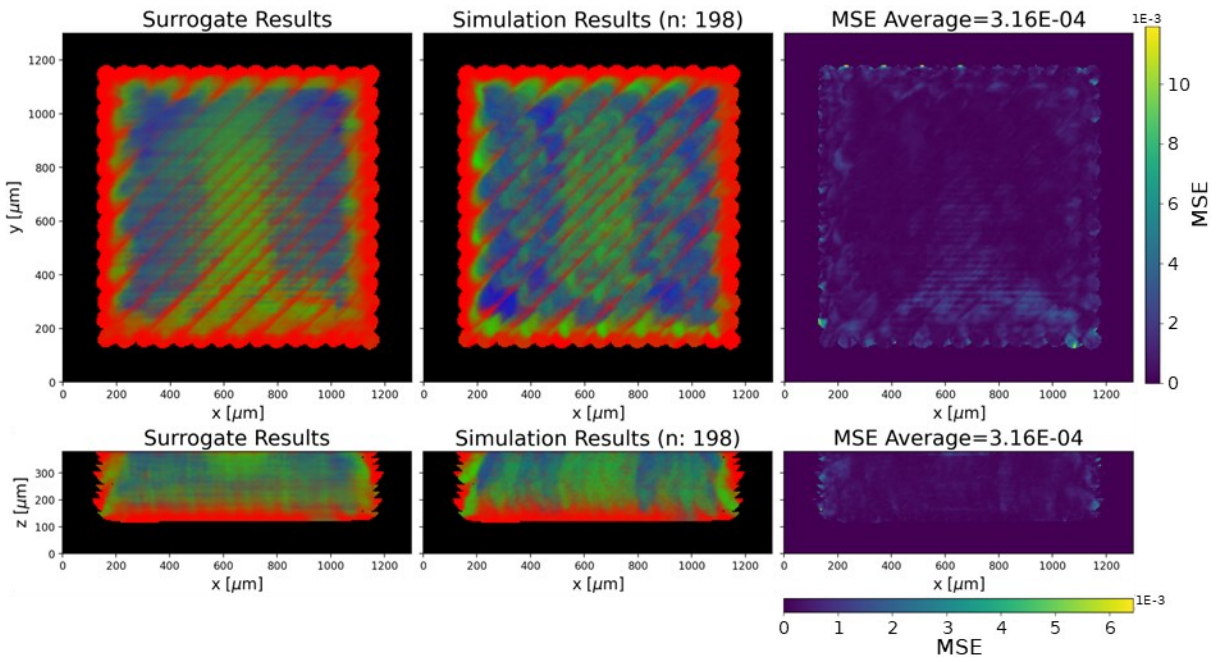

Figure 67 Top surface (top) and cross section (bottom) of the P90V110R45 45-degree rotation case. Red:0-56μm, Green:56-112μm, Blue:112-176μm.

Interestingly, despite being only 10-layers tall, this case does not seem to run into the same issue of overpredicting grain size that has been seen when making predictions about other prints that are less than 1mm tall. While this does occur somewhat at lower layers in this case, peaking with some regions that are 40μm too large in the 6-layer case, the decreased occurrence here is noteworthy. This indicates that the issue the model architecture has been observed to have with intermediate print heights are more complex than previously suspected.

4.5.3.6 Far Extrapolation: A case with a velocity higher than any other

When tasked with making predictions about the P80V130 case without having seen it before, the model performs quite poorly, with an average MSE of 9.79E-4. This can be seen in Figure 68 which shows that the model significantly overpredicts the maximum likely grain size in the interior of the part. To put this difference in numbers, the 99.5% cutoff point for visualizing the model's prediction is 104μm larger than the 99.5% cutoff for the ensemble statistics. Oddly, despite this overprediction of grain size, the model seems to significantly underpredict the occurrence of medium grains around the edges of the part, instead overpredicting the prevalence of smaller grains, with distinct bands of small grains extending across the part. This trend of excess small grains extends to the top surface of the part, shown in Figure 69, where the model predicts almost entirely the smallest grains on the top surface. Here we

see one of the highest magnitude MSE clusters that has been directly observed, with a peak around 1.2E-2 on the edge of the top surface, significantly worse than the average accuracy of 5-simulation ensembles observed in Section 4.4.

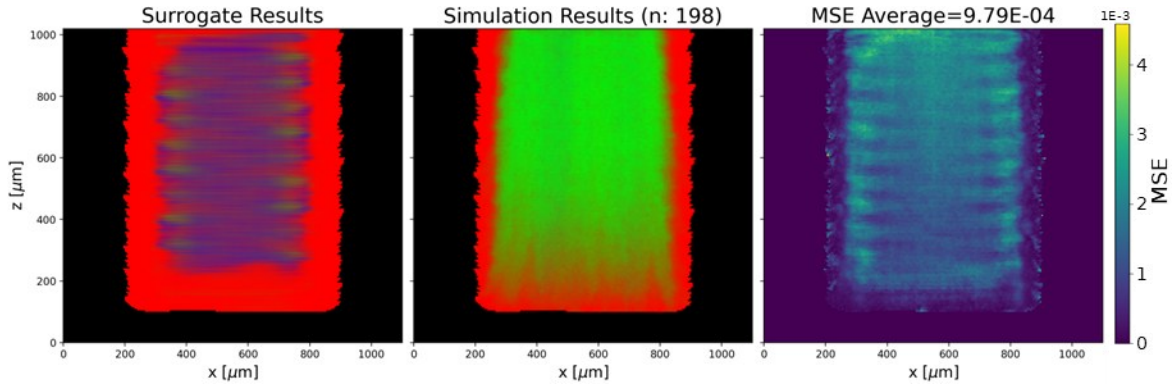

Figure 68 – Cross section of the 60-layer P80V130 (Far Extrapolation) case. The model notably overpredicts the size of likely grains by 104μm, and has significant banding artifacts with a high likelihood of smaller grains. Red:0-120μm, Green:120-240μm, Blue:240-368μm.

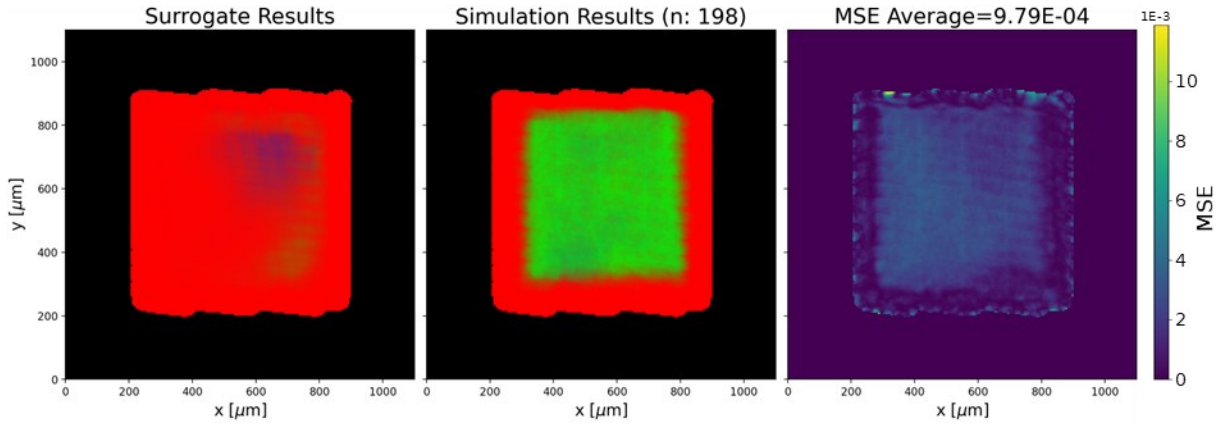

Figure 69 – Top surface of the 60-layer P80V130 (Far Extrapolation) case. The model notably underpredicts the size of the grains along the top surface, with the exception of a small region with very large grains. Red:0-120μm, Green:120-240μm, Blue:240-368μm.

This contrast between the model's prediction and the simulation-based ensembles is slightly more interesting when making predictions about lower intermediate layers for this case. For example, when tasked with making predictions about the microstructure statistics after just 10 of the 60 layers had been printed, the model predicts some regions with reasonable levels of accuracy, and others with such low accuracy that it becomes hard to properly visualize the results. This is illustrated in Figure 70, which shows the results plotted using both the 99.5% cutoff used in the previous figure, alongside the results plotted using a cutoff determined by the last bin in the model's prediction which drops across the 0.5% threshold from one bin to the next. Here we show the top surface, because while the cross section

is also interesting, it does not show the source of the discrepancy. The discrepancy in question is a localized region in the corner of the print with a high likelihood of grains that are 208µm larger than the largest grains according to the 99.5% cutoff of the ensemble-based statistics. Despite this delta being twice the size of that of the 60-layer prediction, the average MSE is actually slightly lower, once again demonstrating the limitations of using the MSE alone to measure the model's accuracy. However, in this case, the visualization alone is sufficient to see that the model is performing poorly, as this behavior is not present in any SPPARKS ensemble.

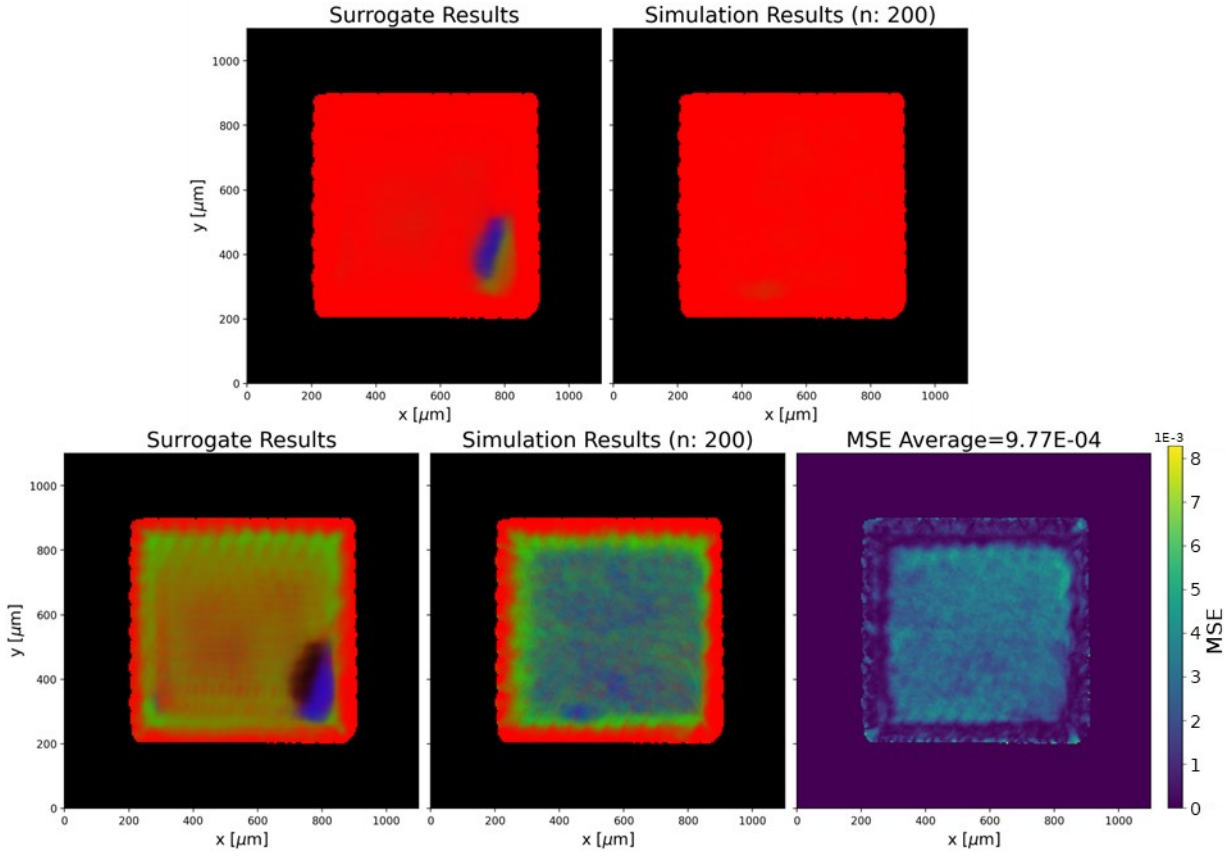

Figure 70 – Top surface of the 10-layer P80V130 (Far Extrapolation) case, plotted using the 99.5% cutoff (top) and the drop below 0.5% cutoff (bottom). Using the 99.5% cutoff results in almost the entire part being binned in the "small" grain size red super-bin due to the predicted likelihood of very large grains in the corner of the part. Using the less refined cutoff method of finding the last bin to cross below the 0.5% threshold shows a much clearer picture, though it results in a region with "0%" probability of anything. Top) Red:0-104µm, Green:104-208µm, Blue:208-320µm. Bottom) Red:0-40µm, Green:40-80µm, Blue:80-136µm.

While the results shown here illustrate certain trends in the model's predictions, it is important to remember that the stochastic nature of the model initialization and training process means that different iterations of the model may have different behaviors. Due to corruption and loss of the model which was trained on this scenario, it was necessary to train a second iteration of the model for testing

this scenario, so that the results could be examined in full. It is from this second version of the model that the results which have already been shown came. However, the initial plots from the first iteration of this model were preserved, which gives us extra insight into the model's performance for this scenario. These initial predictions, which use the last bin of the model's prediction which drops across the 0.5% threshold to determine the cutoff point for plotting can be seen in Figure 71. Interestingly, this version of the model appears to underpredict the maximum size of likely grains, with a dramatic underprediction in the likelihood of such grains. As a result, the ensemble-based data has a large region where the visualization indicates 0% likelihood of any grain size, due to the grains in this region being larger than the cutoff point determined using the model's predictions. This contrasts quite starkly to the previously presented results, from the second version of this model, which showed a significant overprediction of the maximum size of likely grains. For a more direct comparison, the results of the second iteration of the model, plotted using the same cutoff point scheme, can be seen in Figure 72. While both versions of the model produce the same banding artifacts across the central regions of the part, the first iteration of the model does a much better job at predicting the grain size distributions around the extremities of the part. Despite this improved prediction of the patterns, the first iteration of the model performed much worse overall, with an average MSE of 1.43E-3, which is 1.5 times higher than the MSE of the second iteration of the model.

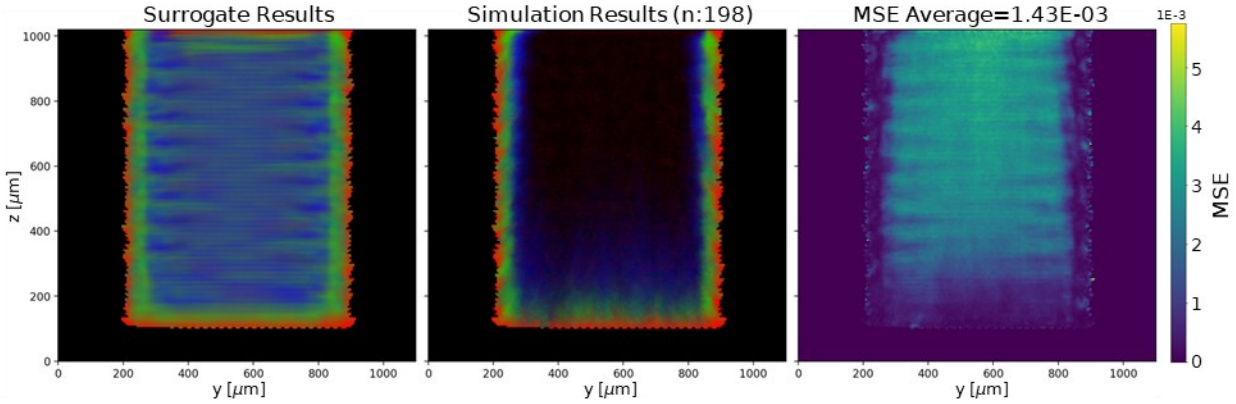

Figure 71 – Cross-section of the 60-layer P80V130 (Far Extrapolation) case as predicted by the first model created for testing this scenario, plotted using the 0.5% cutoff. The model significantly underpredicts the size of likely grains, resulting in the ensemble-based statistics having a large "0%" probability region due to the grains being larger than the cutoff. The magnitude of this under-prediction is unknown. Red:0-32μm, Green:32-64μm, Blue:64-120μm.

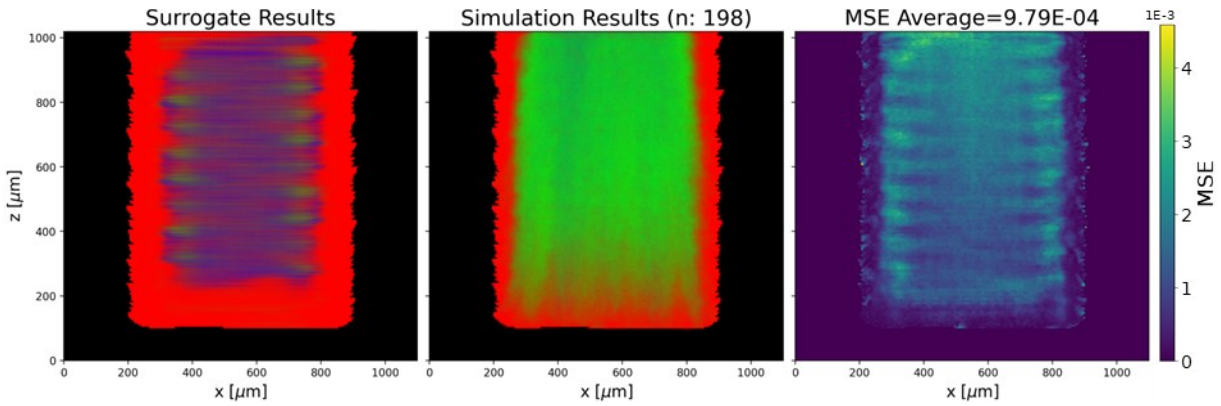

Figure 72 – Cross-section of the 60-layer P80V130 (Far Extrapolation) case as predicted by the second version of the model created for this scenario, plotted using the same cutoff scheme as was used in the previous figure with the first model. Red:0-112μm, Green:112-224μm, Blue:224-336μm.

These differences in the two versions of the model trained on the same data, when making predictions about the same case illustrate the difficulties in definitively quantifying the accuracy of such a model when making predictions. With that said, this case was intentionally chosen as an outlier in the dataset, with a higher laser velocity and many more thin layers than others. These results illustrate the importance of adequately covering the LPBF process-parameter space of interest when creating training data for the model.

4.5.4 Conclusion

Overall, these results show that the model often struggles with making predictions about new situations, particularly in extrapolation and sparse data situations. However, the ability of the model to make reasonable predictions about large regions of many of the cases included here is promising. These results highlight some of the pitfalls that the model architecture currently has, and provide insight about how some of them might be avoided. Particularly these results indicate the need for a more diverse training dataset, and indicate that intermediate layers of prints need further consideration.

4.6 Supplementing the Training Data with Small Ensembles

In addition to the original questions around the influence of ensemble size on model accuracy discussed in Section 4.2, we further wondered if some of the deficiencies in the model's predictive power, caused by gaps in the parameter space covered by the training data, could be improved by augmenting the training data with small ensembles that cover those gaps. Much like with the original hypothesis in Section 4.2, it was reasoned that the model should be able to make predictions that are

more accurate than these small ensembles, possibly to a greater degree due to the other larger ensembles also included in the training data.

To test this, three new versions of the model were trained from scratch using the largest available ensembles for all cases but the P225V72.5 case – the sparse data interpolation case from the previous section – from which only 5, 10, and 25 simulation ensembles were included. The three new models were then used to make predictions about the P225V72.5 case.

4.6.1 Discussion

As noted in with the sparse interpolation predictive power results in Section 4.5.3.4, the results of the predictive power analysis and the ensemble size analysis in Section 4.4, point to a complicated relationship between training data and model accuracy. Specifically, we showed that the model makes more accurate predictions about this case when data for it is not included in the training set, than it does when the entire training set uses small ensembles. This can be seen in Figure 73, which shows the model's prediction for this case when not trained on it, alongside the prediction when all training data uses 5-simulation ensembles. While this outcome is somewhat unintuitive, this behavior makes some sense, as small sampling sizes can significantly skew distributions, and poor-quality data can make it difficult to learn relationships.

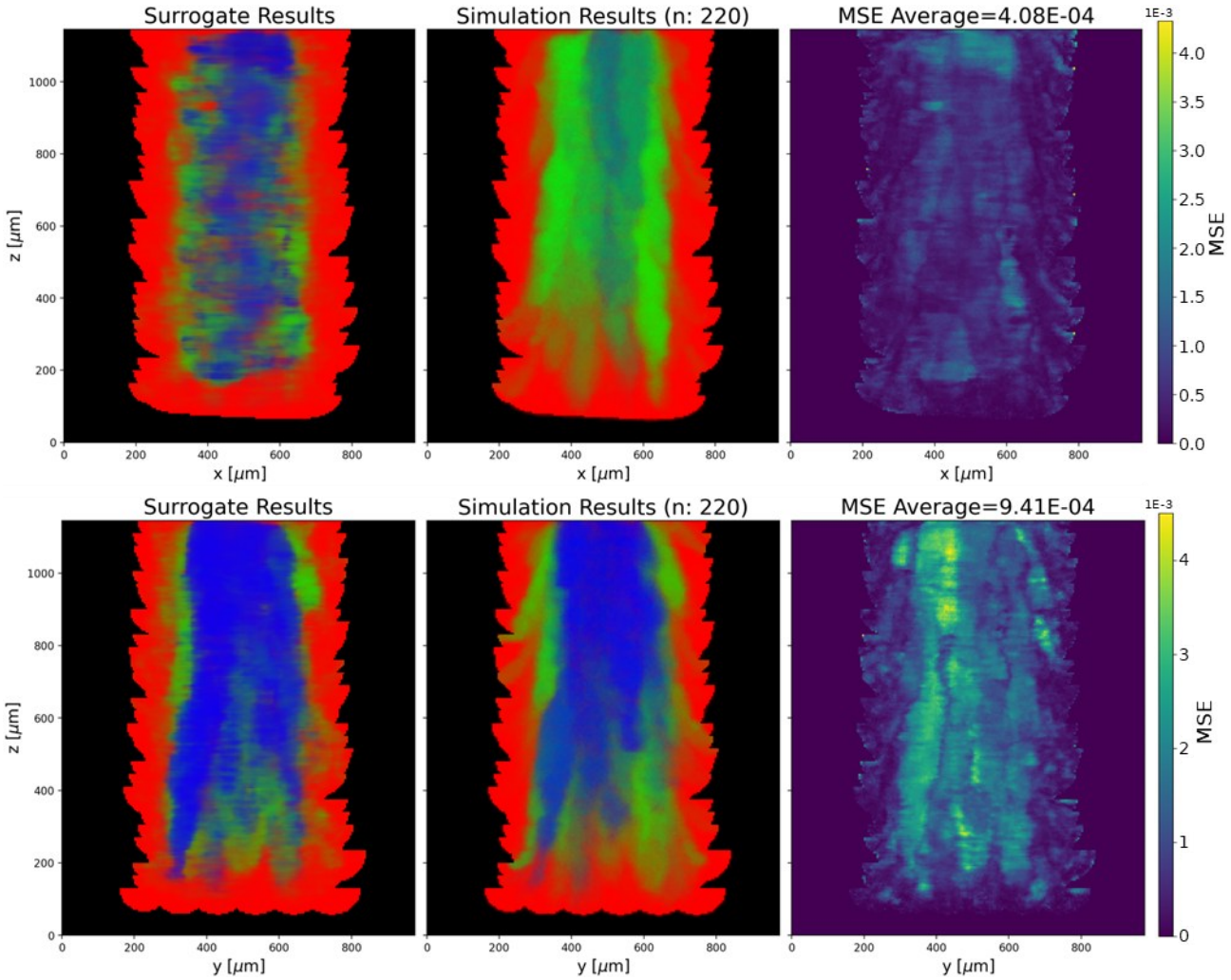

Figure 73 – Cross section showing the model's prediction for the P225V72.5 case when not trained on it, i.e. an ensemble size of zero (top) and when all training data is 5-simulation ensembles (bottom). Despite significantly overpredicting the size of the largest likely grains, the model has a much better overall accuracy in the predictive case than in the 5-ensemble dataset case. Top) Red:0-128μm, Green:128-256μm, Blue:256-392μm. Bottom) Red:0-96μm, Green:96-192μm, Blue:192-296μm.

4.6.2 Results

Starting with the case where only 5 simulations are used in the training data for this test case, shown in Figure 74, we can see that much like in Sections 4.2 and 4.4 the model's predictions when trained on this small ensemble are more accurate than the ensemble itself. Additionally, we see that the model's ability to predict the statistical structures in the microstructure are greatly improved over the model where no data from this case was included in the training data. However, this version of the model has a lower accuracy for this case than both the model trained with only 5-simulation ensembles and the model trained without any data for this case. This behavior may indicate that this specific 5-simulation ensemble, which is the same one used in Section 4.4, may be decreasing the overall accuracy

of the model for this case. It is also possible that the behavior observed here is due in part to particularly lucky or unlucky instances of the random model initialization or random splitting of the data into test and training splits. However, as such large discrepancies between models have not been observed in other tests, and all scenarios being discussed are the same case, it seems much more likely that it is due to the data.

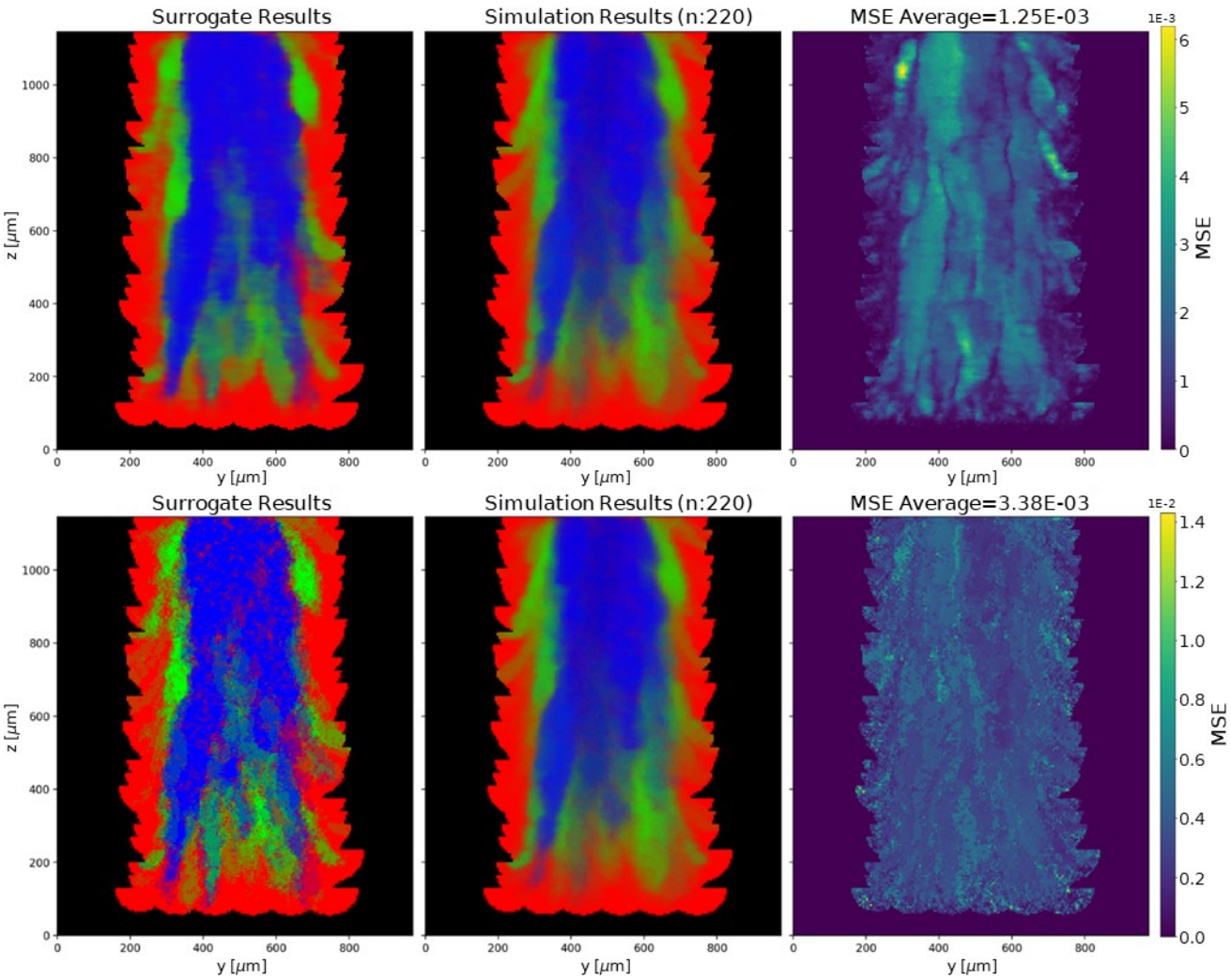

Figure 74 – Cross section of the P225V72.5 case, showing the model's prediction when a 5-simulation ensemble for this case is added to the training data (top) and the 5-simulation ensemble itself (bottom). The model's prediction has a lower MSE than the 5-simulation ensemble in the training data, when compared to the full ensemble for this case. Visualized using the 0.5% cutoff. Top) Red:0-88μm, Green:88-176μm, Blue:176-272μm. Bottom) Red:0-64μm, Green:64-128μm, Blue:128-200μm.

While this prediction is significantly worse than the prediction by the model which did not include any training data for this case, the continued improvement of the model's accuracy over the training data does at least indicate that including small ensembles in the model's training data will result in better predictions than relying on the small ensembles alone.

Moving on to the scenario where the 10-simulation ensemble has been added to the training data in Figure 75, we find that the accuracy has significantly improved with an MSE lower than the only 5-simulation ensembles scenario, but still higher than in the no-data scenario. Qualitatively, we find that the model's prediction of the statistical patterns has improved, with large regions that are nearly identical.

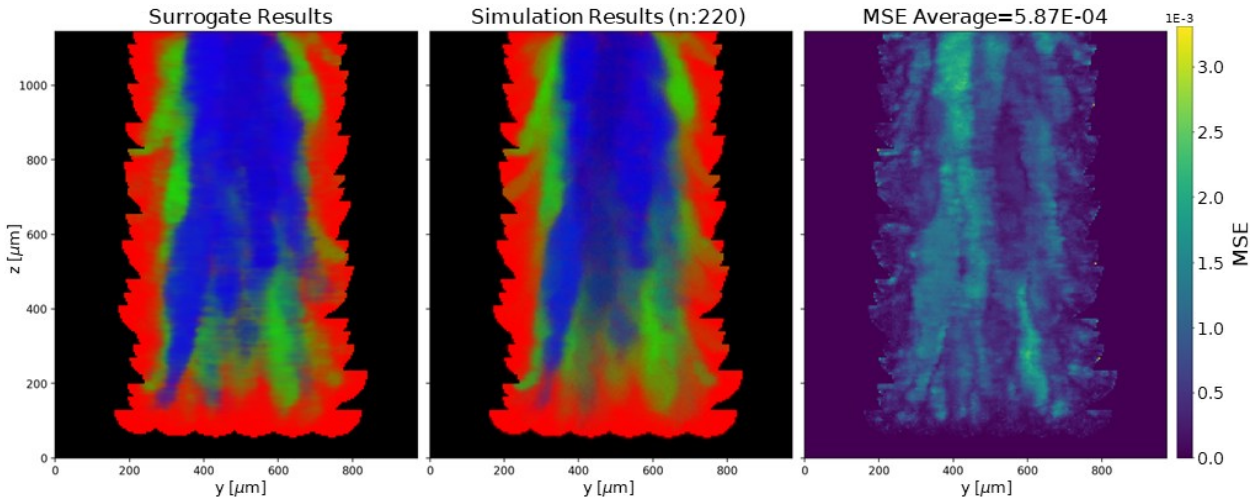

Figure 75 Cross section of the P225V72.5 case, showing the model's prediction when a 10-simulation ensemble for this case is added to the training data, visualized using the 0.5% cutoff. Red:0-88μm, Green:88-176μm, Blue:176-264μm.

Finally, adding a 25-simulation ensemble to the training data in Figure 76 qualitatively results in nearly perfect prediction of the statistical patterns, and an MSE that is lower than that of the prediction in the predictive power scenario. While the MSE is still not as good as if we had trained the model on a larger ensemble for this case, it's clear that amending the model's training data by adding a 25-simulation ensemble improved its accuracy.

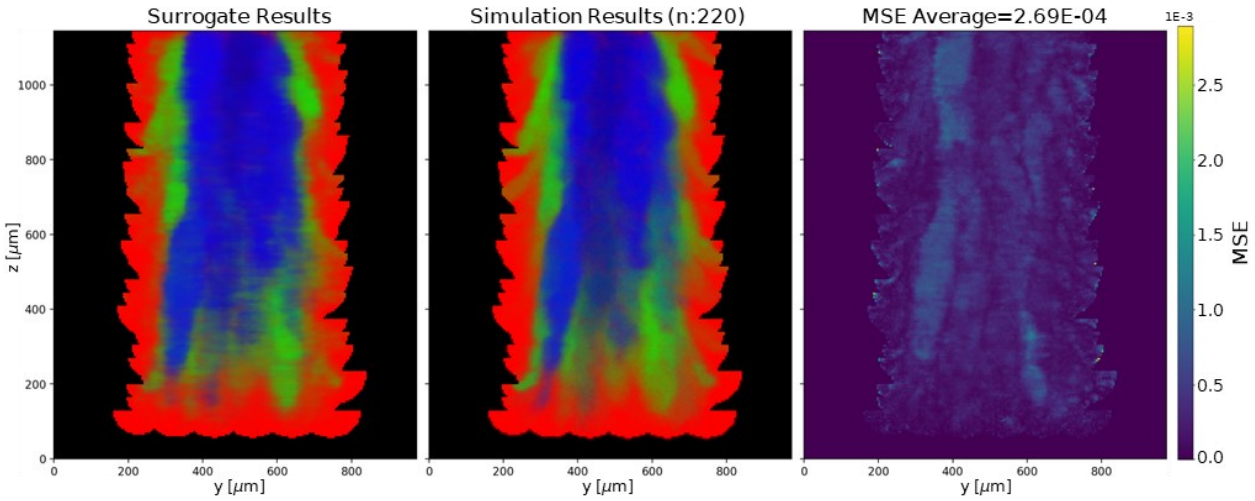

Figure 76 – Plots showing the accuracy of the model when a 25-simulation ensemble is used as training data for this test case.

From these results we can conclude that including some data is better than no data for the purposes of predicting the statistical structures. However, we have also shown that using particularly small ensembles can lead to cases where the model's accuracy decreases due to the low accuracy of the training data. This gets significantly less likely as the ensemble size increases, but a trade-off must be made between parameter space coverage and ensemble accuracy. When adding data in regions of the process space where it is possible that the model has sufficient training data, it is likely necessary to use a larger ensemble to ensure that there is no regression in accuracy due to low accuracy of small ensembles. However, as we have shown here, even much smaller ensembles (i.e. 25) than those used in the rest of our training data (i.e. 100-200) may be sufficient to avoid this.

4.7 Conclusions

Here we have shown that our surrogate microstructure model makes accurate predictions about grain size distributions produced by the LPBF process. More specifically we have shown that the model learned to average out the stochasticity in a way that outperforms the accuracy of the SPPARKS ensembles used to train and test it. Additionally, we showed that the model achieves a reasonable level of accuracy when making predictions for some LPBF process parameters not included in the training data. Finally, we showed that the averaging behavior of the model lends itself to training the model on smaller ensembles than those used to train the majority of the models analyzed in this work. We suspect that this behavior could be leveraged in future versions of the model to fine tune it for regions of the LPBF parameter space not included in the existing dataset.

4.7.1 A Discussion on The Importance of Accuracy

Before we move on, it is worth addressing an interesting question that arises from the behavior observed here: how accurate is accurate enough? While in real-world applications, the model is likely to be used for making predictions for cases that it has not seen before and for which we do not have data against which we can measure its accuracy; here we have shown that we can control the baseline accuracy of the model by adjusting the size of the training ensembles. This is a particularly interesting question because of how different this surrogate model is from both conventional simulations and conventional microstructure representations.

In conventional simulations, the accuracy of a model is determined by some combination of the accuracy of the inputs and parameters, how well the model captures the underlying phenomenon, and the influence of some simulation specific parameters that control numerical accuracy; often in a way that inversely affects the speed of the model. While there are some analogues to the surrogate model, for instance the accuracy of the thermal model can be changed to affect the accuracy of the surrogate model's inputs, the current iteration of the model does not allow for any decisions to be made at run-time. Instead, the decisions made about the one-time cost of creating the training data have the most significant influence we have observed on the baseline accuracy of the model, with no influence on the speed of the trained model. The fact that this is a one-time cost makes it tempting to prioritize large ensembles so that there is certainty in the model's accuracy. However, there is a tradeoff to be made here between the size of the training ensembles and the number of cases included in the training data. As observed in Section 4.5, and consistent with the expected behavior of a data-driven model, the coverage of the parameter space influences the accuracy of the model when making predictions about new cases. While increasing the size of the ensembles used for training increases our confidence in the baseline accuracy, for a fixed computational cost it also decreases the coverage of the parameter space and by extension the predictive power of the model. These considerations make the desired accuracy of the model a particularly important question.

Additionally, the desired accuracy of the model is a particularly interesting question because the model is not trying to predict the individual microstructures produced by conventional models or experimental imaging, but instead trying to predict the statistics one would get from running the same simulation or experiment many times. This is important context for this question because, while we have

shown the model can meet or exceed the accuracy of large training ensembles, it would be unusual to run the same simulation or experiment hundreds of times to get the types of statistics the surrogate model predicts. While this does mean that the actual applicability of the statistical representation used here is unknown, as they are unconventional, it also means that the accuracy achieved by training on even very small ensembles is as good or better than generally achieved using conventional methods. For instance, in Section 4.4 we showed that the model trained on ensembles of just 5 simulations was as accurate as getting the same statistics by running a simulation of every training case 25 times. While we showed that this results in poor reproduction of many of the statistical patterns, and does not necessarily hold when making predictions about new cases, it demonstrates that even a very inaccurate model is more statistically reliable than all but the most thorough uses of conventional methods.

This does raise the question of how accurately the model can predict the real-world statistics of the LPBF process, as every layer of the model creation requires approximations, and none of those layers have been calibrated to the same extent that our model has been trained. However, the reliance of the underlying models on physical relationships is more reliable than the data-driven approach taken by our model, and analyzing the statistical significance of the SPPARKS model is not within the scope of this research.

Chapter 5 Understanding The Surrogate Model

5.1 Introduction

While the findings of the previous chapters indicate that the model framework can create a surrogate LPBF microstructure model that can quickly and accurately predict microstructure statistics, the framework relies on several assumptions which have not been directly tested. In this chapter, we will work to better understand the accuracy and implications of two such assumptions by exploring the relationships between the surrogate model and the thermal characteristics.

First, we will work to understand if all the thermal characteristics selected to represent the LPBF process to the surrogate model are necessary, and determine to what extent the model is learning meaningful relationships. Second, we will explore how well the model framework creates the diversity of data needed to train a surrogate model that can generalize to new situations.

5.2 Measuring the Importance of the Selected Thermal Characteristics

The thermal characteristics chosen to represent the thermal history of the LPBF process to the surrogate model were selected based on known relationships between thermal processes and microstructure formation. However, these thermal characteristics, and the relationships used to choose them, do not capture the full complexity of the solidification and microstructure formation process. Instead, the chosen thermal characteristics are a best guess at the minimal amount of information needed to accurately predict the microstructure statistics that will result from a given LPBF process. Though it is not definitive, the previously demonstrated ability of the model to learn from the training data and make reasonably accurate predictions indicates that the chosen thermal characteristics are at least somewhat sufficient for modeling the LPBF process. Here we will work to better understand the relationships the model uses to make predictions and determine the necessity of each of the thermal characteristics for making accurate predictions about the resulting microstructures.

If the model relies on all nine of the thermal characteristics to some extent when making predictions, it would indicate that the selected thermal characteristics are a useful representation of the LPBF process, though it would not rule out the potential usefulness of other thermal characteristics that

were not included. Conversely, if any of the thermal characteristics are not consistently used by the model, it would indicate that the thermal characteristics in question are not needed to represent the process. In such a case, it may then be possible to reduce the number of thermal characteristics calculated by the thermal model, and simplify the surrogate microstructure model, accelerating both model training and inference. However, further analysis would be necessary, as it may also be possible that the model relies on certain thermal characteristics only some of the time. In addition to the potential reduction in the computational cost of the model, the insight this analysis provides into the model will allow us to determine if the model is consistent with known relationships between thermal processes and microstructures. This is particularly important given the model's intended use in engineering optimization applications, where having some confidence that the model is using reasonable relationships to make predictions is valuable.

5.2.1 Method

At its core, this analysis is a machine learning interpretability analysis, as we wish to examine the influences of the model's inputs on its predictions. However, due to differences between the surrogate model and the neural networks for which most machine learning interpretability methods have been designed, existing methods are not directly applicable. Despite this, existing machine learning interpretability methods provide a useful starting point for our analysis.

One of the most straightforward classes of methods for machine learning interpretability of spatial data is the saliency map, examples of which can be seen in Figure 77. The most basic method for generating saliency maps is to take the derivative of the output of interest with respect to each value in the input [160]. This provides insight into which values of the input would affect a given output from the model the most if they were to change, in effect highlighting which regions of the input are the most important for a given model prediction. This works particularly well for image classification and language models, where the patterns that emerge are easily understood by humans; however, the surrogate model has two key differences from such models. First, unlike with image classification and language models, where only one model output is of interest at a time[1], all 51 of the surrogate model's predictions

[1] Image classification and large language models typically also produce a vector of "probabilities" like the surrogate model, but they typically only return which value is considered the most likely.

are of equal importance. Second, the true relationships between the thermal characteristics and microstructure statistics are not already known, so this method of highlighting regions of importance for a given model prediction is less intuitive.

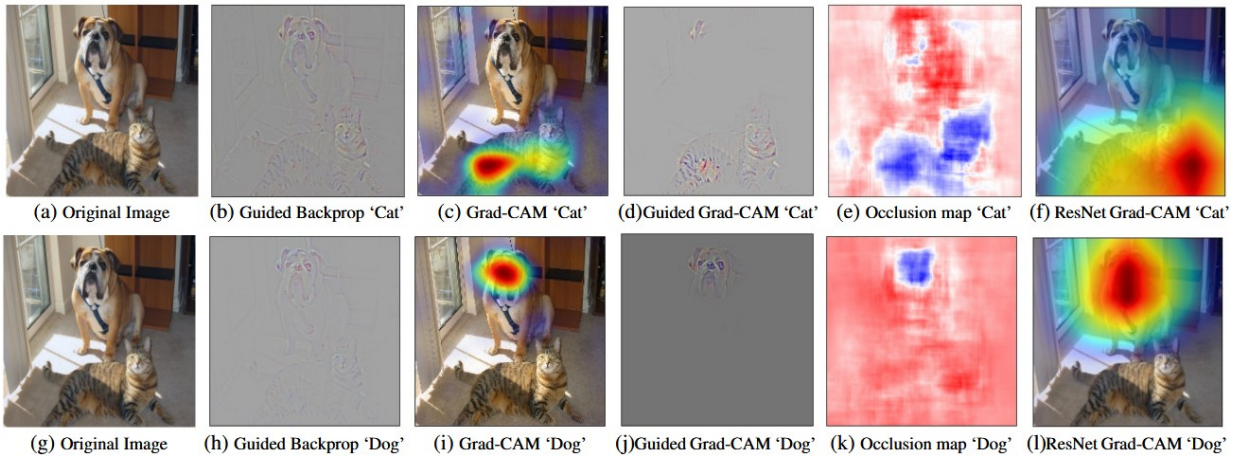

Figure 77 – Illustrations from Selvaraju et al [161] depicting saliency maps generated using several methods, reprinted with permission from Springer Nature.

As we are interested in the overall importance of each of the thermal characteristics, rather than the specific relationships used in each prediction by the model, we can overcome these differences by generating a saliency map for the statistical behavior of the model. To do this, we start by expanding the most basic method for generating saliency maps – the derivative of the output of interest – to cover all of the values output by the model, giving us the Jacobian matrix – the derivative of all values in the output with respect to all values in the input. As we are not interested in the influence of the thermal characteristics on each of the grain size bins individually, but rather as a whole, we then take the mean and standard deviation of these derivatives across the 51 output values. Additionally, because we are not interested in the influence of the thermal characteristics on any given surrogate prediction – as we do not know the true relationships – we calculate these statistics across many predictions. The result, which is an array with the same size as the model's input window, is a statistical representation of the influence of each of the 9 thermal characteristics of each voxel of the model's input window on the model's predictions across the entire dataset.

Because the Jacobians have both positive and negative values due to the input values both increasing and decreasing the probabilities of each grain size, the means of the Jacobians are not necessarily representative of the average influences. Instead, it is necessary to consider both the means

and the variances of the Jacobians to determine how frequently the influences of the thermal characteristics are near zero. To reduce the computational cost of calculating the variance, Welford's method (Equation (7)) was used, which allows for the calculation of both the mean and variance in a single pass over the data. While this does reduce the accuracy of the variance due to compounding numerical precision error, the high computational cost of the Jacobian calculations made this a worthwhile tradeoff [162,163].

$$s_N^2 = \frac{(x_N - \bar{x}_N)(x_N - \bar{x}_{N-1})}{N-1} \qquad (7)$$

s_N^2 is the variance being measured, as of sample N, x_N is the sample being used to update the variance, and $\bar{x}_N$ and $\bar{x}_{N-1}$ are the current and prior means of all samples measured, respectively. With this data, we have a representation of the influence of each voxel of the input, for each of the nine thermal characteristics, on the model's predictions across a wide variety of situations. However, two problems remain which must be addressed to make the data useful. First is that the Jacobian is an indirect measure of importance, in that it indicates how much the output would change if a given value of the input were to change, with no consideration for the likelihood of that change. Say for example that a given value in the input was fixed across all samples while training. This could potentially result in a model that is mathematically highly sensitive to changes in that value simply because the training algorithm had little guidance on how to treat changes in that value. While this is a contrived example, it demonstrates the challenges of using derivatives as a measure of importance. The second problem is that, unlike with image data, where the 3-channels have the same bounds and similar behavior across all possible images, the nine thermal characteristics used to represent the LPBF process have different behaviors and no prescribed bounds. As a result, the magnitudes of the Jacobians cannot be directly compared across the nine thermal characteristics.

To account for both of these challenges, we can scale the resulting Jacobian data according to the values found in the inputs. Because the Jacobians are a measure of how much the output would change for a given change in the inputs, we can scale the Jacobians by the variances of the thermal characteristics, which represent how much each of the values changes across the dataset. The result is a representation of the importance of each of the nine thermal characteristics that accounts for how much variation occurs within each of the thermal characteristics.

5.2.2 Intermediate Results

While the intermediate results – the raw statistics of the thermal characteristics and Jacobians – do not provide the full picture of the relative influences on the model's predictions, they can provide some useful insights. Because there are 9 thermal characteristics, each with 4 intermediate results (the means and variances of both the Thermal Characteristics and the Jacobians), here we only show a subset of the results that demonstrate interesting features.

Thermal Characteristic Statistics

Starting with the variance of the melt count across the entire dataset, visualized in Figure 78 as a volumetric rendering of the input window, and the corresponding cross section. Here we can see the statistical variance of the melt count of each voxel of the input window, measured across all data in the dataset. Because the cross sections reveal more detail, the remainder of the results will be shown as cross sections. While this does result in the loss of some information, the centermost voxel is the voxel about which the model is making predictions, making the central region of these results generally the most interesting. In this figure, we see that there is a slightly lower variation of the melt count along the central lateral plane of the input window, with the lowest variance near the center, though all values are in the same order of magnitude.

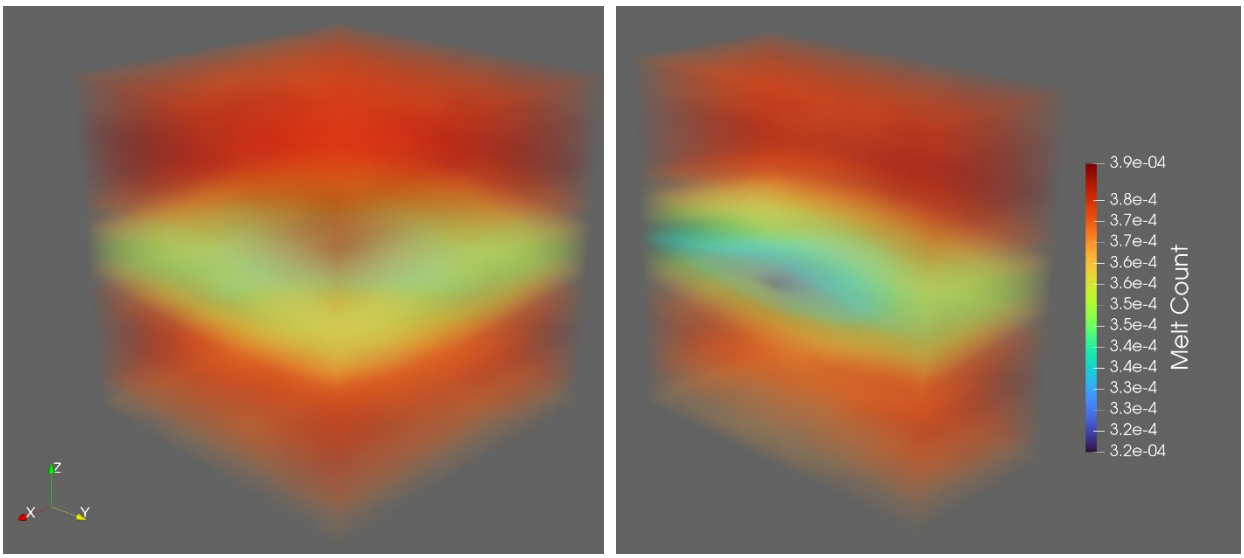

Figure 78 – A volumetric render of the deviation of the melt count for every voxel in the input window, measured across all 77 million input windows in the dataset. Left) The whole input window. Right) The cross-section of the input window, which reveals more information.

Looking at the variance of the temperature gradient in the z-direction – in the build direction – in Figure 79 we see a case where the values are directionally biased, with a somewhat higher deviation below the central plane.

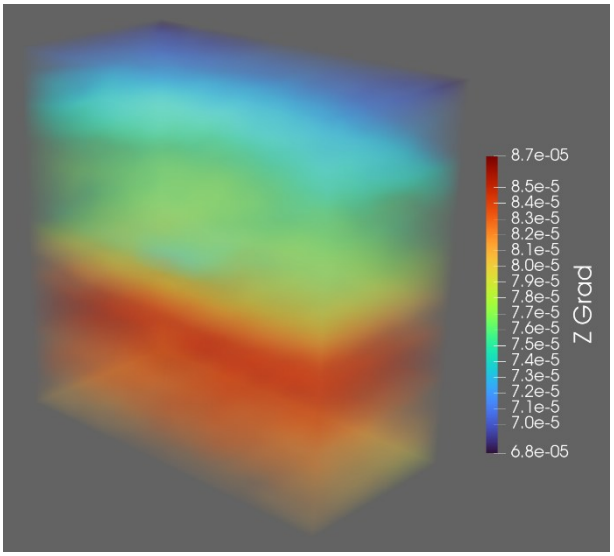

Figure 79 – Deviation of the temperature gradient in the z-direction.

In addition to the regular patterns visible with some of the characteristics, in Figure 80 we see an example of a characteristic where the deviation is much less regular: the time spent in the nucleation temperature range.

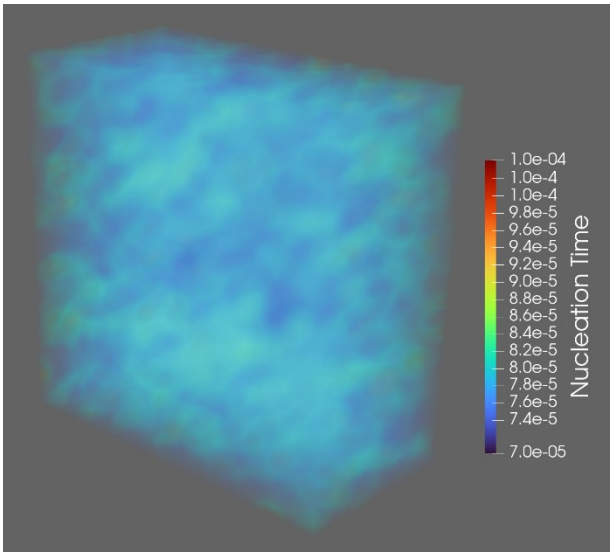

Figure 80 – Deviation of the time spent above the nucleation temperature.

Jacobian Statistics

Next, we will consider the statistics of the Jacobians, starting with the mean Jacobian for the time spent above the nucleation temperature in Figure 81, which has a distinct pattern in its values when cut along the x-z plane. While, for the most part, the means of the Jacobians were not found to be particularly revealing on their own, this case illustrates several patterns which can exist particularly well.

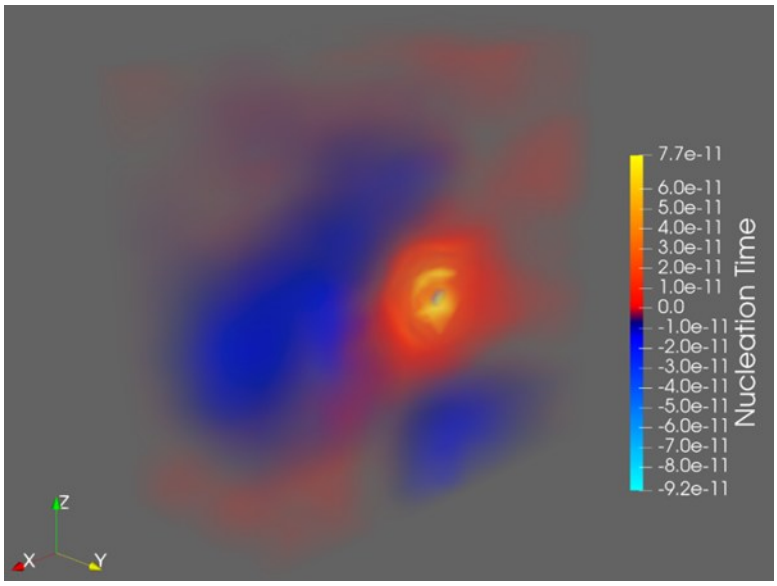

Figure 81 – The mean Jacobian of the surrogate model's predictions with respect to the nucleation time, averaged across all grain size bins and model inputs.

Here we can see, starting at the centermost voxel of the input window – which corresponds to the voxel for which the model is predicting the microstructure statistics – a particularly negative average Jacobian. This is likely indicative of a strong bias in the underlying relationships the model has learned, as the centermost point cannot be biased by the directionality of the data. Specifically, it appears to indicate that for at least some of the grain size bins in the output distributions, the amount of time spent in the nucleation regime is inversely proportional to the probability of those bins occurring, to a significant enough degree that it cancels out any positive correlations. Because these values are averaged across all the bins of the output probabilities, it is impossible to say exactly what grain sizes this strong inverse correlation belongs to without the significant computational costs of re-calculating the Jacobians. While it ultimately fell outside the scope of this work, such an analysis could reveal interesting relationships that are broadly applicable, beyond this surrogate model, between these thermal characteristics and grain size distributions which may not be readily apparent using more conventional methods.

Moving out from this central point, we find that the correlation quickly becomes positive to a similar degree, indicating the opposite, that there are some grain size bins that have increased probabilities when close neighbors spend elevated times in the nucleation temperature range. This is likely, once again, a bias in the underlying relationship the model has learned, as the pattern appears to be largely symmetrical, indicating a lack of directionality.

Moving out from the center once more, we find that the mean Jacobian has dropped down near zero. As discussed in the methodology, a mean Jacobian value near zero is more in line with what we expect to see across the majority of all voxels in the nine thermal characteristics, except where there is either a strong bias in the underlying relationships the model has learned – as with those we have just discussed – or in the data. Moving out from this near-zero shell, we see a much less symmetrical pattern, indicating that the directionality of the spatial relationships which exist in the data fed into the model while calculating the Jacobians has resulted in a slight directional bias in the Jacobian field. As discussed in the methodology, this is not indicative of a failure in the model or data, but instead of a pattern in the data, on average, which the model has used when making its predictions.

Switching from the means of the Jacobians to the variances, we find that the values are often much more homogeneous across the domain of the window. This can be seen in Figure 82, where we show the variance of the nucleation time Jacobian using both a linear scale and a logarithmic scale, which does a slightly better job of highlighting small variations across much of the domain. While the mean of the nucleation time Jacobian had a very distinct pattern when cut along the x-z plane, the variation does not have a similar pattern, as can be seen in Figure 83.

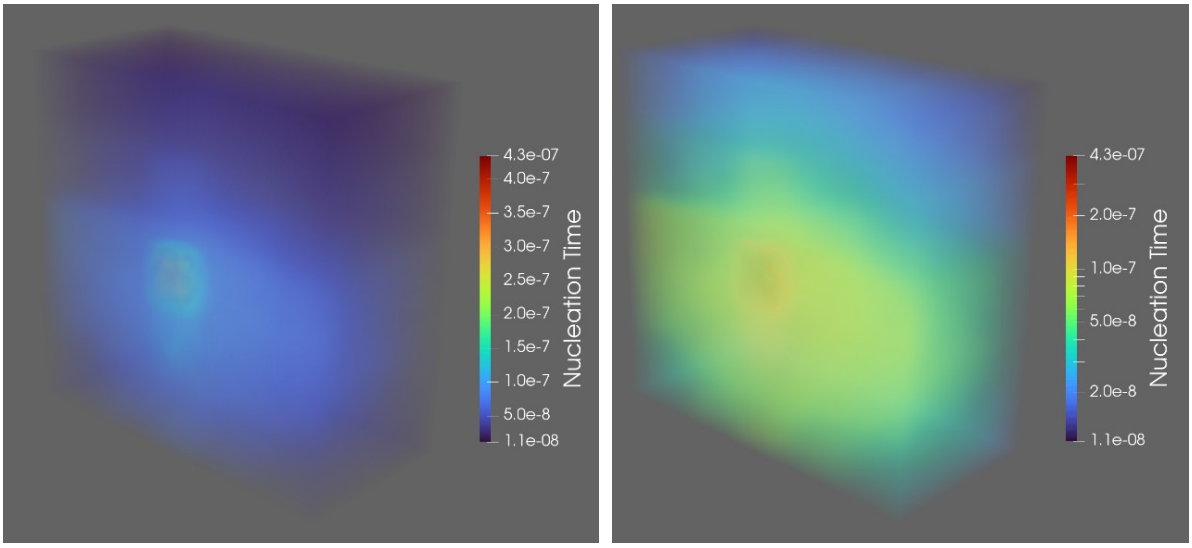

Figure 82 – The variation of the Jacobian with respect to the nucleation time, plotted using a linear and log scale to better highlight small variations.

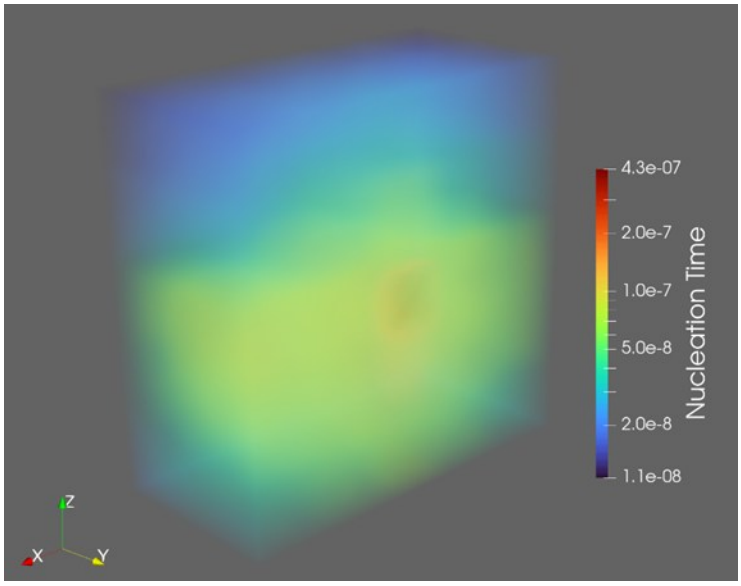

Figure 83 – The variation of the nucleation time Jacobian, cut along the same plane as the mean. The variation does not mimic the distinct pattern seen in the mean.

While the Jacobian deviations for many of the thermal characteristics are similarly homogeneous, in some of them we find distinct evidence of the first down-sampling layer of the model architecture, where the input window is reduced from 25 voxels across to 5 patches across. While this is visible to some extent in many of the thermal channels, it is particularly visible with the maximum reheat temperature as can be seen in Figure 84. Here we can see, particularly when moving along the top in the y-direction, a repeating pattern of 5 cubes, with slightly higher variation along the edges and at the

centers of the cubes. It's not immediately clear if this is problematic for the performance of the model, though it is not reflective of any patterns seen in the raw thermal characteristic data.

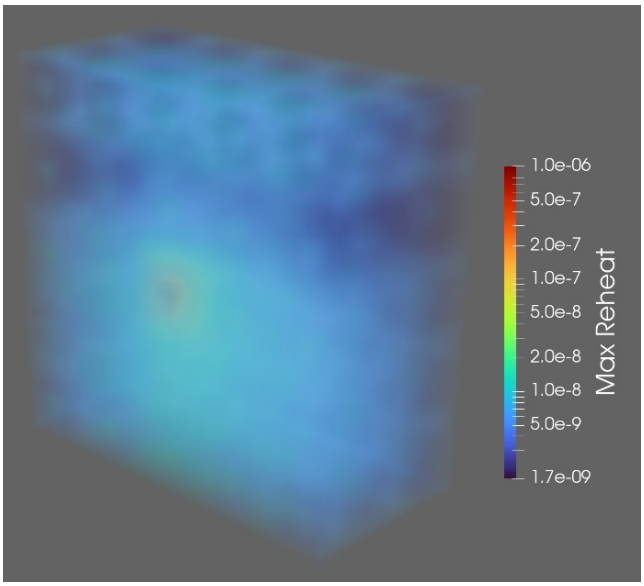

Figure 84 – Deviation of the maximum reheat temperature Jacobian, which clearly illustrates the influence of the first down-sampling layer of the model architecture.

In addition to this distinct influence of the model architecture, the variation of the Jacobian also shows evidence of the model learning particular relationships. This can be seen in Figure 85, which shows the deviation of the Jacobians with respect to the three thermal gradient channels. In these figures, it is clear that the model's predictions often rely on the thermal gradients surrounding the centermost voxel – the voxel for which predictions are being made. And, most notably, for all three gradients, the model has learned to look in the direction of the gradient in question, indicating that it has learned relevant spatial relationships from these channels.

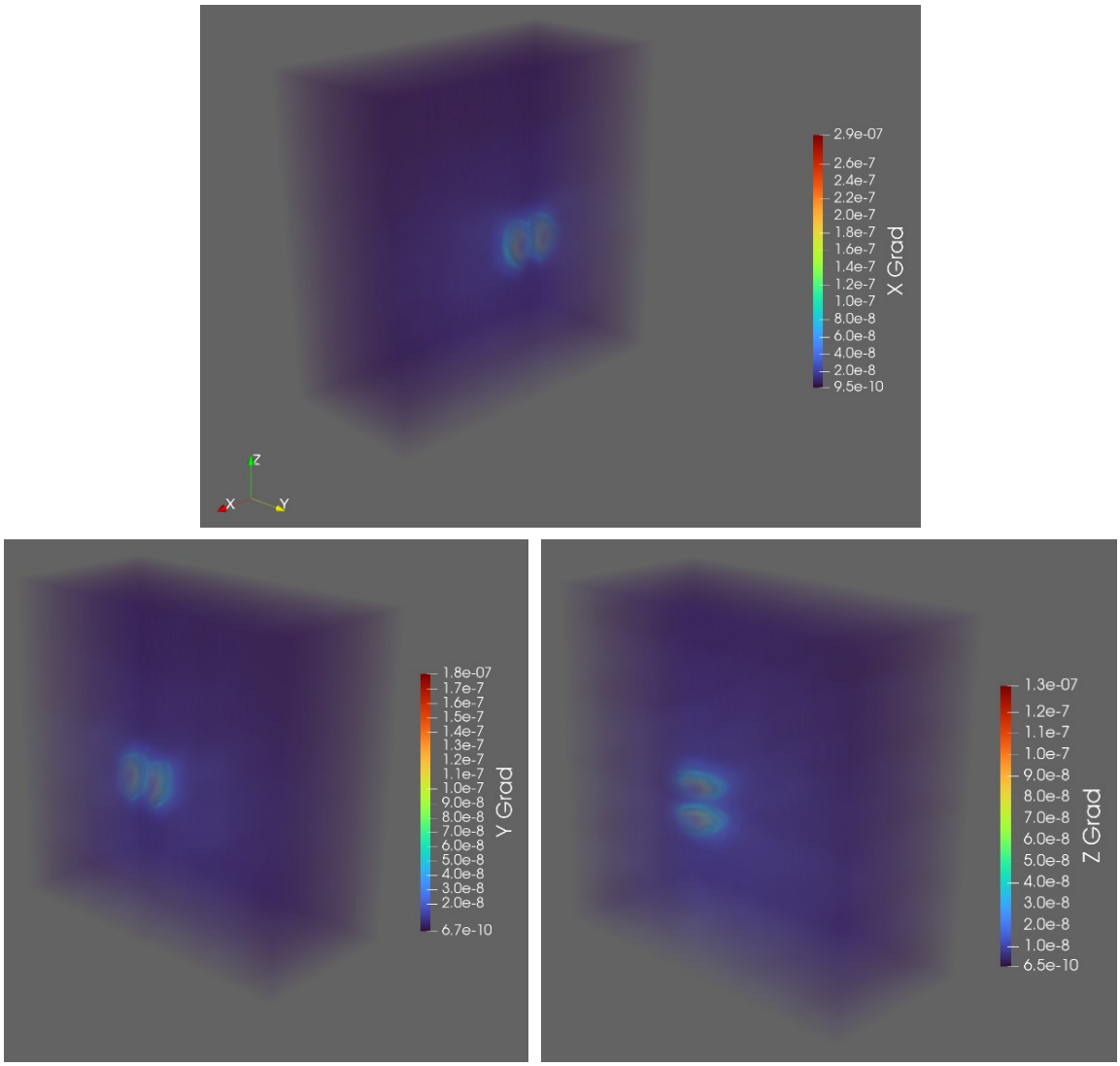

Figure 85 – Deviation of the Jacobians with respect to the three directional temperature gradients. The model assigns the most importance to the neighboring voxels in the same direction as the gradient components, indicating that it has learned the directionality of the three components.

5.2.3 Results

Now that we have explored some of the intermediate steps of this analysis, we will first discuss the overall findings from scaling the Jacobians with the thermal characteristic variances before exploring the results in detail. After scaling the statistics of the Jacobians with the variations of each of the thermal characteristics, we find that the thermal characteristics have similar levels of influence on the surrogate model's predictions. As previously noted, the averages of the Jacobians are not uniformly zero, indicating that the thermal characteristics all have regions which are in some way correlated with the average behavior of the predicted grain size distributions. This does not mean that these regions are correlated

with specific grain sizes or the average grain size, instead it indicates that across all grain sizes, there is on average a net increase or decrease in the probabilities that is correlated with the regions with non-zero mean influence. Interestingly, all thermal characteristics appear to have regions of both positive and negative average influence. For some thermal characteristics such as the solidus and annealing times, the regions with a negative average value are larger in magnitude than the positive values, though they occupy a smaller overall region. These regions of positive and negative average influence are generally most prominent in the immediate vicinity of the central voxel of interest, indicating that the thermal history of these neighboring voxels are of particular importance for determining the grain size distribution due to the consistency of the relationships.

While there are clear net influences present, the magnitudes of these influences are much smaller than the variation present in the influence, indicating that the average net influence is not the deciding factor in the predicted grain size distributions. Because of this, while the net influences are an interesting phenomenon, and may be worth studying further, they do not indicate that the surrogate model can be simplified to a small number of correlations. Looking at the magnitudes of the expected influence more closely, we find that each of the thermal characteristics has a similar overall influence, with only about one order of magnitude separating the thermal characteristics with the highest and lowest influences.

Cooling Times

Starting with the importance of the times spent above each of the 3 cooling temperatures, depicted in Figure 86-Figure 88, we find that all three of them have mean influences that are largely close to zero. Interestingly, all three have a negative average at the centermost voxel and a lower magnitude positive region surrounding it. Additionally, both the time spent above the nucleation temperature and the time spent above the solidus temperature have regions of slightly negative mean influence directly below the central region, a behavior that is not readily apparent with the annealing time. Overall, of the three, the annealing time appears to have the largest overall influence, as evidenced by having the largest scaled variance. While the annealing time itself also has the most variation of the three, the maximum variance of the annealing time is 6.3 times larger than that of the nucleation time, as shown in Figure 89, which does not account for the full 17.6 times increase of the scaled influences. This suggests that the surrogate model is influenced by the annealing time about 3 times more than it is

by the nucleation time. Another feature of interest here is the apparent increased influence of the cooling times at or below the center layer of the input window, as indicated by the higher values in the variation of the influence. As there is no matching pattern in the variation of the cooling times themselves, particularly the nucleation time where this effect is most visible, this indicates that the model itself is assigning increased importance to the lower half of the input window. While it is hard to draw any definitive conclusions from this, this behavior may contribute to the poor model performance for intermediate layers, where the regions above the point of interest should be of increased importance.

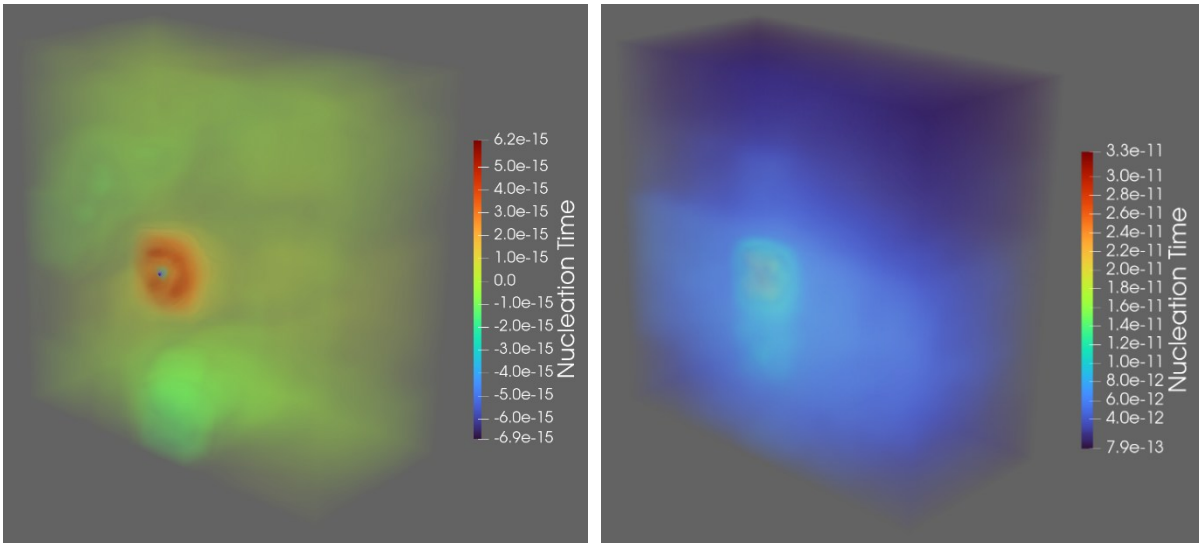

Figure 86 – Mean and variance (left and right respectively) of the Jacobians with respect to the Nucleation time, scaled by the variance of the Nucleation time.

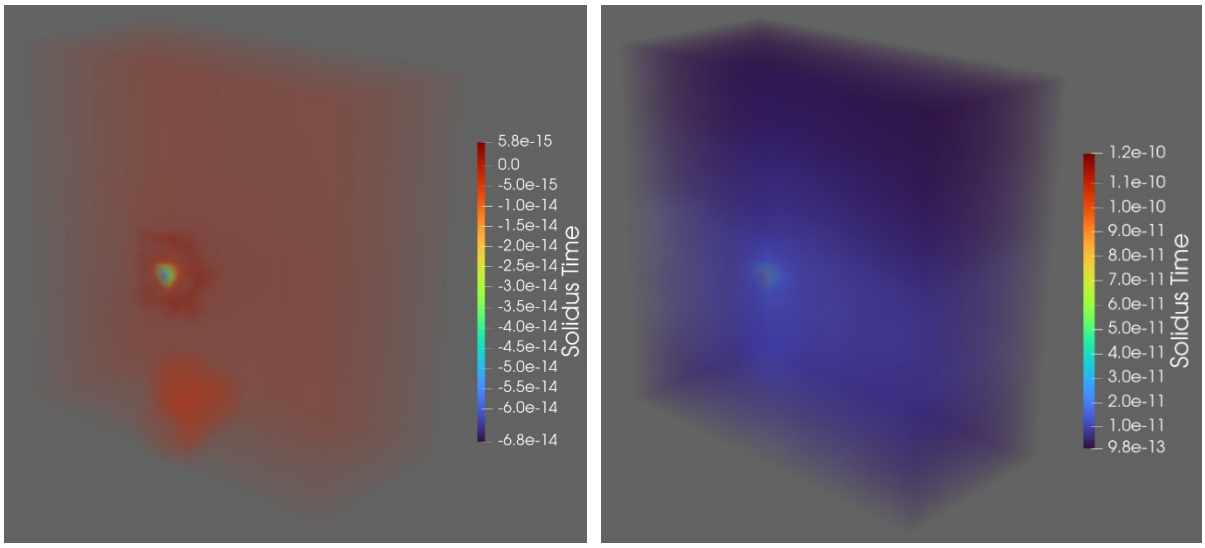

Figure 87 – Mean and variance (left and right respectively) of the Jacobians with respect to the Solidus time, scaled by the variance of the Solidus time.

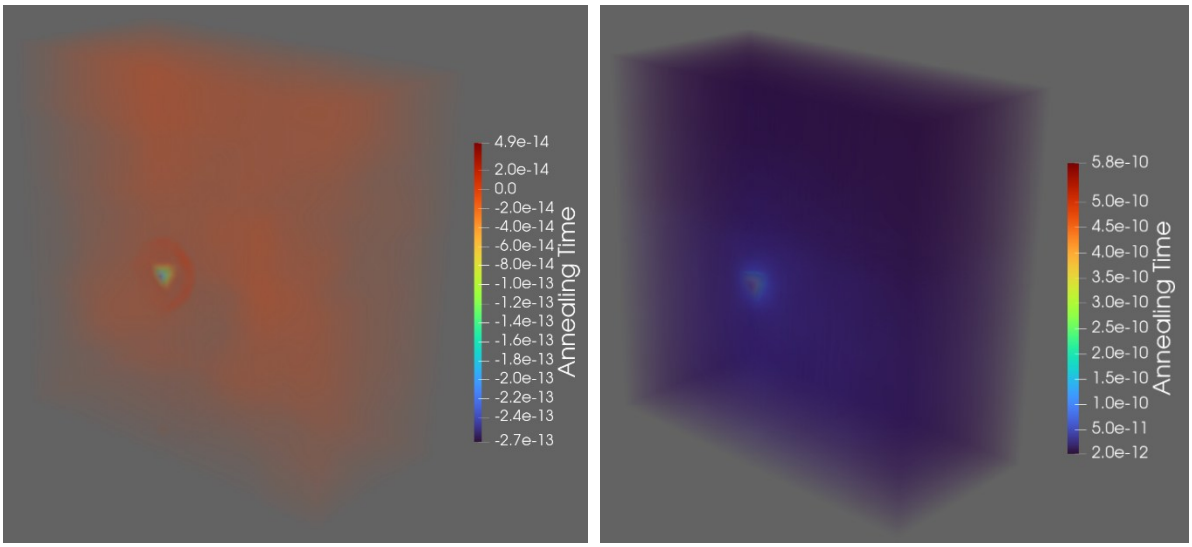

Figure 88 – Mean and variance (left and right respectively) of the Jacobians with respect to the Annealing time, scaled by the variance of the Annealing time.

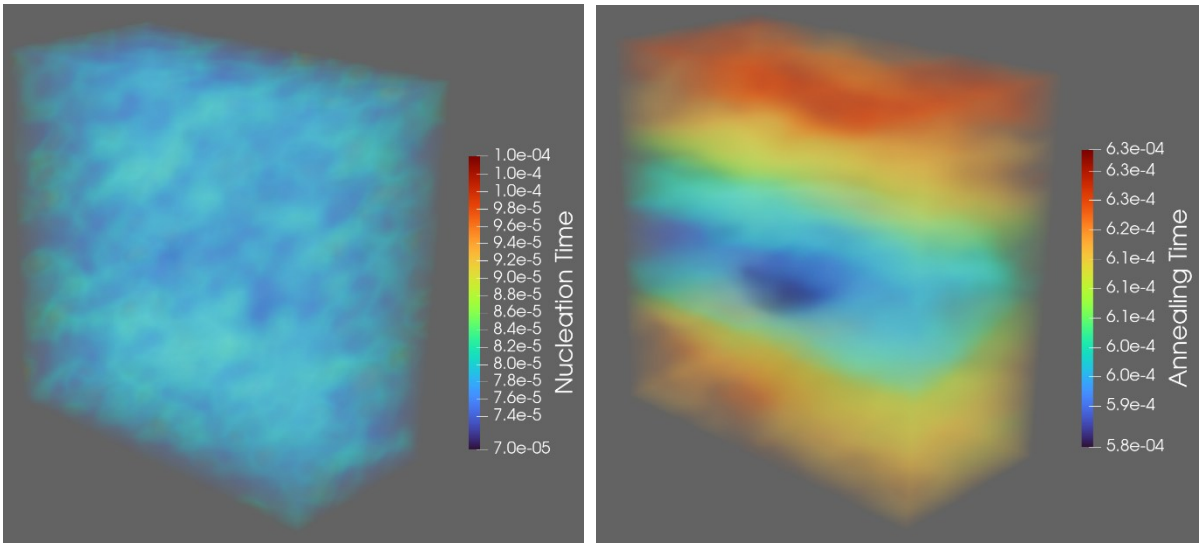

Figure 89 – Variances of the nucleation time and annealing time thermal characteristics. The maximum variance of the annealing time is 6.3 times that of the nucleation time.

Maximum Reheat Temperature

Moving on to the influence of the maximum reheat temperature, Figure 90, we find a very similar behavior to the influence of the cooling times. The primary difference is in the mean influence, which has subregions scattered throughout the input window with different mean influences. As it turns out, these are the same artifacts that were visible in the unscaled variances of the Jacobian with respect to the reheat temperature shown in Figure 84, as can be seen in Figure 91, which shows the mean influence plotted on a false scale. Overall, the maximum reheat temperature appears to have a similar magnitude of influence on the model's predictions to the cooling times.

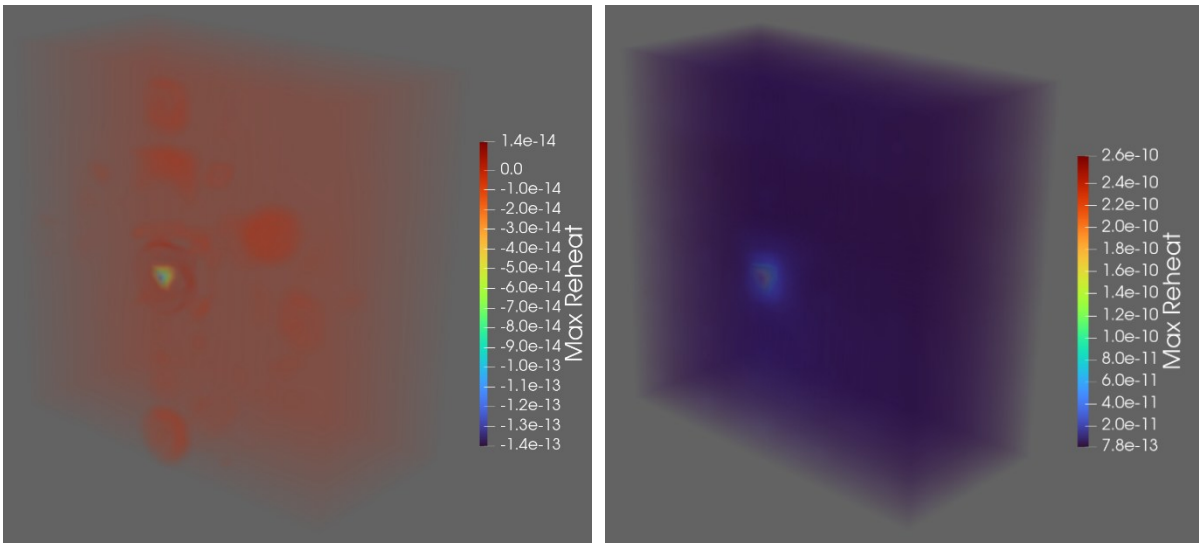

Figure 90 – Mean and variance (left and right respectively) of the Jacobians with respect to the Maximum Reheat Temperature, scaled by the variance of the Maximum Reheat Temperature.

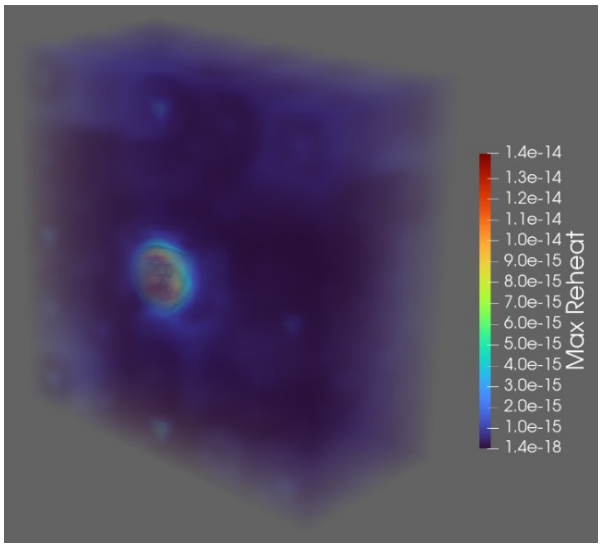

Figure 91 – Mean Jacobian with respect to the Maximum Reheat Temperature, scaled by the variance of the Maximum Reheat Temperature, adjusted to emphasize pattern created by the down sampling layer of the model.

Melt Count

The mean influence of the melt count on the grain size distribution is particularly interesting, as it has a very distinctive pattern that is not present in other results. As can be seen in Figure 92, there are six specific voxels along the 3 axes, surrounding the central point, which have a net positive correlation with the grain size distribution. Within this same vicinity around the central voxel, there are specific regions with an overall negative correlation. Because this pattern appears to be aligned to the non-physical grid and axes of the simulation, it is likely that this is in some way an artifact of the simulation

data and should be explored further. One possible explanation is that the melt count is a particularly good indicator of the microstructure formation behavior when the laser scan direction is aligned with the grid. Alternatively, it is possible that the large number of samples that had a grid aligned scan direction skewed the data, as all cases have at least one grid aligned layer. Another similar explanation would be that the cases with 90 and 45-degree laser scan direction rotations between layers significantly skewed either the samples or how the model interprets the melt count. Another possible cause is that it is a deeper problem with the data generation, and is indicative of some non-physical behavior of the SPPARKS microstructure simulations.

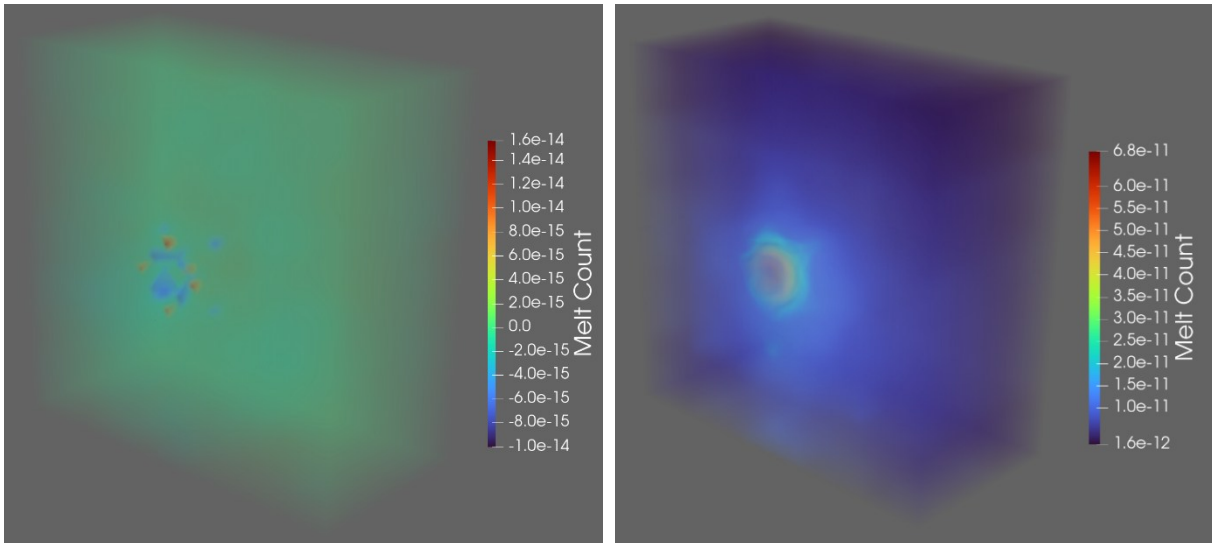

Figure 92 – Mean and variance (left and right respectively) of the Jacobians with respect to the Melt Count, scaled by the variance of the Melt Count.

While this behavior warrants further investigation, the variance of the influence appears more consistent with the other results, though it also shows strong artifacting from the down-sample convolution layer. Overall, the melt count appears to have slightly more influence on the model's predictions than the nucleation time.

Cooling Rate

Of the thermal characteristics, the cooling rate appears to have the most localized influence on the surrogate model's predictions. While the overall variation of the influence of the cooling rate is similar in magnitude to the other thermal characteristics, both the variation and mean have a very localized peak at the central voxel. Though the localized cooling rate, in conjunction with the temperature gradient, is known to have a direct relationship with microstructure formation kinematics, it

was expected that, because there is a definitive relationship with this thermal characteristic, the model would rely heavily on the behavior of the surrounding voxels. Instead, it appears that the model relies heavily on this characteristic for the voxel of interest, with a similar importance to the other thermal characteristics for neighboring voxels. The variance of the importance also appears to once again indicate that the model assigns slightly increased importance to the bottom half of the input window, similar to the cooling times. As the cooling rate and the cooling times are correlated for regions that are not reheated, the consistency of this behavior makes sense here.

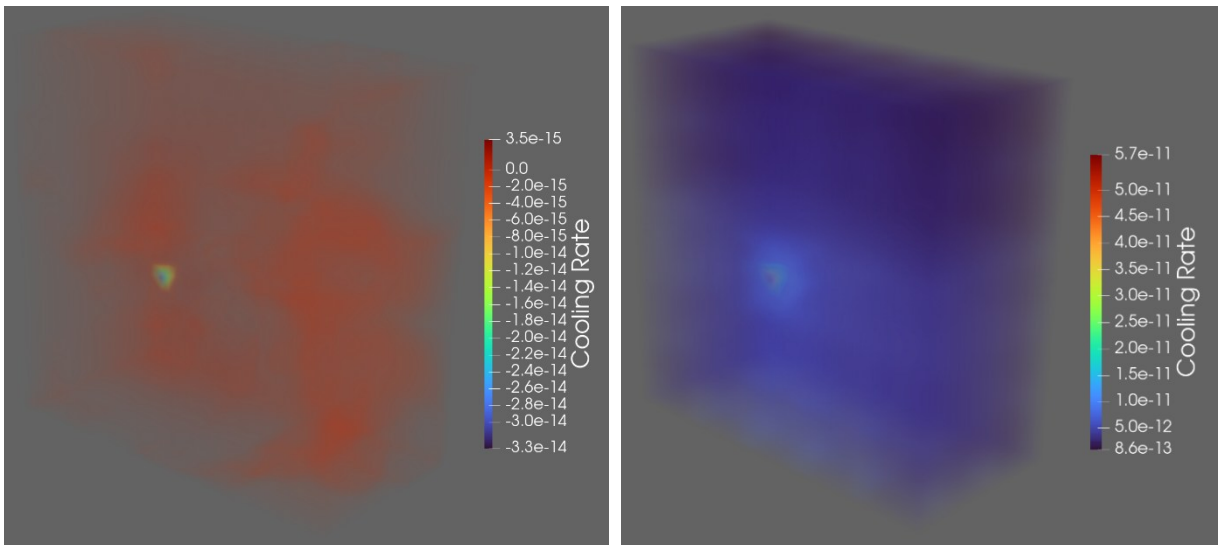

Figure 93 – Mean and variance (left and right respectively) of the Jacobians with respect to the Cooling Rate, scaled by the variance of the Cooling Rate.

Temperature Gradients

Finally, looking at the temperature gradients, we find the most distinct indication that the model has learned a physically meaningful concept. Take for instance the importance of the x-component of the thermal gradient, shown in Figure 94, here we see that the model assigns increased importance to the voxels on either side of the central voxel of interest in the direction of the x-axis. We find that this behavior holds true for the y and z-components, illustrated in Figure 95 and Figure 96 respectively. Furthermore, when looking at the mean influence of each component, we find that the average influence of these regions is either positive or negative, depending on which side of the central voxel they fall. This indicates that the model has not only learned what each of these three thermal characteristics represent, but also learned the spatial meaning of the sign of the gradients. Looking at

the influences of the thermal characteristics as a whole, it is clear that they have a similar overall importance to the model's predictions as the other thermal characteristics.

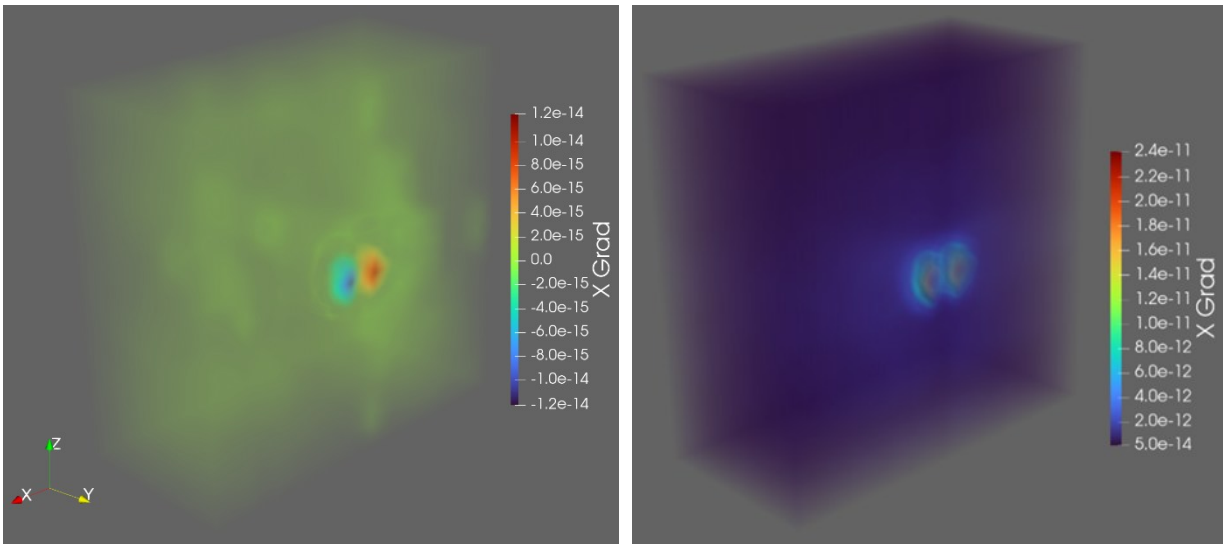

Figure 94 – Mean and variance (left and right respectively) of the Jacobians with respect to the Temperature Gradient in the x-direction, scaled by the variance of the Temperature Gradient in the x-direction.

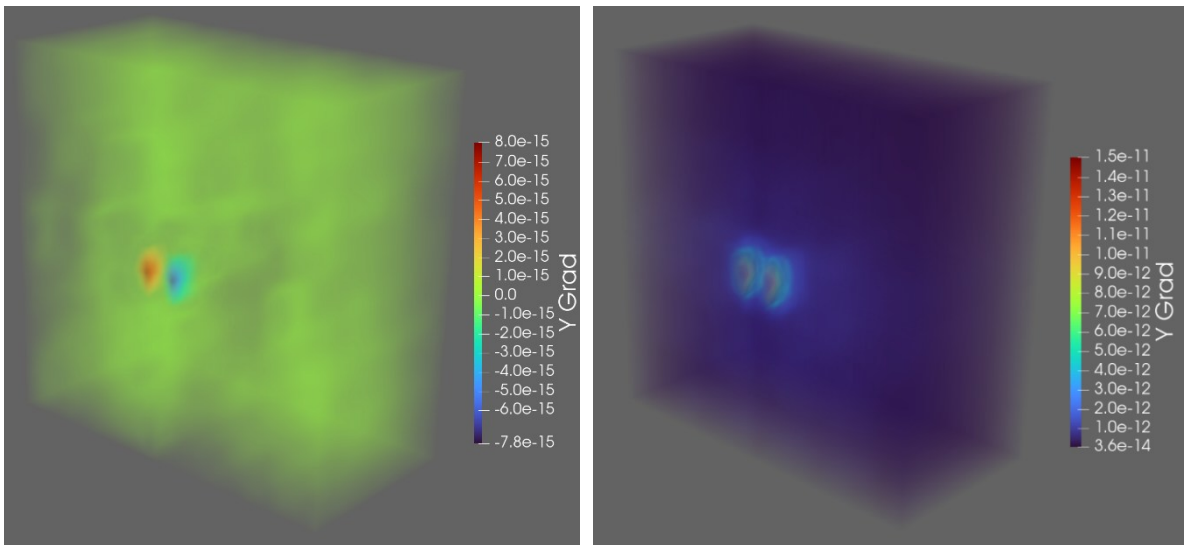

Figure 95 – Mean and variance (left and right respectively) of the Jacobians with respect to the Temperature Gradient in the y-direction, scaled by the variance of the Temperature Gradient in the y-direction.

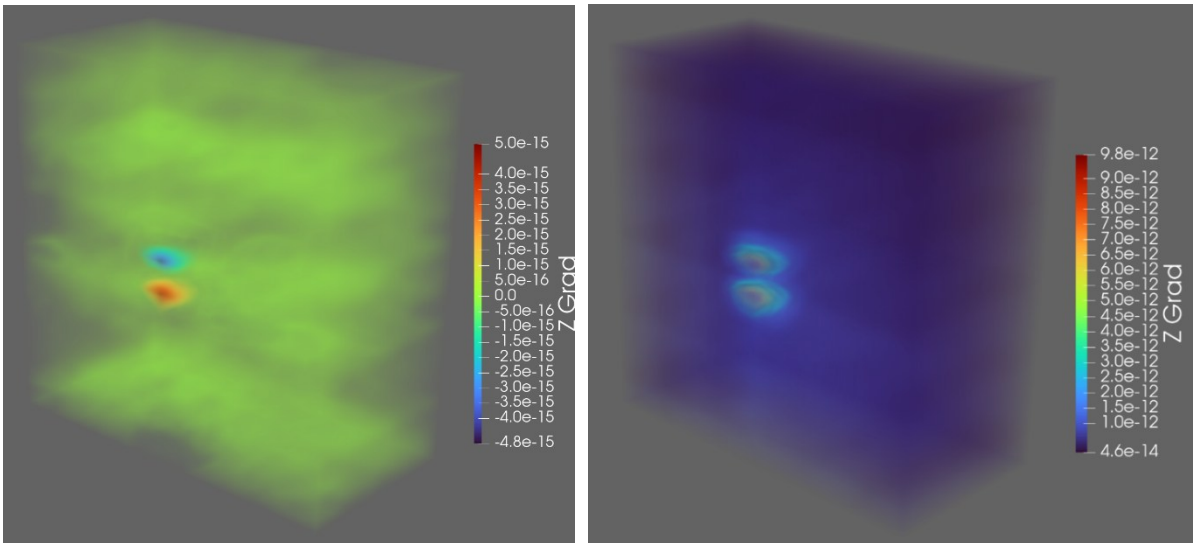

Figure 96 – Mean and variance (left and right respectively) of the Jacobians with respect to the Temperature Gradient in the z-direction, scaled by the variance of the Temperature Gradient in the z-direction.

Because we have chosen not to apply the orientation transformations utilized during training to this analysis, the three gradient components have differing magnitudes of importances, reflecting the directional biases present in the dataset. Were these results to be recalculated using all 48 orientations of the data, we would expect to see more consistent behavior between the three components, with any variation reflecting the variation present in the model's learned weights.

5.2.4 Takeaways

From these results, we can conclude that all the thermal characteristics are of similar levels of importance to the predictions of the surrogate model. While only some of the thermal characteristics appear to have distinctive spatial relationships, the strong spatial relationships of the components of the temperature gradients indicate that the model is sufficiently capable of learning necessary spatial relationships. Overall, we find that the model appears to have learned relationships which it consistently relies on, as indicated by the distinct patterns present in the importances, indicating that the model's predictions have some level of physical validity, despite not always achieving an ideal level of accuracy.

While these results show that the selected thermal characteristics were valid choices to represent the LPBF process to the surrogate model, they also reveal some areas that are worth exploring further. Specifically, the importance of the melt count should be further investigated to determine if the pattern present in the mean influence is an artifact of a deeper problem. Additionally, as the temperature gradient importances revealed, it would be worth revisiting these results with random

orientation transformations to ensure that the model is properly orientation invariant. Finally, the increased variance of the influence of the cooling rate and cooling times below the central layer of the input window should be further explored as a potential cause for the decreased accuracy when predicting intermediate layers.

5.3 Coverage of the Thermal Characteristic Space

Because neural networks generally require a large and diverse training dataset to make accurate predictions about cases not included in the training dataset, maximizing the size and diversity of the dataset while minimizing the total cost of the expensive simulations used to generate the dataset was one of the key considerations when developing the model framework. To achieve this, the voxel-by-voxel moving window approach was adopted under the hypothesis that this would result in significantly more variation in the dataset than was present in the LPBF process parameters used to generate the dataset. The reasoning for this hypothesis was that, despite the inherent repetition of the raster laser scan patterns, the complexity of the LPBF process makes it unlikely that any given segment of an LPBF print will have the exact same thermal history as another segment from the same print.

While this hypothesis is somewhat supported by the ability of the model to generalize to some LPBF parameters not previously seen, as shown in the previous chapters, here we aim to more directly analyze the validity of this fundamental assumption. At its core, this requires addressing how well the dataset covers the thermal characteristic space – the space that contains all possible valid combinations of the thermal characteristic values that make up the input to the model. Because each input to the model consists of 140,625 values – 15,625 voxels with 9 thermal characteristic values each – the thermal characteristic space can be thought of as some subregion of a 140,625 dimensional space. For the purposes of this surrogate model the bounds of this subregion are a factor of both what is physically possible, and what is achievable with the LPBF process. For instance, it is not physically possible to have a negative cooling time, and it is not reasonable to expect our model to accurately model wire arc additive manufacturing. However, while we can provide such general limitations on the thermal characteristic space, it is not possible to provide precise bounds for the thermal characteristic space as none of the thermal characteristics have physical upper bounds, and establishing practical bounds would require knowledge of all possible LPBF prints.

Because of this, it is not possible to quantify how well the dataset covers the thermal characteristic space, and we must instead qualitatively evaluate the dataset. In this section we develop a methodology for simplifying the high-dimensional data down to an easier to visualize form, which we can use to better understand the spread of the dataset.

5.3.1 Simplified Thermal Characteristic Space

Though the full 140,625 value model inputs exist in a space that is too high-dimensional to visualize directly, we can gain some initial insight into the data by inspecting the distributions of the individual thermal characteristic values present in the dataset. This representation of the thermal characteristics completely removes all spatial information available to the model, and decouples the thermal characteristics that belong to a given voxel. Despite this, this limited representation provides us with insight into the behavior of the individual thermal characteristics across the dataset.

Ideally, the distributions of each of the nine thermal characteristics would cover all values that are physically possible in the LPBF process with 316 stainless steel, as it would indicate complete coverage of the thermal characteristic space. While it is impossible to know the true ranges of many of the thermal characteristics, the majority of them have a lower bound of zero, due to their physical meanings. The two exceptions to this are the X and Y components of the temperature gradients due to the sign's representation of directionality, something that is not true for the Z component due to the highest temperatures always occurring at the top surface of the print bed. Additionally, the maximum reheat temperature has an upper bound at 1723K – the melting point of the 316 stainless steel being modeled – as any voxels heated above this temperature are melted and have their reheat temperature reset to zero. While the distributions would also ideally cover their ranges uniformly, due to the general importance of balanced datasets for machine learning, this simplified representation does not account for the full complexity of the problem, and any biases present are representative of the physics of the LPBF process. For this reason, uniform distributions are not necessary for this data.

Figure 97 shows the distributions of each of the nine thermal characteristics for a subset of the full dataset, primarily the single-layer cases, plotted as stacked histograms, where each layer represents a different LPBF process parameter case. This demonstrates the variations present in each of the nine thermal characteristics created by changes in the LPBF process parameters and part geometries.

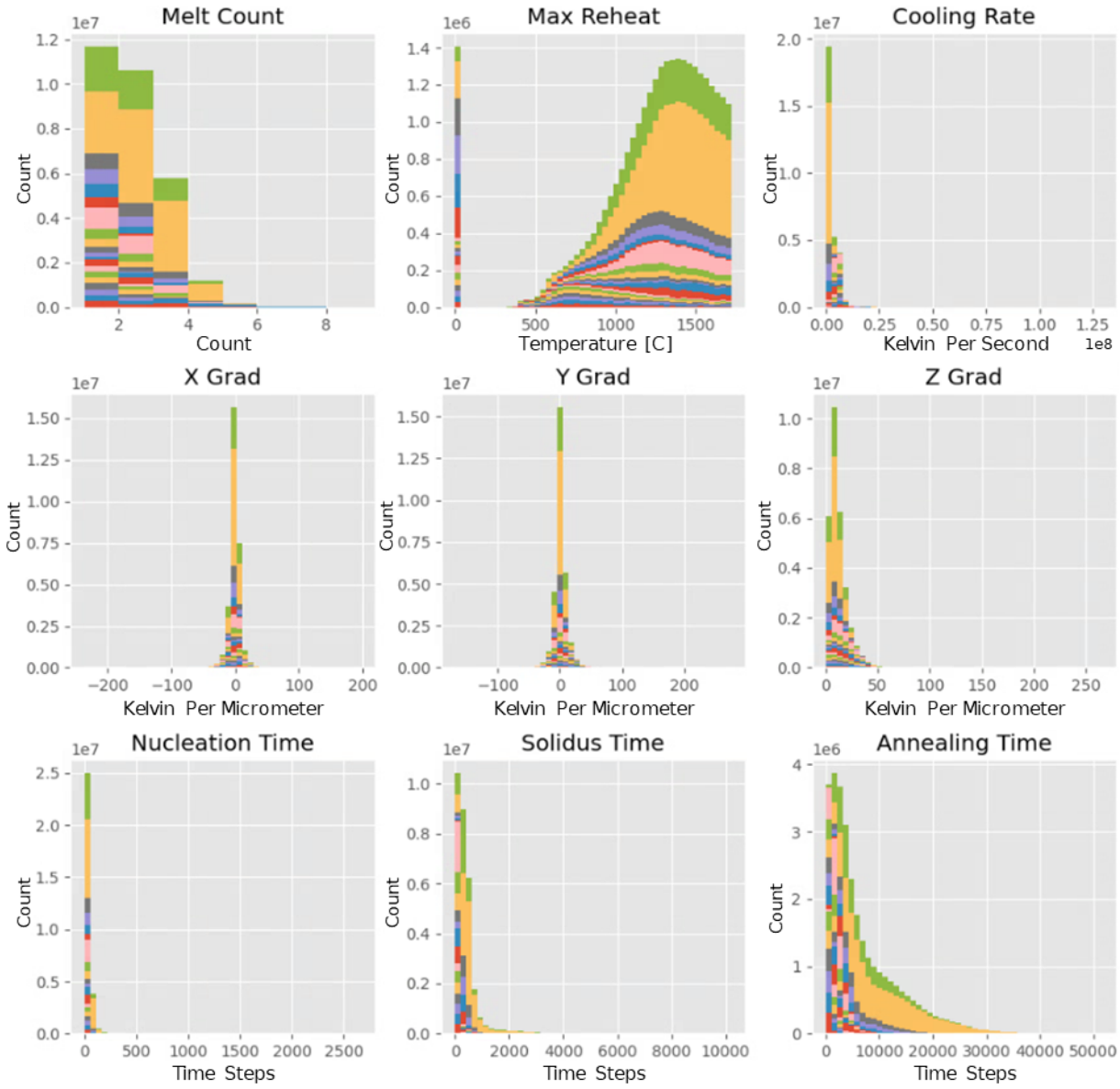

Figure 97 – Stacked histograms showing the distributions of the thermal characteristics across all voxels for several LPBF process parameter cases.

Due to the significant variation in the number of voxels in each simulated part, some of the LPBF parameter cases have significantly more influence on the resulting distributions. While this influence may be important to understand for the purposes of balancing the machine learning dataset, it obscures how the variation in the process parameters influences the variation within the thermal characteristics. To counteract this influence, we can first normalize the distributions belonging to the individual cases

before agglomerating them into a cumulative stacked probability distribution[2]. Distributions generated using this method, with a much larger subset of the full dataset, can be seen in Figure 98.

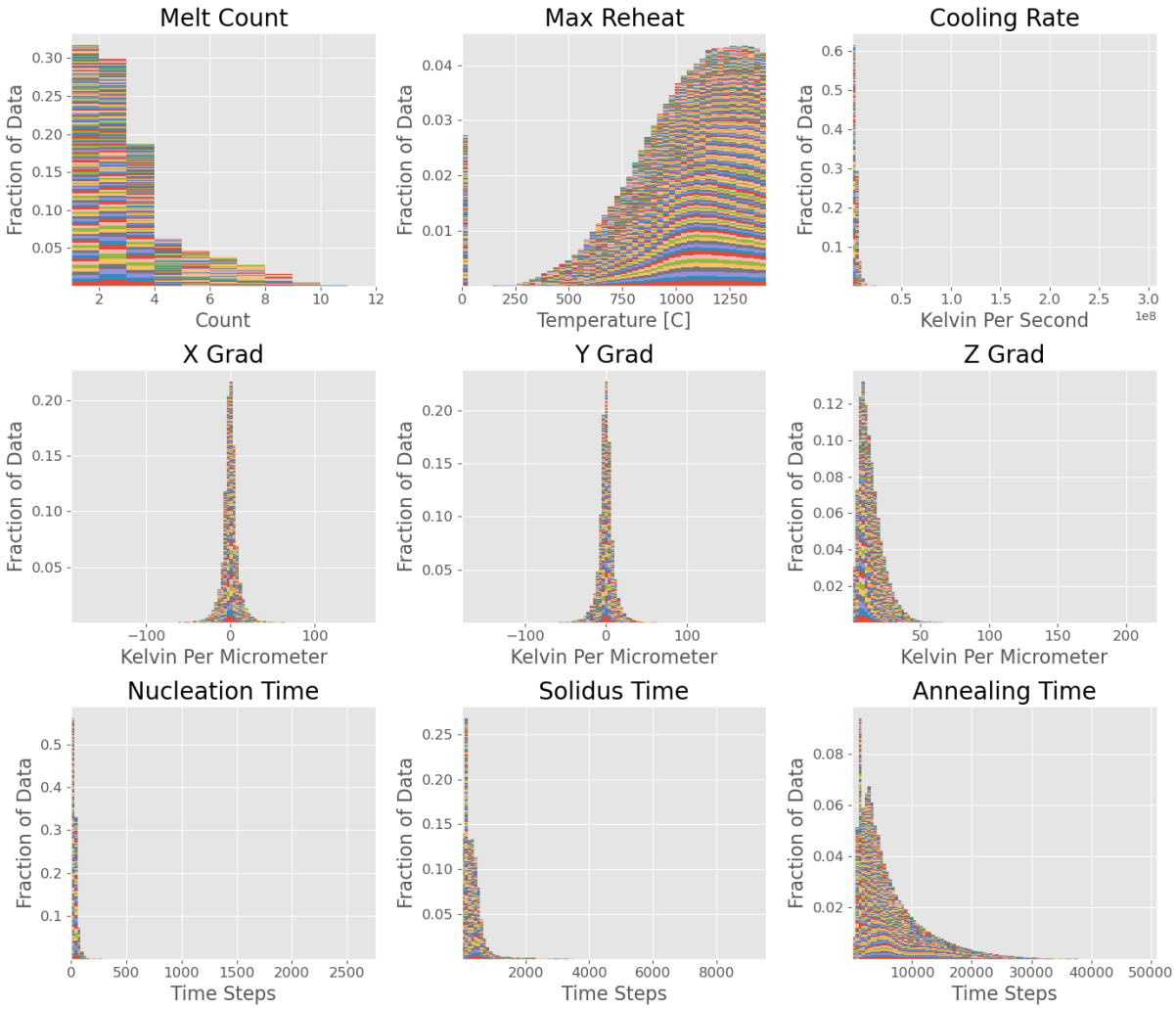

Figure 98 – Stacked histograms showing the distributions of the thermal characteristics, across a much larger subset of the LPBF process parameter cases, which have been normalized before and after stacking.

These results show very similar overall behavior to the more limited Figure 97, with somewhat larger ranges for the melt count and cooling rate. From these plots, it is clear that there are consistent trends in the thermal characteristics which result from the LPBF process, regardless of the specific process parameters and geometries. This indicates that while each individual process parameter case results in a variety of thermal characteristics, it is unlikely that any process parameter sets exist which have significant deviations from these trends. This means both that it would be hard to create a dataset

[2] Not to be confused with the statistics terms cumulative distribution or cumulative density function (CDF)

which more uniformly covers the thermal characteristic space, and that such coverage may not be necessary for the creation of a surrogate model that is generalizable to new process parameters.

5.3.2 Dimensionality Reduction

To make the full 140,625 value datapoints more easily representable, while maintaining the influence of the spatial information, we can use dimensionality reduction techniques. While there are many existing dimensionality reduction methods, these methods generally focus on algorithmically reducing the number of dimensions to the smallest representative set [164]. While such methods may be useful for some analyses of this dataset, for the purposes of this analysis it is desirable to preserve the meaning of the nine thermal characteristics. Additionally, this dataset poses an additional complication that must be considered: there are 48 valid orientations for each datapoint (i.e. input cube), which must be considered in order to fully understand the distribution of the data. For these reasons, we do not use conventional dimensionality reduction methods, but instead develop a method to generate a simplified representation of the nine thermal characteristic channels in an orientation independent manner.

Because the spatial relationships present in the thermal characteristic data are important for making predictions about the microstructure, it is necessary to encode the spatial relationships when dimensionally reducing the data for analysis. Additionally, because the relationships between the thermal characteristics and microstructure are not orientation dependent, it is necessary that this spatial information be encoded in a way that is also orientation independent. To accomplish this, we first establish an orientation invariant reference point against which we can measure. The datapoints can then be measured against this datapoint to give dimensionally reduced relative coordinates. To create this rotation invariant reference point, we start by averaging all the datapoints in the dataset, giving us a datapoint that falls somewhere in the middle of the full dataset when represented in the 140,625 dimensional space. To make this reference point orientation invariant, we take the 4-dimensional representation and average all 48 orientations of the array, maintaining the independence of the thermal characteristics in the same way used while training the surrogate model. With this, we now have a fully symmetric, rotation invariant average of all possible orientations of all the datapoints in the dataset against which to measure.

Using the root mean squared error (RMSE), or other distance measures, we can measure the distance of each datapoint from this reference point on a channel-by-channel basis, giving us a 9-dimensional measure of how close each datapoint is to the average of the dataset. While the resulting 9-dimensional space is still too many dimensions to fully visualize, some of the trends and relationships in the data can still be explored using subsets of the dimensions. For example, Figure 99 shows the relationships between the dimensionally-reduced cooling rate and each of the 8 remaining thermal characteristics, for a subset of the LPBF parameter sets, with each parameter set plotted using a different color[3]. Because each process parameter case has tens of thousands to millions of data points, these plots are made using a random sample of 1000 data points from each case, resulting in the clusters appearing sparser than they actually are. Despite this artificial sparsity, the overall trends present in each of the LPBF cases are clear, and we can see that there is substantial variation both within and between LPBF parameter sets.

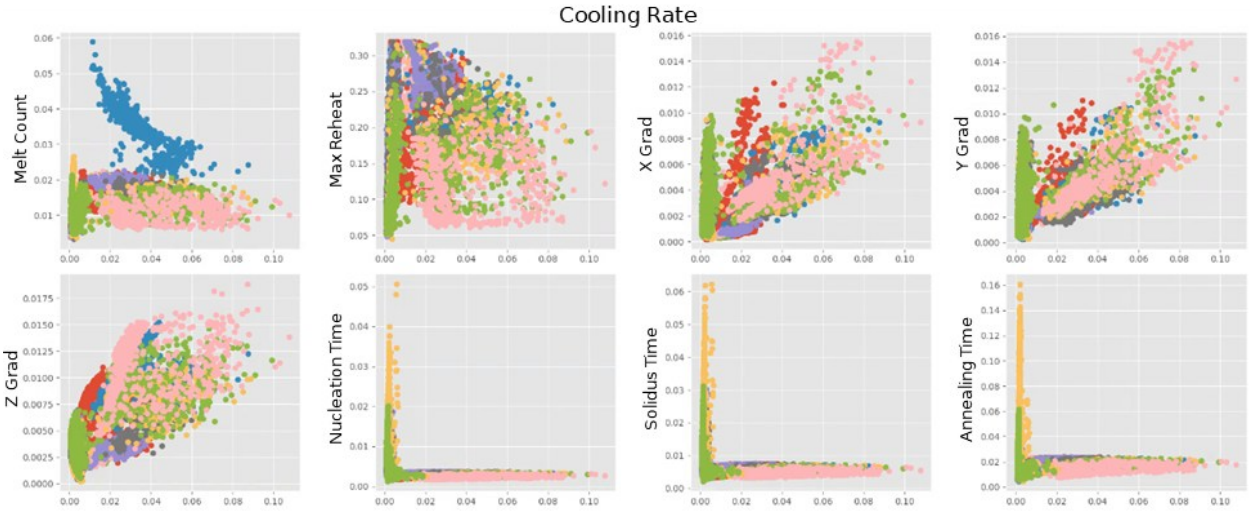

Figure 99 – Plots showing the relationships between the distances of the cooling rate and each of the other 8 thermal characteristics for a sample of datapoints from a subset of the LPBF cases.

Looking more closely at Figure 99, we see that all of the thermal characteristics can vary independently of the dimensionally reduced cooling rate, particularly when the distance of the cooling rate from the mean/reference is small, as evidenced by the vertical clusters along the y-axis. In the case

[3] There are fewer colors in the color map than there are LPBF parameter cases, so not all points with the same color are from the same LPBF case. However, the overlapping of different colors usually make such clusters at least partially distinguishable.

of the three cooling times, the majority of the variation in the dimensionally reduced cooling rate occurs when the cooling times are very near to their mean/reference points. Additionally, we can see that the dimensionally reduced cooling rate has a positive correlation with the three dimensionally reduced thermal gradients. Because this is dimensionally reduced data, this correlation means that cooling rate further from the mean cooling rate corresponds to a thermal gradient further from mean thermal gradients, with no bearing on the actual values of the cooling rates and thermal gradients. In contrast, the LPBF parameter case corresponding to the blue cluster of the cooling rate-melt count plot has a negative correlation between the two dimensionally reduced characteristics.

Turning our attention to the melt count subplot, we can see that for the LPBF parameter set corresponding to the blue points, there is a fairly strong negative correlation. This implies that the further from the mean the cooling rate is, the closer to the mean the melt count is, and vis versa. However, there does also appear to be a small subset of the blue cluster which has a slight positive correlation between the distances of the melt count and cooling rate. This dual behavior supports the concept that the complexity of the LPBF process can produce a diverse array of thermal characteristics within a small number of LPBF parameter sets.

While the magnitudes of the distances of each of the thermal characteristics from the reference points are all comparable, the values themselves are not directly comparable due to each of the thermal characteristics having different physical meanings and the large differences in the magnitudes of their respective ranges. As the magnitudes of these distances have no practical meaning beyond comparing datapoints within a thermal characteristic, the range of the distances of each thermal characteristic can be normalized to a maximum magnitude of one. While this does remove some information about how much of the theoretical thermal characteristic space the dataset covers, the lack of prior knowledge about the true thermal characteristic space makes this information valueless. However, this scaling does allow us to combine the thermal characteristic distances into lower dimensions, for the purposes of visualization, without any thermal characteristics overwhelming the others. These normalized and combined values allow us to visualize relationships amongst all thermal characteristics simultaneously. Figure 100 shows the distances of each of the nine thermal characteristics from the reference point – on the x axes – versus the combined distances of the remaining eight thermal characteristics – on the y axes – for a random sample of 1000 datapoints from each process parameter case.

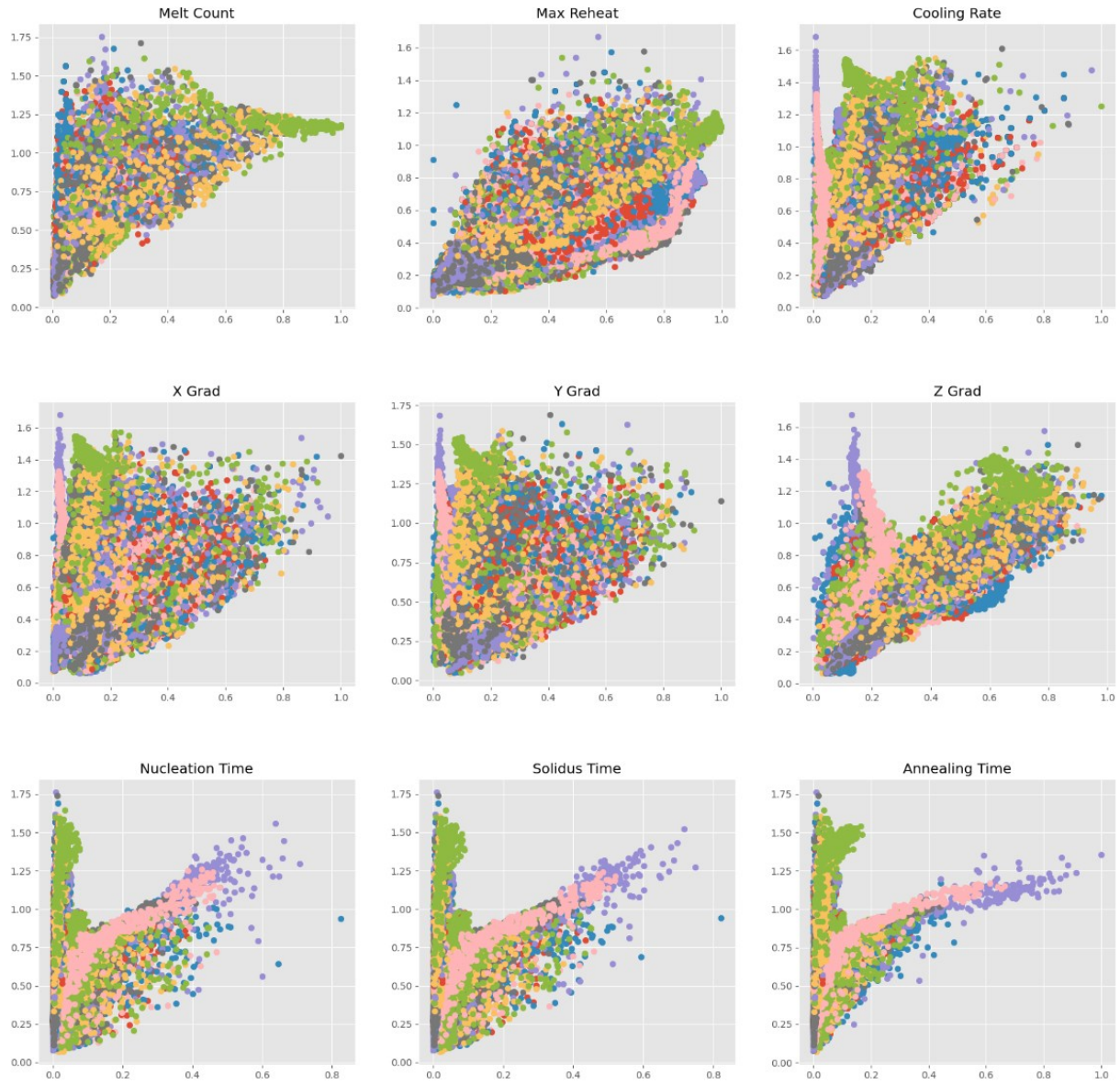

Figure 100 – Plots showing the relationships between each of the thermal characteristics and the combined total of the remaining 8 characteristics for 1000 datapoints from each LPBF parameter set.

From these plots it is clear that while there are trends within the data, there is also a significant amount of variation. Additionally, the maximum value of the combined values is consistently around 1.75, much lower than the theoretical maximum of 8, indicating that datapoints with the furthest distances from the reference point on one thermal channel are not necessarily the furthest on the other thermal channels.

Taking a closer look at the individual plots, starting with the melt count, we see that there is a significant amount of variation among the other thermal characteristics when the melt count is near the

mean. Interestingly, this variation significantly decreases as the melt count gets further from the mean. This holds true for all of the LPBF parameter sets plotted, but is particularly distinct with the green dataset. While it's not clear what the cause of this behavior is, it is hard to say definitively if this behavior is incompatible with our goal of maximizing the coverage of the thermal characteristic space without further investigation of this scenario. This is because, while at face value a strong correlation in the data indicates a lack of spread, a strong correlation may simply indicate that we are reaching the bounds of possible outcomes of the LPBF scenario, which is ideal for fully covering the thermal characteristic space.

Moving on to the maximum reheat temperature plot, we can see that there is room to expand the diversity of the dataset for low distances of the reheat temperature from the mean, as indicated by the few datapoints which do fall in this range. Interestingly, there is a fairly consistent lower bound to these clusters, which may indicate that we are nearing the bounds of the thermal characteristic space, though it may be possible to expand beyond this with process parameter variations not included in the dataset.

Next, looking at the cooling rate plot, we can see that the majority of the cooling rate distances are less than six tenths of the maximum. While this may be exacerbated by the random sampling of the datapoints, it indicates that the regions of the thermal characteristic space furthest from the reference point are not as well covered as the closer regions. Additionally, here we see an interesting pattern, where the highest values of the 9 combined thermal characteristics drop sharply as the cooling rate moves further from the reference before quickly increasing after passing a certain point, creating a cleft in the plot, a feature which is mirrored to some extent in the remaining plots.

Moving on to the x and y directions of the thermal gradients, we see nearly identical behaviors. This is to be expected as, for the most part, the behavior of the LPBF process is identical in the x and y directions, which are both in-plane. The only variations we expect to see here are due to sampling and the influence of the default orientation used when setting up the simulated LPBF prints. Once again, we can see a distinct cleft in these two plots, and an overall positive correlation resulting in an "upper triangular" shape. While the z-direction of the thermal gradient has similar overall patterns, with each cluster having a similar overall shape and position to the corresponding x and y cluster, each of the clusters appears to be spread across a wider range of z-gradients. This is interesting as it indicates that the z-direction of the thermal gradient has slightly different correlations with the other thermal

characteristics than the x and y directions. Additionally, it appears that some of the clusters – particularly the vertical pink cluster – also have a slightly weaker correlation; however, the much stronger diagonal behavior indicates a stronger correlation overall.

Finally, looking at the cooling times, we see a distinct pattern across all three. From this pattern it is clear that there is a strong relationship between the distance of the cooling times from the reference points and the combined distances of the other channels. Interestingly, the strength of this relationship appears to increase as we go from the nucleation time to the annealing time, as indicated by the smaller spread of the datapoints. This may however be an artifact of the differing magnitudes of the three cooling times, as a small variation in cooling time may be a much larger percentage of the nucleation time than of the annealing time, resulting in a more visually obvious difference. Despite this strong relationship, there is also a large amount of variation in the other channels when the cooling rates are close to the reference points. In the annealing time plot, the green cluster which protrudes from the upper regions of this vertical region seems to indicate that it may be possible to find process parameters which reduce the size of these cleft regions. These parameter sets would be important to include in the training data, as while they appear to be uncommon, their behavior differs substantially from the rest of the dataset, which may result in decreased model accuracy. Additionally, like with the cooling rates, the majority of the data occurs in the lower half of the cooling times, with a few outliers and parameter sets in the upper half, indicating that more data may be needed to fully cover these thermal characteristics.

5.3.3 Takeaways

While it is not possible to quantify how well the dataset covers the thermal characteristic space, it is clear from this data that the hypothesis – that the moving window approach would create significantly more variation than is present in the process parameters alone – holds true. Despite this, it is also clear that the cooling rate and cooling times channels are not as well covered as the other thermal characteristics, and that more process parameter sets need to be included to create consistent coverage. While the distinct edges and strong correlations of many of the clusters appear to indicate regions which the dataset does not cover, these may instead be indicative of regions which are unlikely to occur during the LPBF process. Overall, it is clear that the dataset could use more process parameter cases, and that the moving window approach accomplishes the goal of maximizing data diversity.

Chapter 6 Conclusions

In this work we have introduced a framework for the creation of machine-learning based surrogate microstructure models for the laser powder bed fusion process. We then demonstrated the potential of the framework with the development and analysis of a microstructure surrogate model for 304 stainless steel manufactured with the laser powder bed fusion additive manufacturing process.

As part of the model framework, it was shown that using ensembles of microstructure simulations allows us to make predictions about microstructures on a voxel-by-voxel level, providing us with spatially defined grain size distributions. While these spatially defined microstructure statistics do not have an equivalent in traditional materials science methods, they may provide potential paths for discovering microstructure relationships which would not otherwise be accessible. In the process of creating and analyzing this surrogate model a method for visualizing spatially defined distributions was developed, allowing for the observation of spatial trends in the probabilities of microstructure statistics within printed parts.

Nine thermal characteristics were selected to represent the complete thermal history of the LPBF process for the purposes of predicting grain size distributions using the surrogate model framework. An existing state of the art image classification neural network architecture was adapted to fit the needs of the surrogate model framework, and shown to learn to accurately predict microstructure statistics during training. Further iterations of the model hyperparameters were performed using Bayesian optimization and resulted in a significant decrease in the number model parameters with no meaningful decrease in model accuracy.

Initial predictions from the surrogate model revealed an apparent decrease in the stochasticity of the predictions compared to the training data. Experiments were carried out which revealed that the model learns to average out the stochastic influences of the data in a way that is consistent with the statistics produced by much larger simulation ensembles. The implications of this result were further explored by training several versions of the surrogate model on varying ensemble sizes. This experiment revealed that the model's predictions are consistent with much larger ensembles than those it was trained on for all ensemble sizes tested.

The model's accuracy in realistic use cases, where the LPBF parameters of interest would not be included in the training data, was tested by iteratively removing subsets of the dataset when training the model and comparing the model's predictions against the excluded data. This revealed that for regions of the parameter space that are densely covered by the training data, the model can make reasonably accurate predictions for new process parameters. As the process parameters entered regions of the process space that were not as well covered by the training data, the model's predictions became less accurate, but largely captured the overall trends. Only when the process parameters varied significantly from the rest of the dataset, as was the case with both the 45° and 90° scan rotation between layers, did the model's predictions substantially diverge from the simulation ensemble-based ground truth.

While the accuracy of the model appeared to indicate that the model was learning relationships between the selected thermal characteristics and the resulting grain size distributions, this was tested directly by adapting machine learning interpretability methods to provide a statistical insight into the model's internal workings. This revealed that the model was relying on all nine thermal characteristics to similar degrees. Furthermore, the distinct behavior of the model with respect to the temperature gradients, revealed that the model was learning at least some physically meaningful relationships.

Finally, an underlying assumption of the model framework – that the moving window approach would create more variation in the training data set – was explored via reduced dimensional representations of the dataset. These representations revealed interesting trends in the data, and indicate that while the existing dataset does have a large amount of variation, there is room for improvement.

In summary, it was shown that the framework for the development of a surrogate microstructure model introduced here is capable of producing surrogate models that are both significantly faster than equivalent microstructure simulations, and maintain a reasonable degree of accuracy given sufficient training data. While the research presented here focuses on a surrogate model for the prediction of grain size distributions, the framework is agnostic with respect to the microstructure metric being predicted. Additionally, while the framework, as presented, is intended for use with microstructure statistics, it is likely that the framework can be used for the creation of a surrogate model capable of predicting spatially defined distributions of any material characteristic, as produced by the LPBF process. Furthermore, it may be possible to adapt the framework for the

prediction of less stochastic outcomes of the LPBF process, such as the residual stress within printed parts. While we have not tested any of these directly, we have demonstrated that thoughtful design of the model framework, along with careful selection of the characteristics used to represent the LPBF process, allow for the successful creation of a surrogate model capable of making predictions about the desired LPBF process outcome.

6.1 Future Work

While we believe that the research presented here adequately introduces and analyzes the framework for the creation of surrogate microstructure models for the LPBF process, it has opened up many avenues for further exploration. Several such avenues have been identified for furthering this research and exploring its broader applicability.

The first of these avenues is exploring the applicability of the spatially identified microstructure statistics created as part of this modeling framework. While this method for representing microstructures was invaluable for the purposes of creating the surrogate model, it is not a widely used representation of microstructures – in fact, no instances of such a representation were found in literature. Developing methods for applying conventional understandings of microstructures to this type of representation would greatly increase its value. Furthermore, we believe that as high-throughput experimentation becomes more viable[1], this method for representing microstructure statistics could find uses in conventional experimental materials science. Finally, while such microstructure representations are not traditionally used when modelling the stress response of materials, we believe that a surrogate model for the prediction of mechanical behavior could be trained using these representations as inputs, in a way that reduces the stochastic influence of microstructures.

Another avenue which deals with the data presented here is the further exploration of the dataset, particularly with regards to the thermal characteristic space. While we explored the implications of the spread of the dimensionally reduced thermal characteristic space, we were unable to identify a metric that could be used to quantitatively answer questions about the data and the thermal characteristic space. We hypothesize that were such a quantitative metric to be identified, it could be

[1] Though they do not accelerate the printing process, newer analysis techniques such as Large-Area Polarized-Light Microscopy [165] and 3D EBSD [166] promise to increase the amount of data that can be collected about experimental microstructures.

used as a heuristic for predicting the accuracy of the model's predictions about new process parameter sets, providing some level of uncertainty quantification.

The next avenue for exploration is the use of this framework for the development of surrogate models that can predict other microstructure metrics. While we have proven here that this framework can generate a model that can predict grain size distributions, we have not explored what challenges might arise when predicting other grain morphology metrics. We believe that surrogate models for the prediction of crystallographic orientation would be particularly interesting, due to both the multi-dimensional nature of crystallographic orientations, and the increased computational costs of modeling crystallographic orientation with conventional simulations. A potential stepping-stone for arriving at such a model would be the development of a surrogate model for the simultaneous prediction of the grain diameter and aspect ratio. While grain diameter is an invaluable metric for the prediction of mechanical behavior, the increased formation of elongated columnar grains is one of the main concerns of LPBF practitioners. Simultaneous prediction of the aspect ratio and grain size would do a much better job at capturing this behavior than the current diameter only metric.

Finally, continuing with the theme of further development of the model itself, we have identified two modifications to the overall model framework that we believe are worth exploring. The first such modification is the switch away from the moving window architecture in its current form, to an architecture that is capable of providing the model with context about the full domain. It's unclear what such an architecture would look like – perhaps a U-net – but we believe that this would better allow the model to make accurate predictions about varying print sizes, where the current moving window is incapable of providing sufficient context about the height of tall prints. The second modification we believe is worth exploring is the bypassing of the thermal model altogether. While we believe that having a physics driven input to the model is invaluable, both for reducing uncertainty and for exploring the model, the thermal model is currently the bottleneck for making fast predictions.

Appendix A Table of All Case Parameters

Case	Layers	Power	Velocity	Hatch Spacing	Scan Length	Scan Width	Domain Length	Domain Width	Base Height	X Offset	Y Offset	Layer Height	Rotation
P90V95	1	90	95	50	1500	950	360	260	21	30	30	N/A	N/A
P90V105	1	90	105	50	1500	950	350	250	21	30	30	N/A	N/A
P100V90	1	100	90	50	1500	950	350	250	21	30	30	N/A	N/A
P100V100Case1	1	100	100	50	1500	950	350	250	21	30	30	N/A	N/A
P100V100Case2	1	100	100	50	1500	950	340	245	21	30	30	N/A	N/A
P95V95	1	95	95	50	1500	950	350	250	21	30	30	N/A	N/A
P105V90	1	105	90	50	1500	950	350	250	21	30	30	N/A	N/A
P105V95	1	105	95	50	1500	950	350	250	21	30	30	N/A	N/A
P105V105	1	105	105	50	1500	950	350	250	21	30	30	N/A	N/A
P124V109Case1	1	124	109	65	669	1430	204	346	34	30	30	N/A	N/A
P124V109Case2	4	124	109	65	669	1430	234	346	34	30	30	20	67
P75V120	3	75	120	25	1007	1482	240	340	32	20	20	15	67
P90V110	10	90	110	50	1000	1000	260	260	30	30	30	25	67
P250V75	30	250	75	110	1000	1000	260	260	30	30	30	40	67
P200V70	50	200	70	100	500	500	180	180	30	40	40	40	67
P100V100LH15Case1	5	100	100	50	950	1500	250	350	21	30	30	15	67
P100V100LH15Case2	15	100	100	50	500	500	160	160	21	30	30	15	67
P95V97.5	1	95	97.5	50	950	1500	250	360	21	30	30	N/A	N/A
P102.5V92.5	1	102.5	92.5	50	1500	950	360	250	21	30	30	N/A	N/A
P130V110	5	130	110	50	500	500	160	160	21	30	30	20	67
P135V112	5	135	112	50	500	500	160	160	21	30	30	20	67
P140V115	5	140	115	50	500	500	160	160	21	30	30	20	67
P100V100H45	1	100	100	45	500	500	160	160	21	30	30	N/A	N/A
P100V100H40	1	100	100	40	500	500	160	160	21	30	30	N/A	N/A
P225V72.5	30	225	72.5	105	525	525	195	195	25	45	45	35	67
P90V110R45	10	90	110	50	1000	1000	260	260	30	30	30	25	45
P76V104R90	40	76	104	50	650	650	210	210	40	40	40	25	90
P80V130	60	80	130	25	650	650	220	220	26	45	45	15	67

Appendix B Table of Hyperparameter Optimization Options

Parameter	Initial value	Minimum	Maximum	Step	Result
Layer 1 depth	3	1	6	1	1
Layer 2 depth	3	0	6	1	1
Layer 3 depth	9	0	11	1	3
Layer 4 depth	3	0	6	1	0
Layer 1 Kernel Size	7	1	4	1	3
Layer 2 Kernel Size	7	1	4	1	3
Layer 3 Kernel Size	7	1	4	1	4
Layer 4 Kernel Size	7	1	4	1	N/A
Layer 1 Number of Channels	96	16	400	8	64
Layer 2 Number of Channels	192	32	352	8	96
Layer 3 Number of Channels	384	64	560	8	304
Layer 4 Number of Channels	768	192	896	8	N/A
Layer 1 Scale	4	1	11	1	7
Layer 2 Scale	4	1	11	1	8
Layer 3 Scale	4	1	11	1	1
Layer 4 Scale	4	1	11	1	N/A

References

[1] Kalpakjian, S., and Schmid, S., 2013, *Manufacturing Engineering & Technology*, Pearson Higher Ed.

[2] Swainson, W. K., 1977, "Method, Medium and Apparatus for Producing Three-Dimensional Figure Product." [Online]. Available: https://patents.google.com/patent/US4041476A/en. [Accessed: 30-Sept-2024].

[3] John, M. O., 1956, "Photo-Glyph Recording." [Online]. Available: https://patents.google.com/patent/US2775758A/en. [Accessed: 16-Dec-2025].

[4] Pollack, S., Venkatesh, C., Neff, M., Healy, A. V., Hu, G., Fuenmayor, E. A., Lyons, J. G., Major, I., and Devine, D. M., 2019, "Polymer-Based Additive Manufacturing: Historical Developments, Process Types and Material Considerations," *Polymer-Based Additive Manufacturing: Biomedical Applications*, D.M. Devine, ed., Springer International Publishing, Cham, pp. 1–22. https://doi.org/10.1007/978-3-030-24532-0_1.

[5] Leinster, M., 1945, "Things Pass By," Thrilling Wonder Stories, **27**(02), pp. 11–35.

[6] Tempie, W. F., 1939, "The 4 Sided Triangle," Amazing Stories, **13**(11), pp. 6–22.

[7] Wong, K. V., and Hernandez, A., 2012, "A Review of Additive Manufacturing," International Scholarly Research Notices, **2012**(1), p. 208760. https://doi.org/10.5402/2012/208760.

[8] Jiménez, M., Romero, L., Domínguez, I. A., Espinosa, M. del M., and Domínguez, M., 2019, "Additive Manufacturing Technologies: An Overview about 3D Printing Methods and Future Prospects," Complexity, **2019**(1), p. 9656938. https://doi.org/10.1155/2019/9656938.

[9] Shusteff, M., Browar, A. E. M., Kelly, B. E., Henriksson, J., Weisgraber, T. H., Panas, R. M., Fang, N. X., and Spadaccini, C. M., 2017, "One-Step Volumetric Additive Manufacturing of Complex Polymer Structures," Science Advances, **3**(12), p. eaao5496. https://doi.org/10.1126/sciadv.aao5496.

[10] Shusteff, M., Panas, R. M., Henriksson, J., Kelly, B. E., and Browar, A. E. M., 2016, "Additive Fabrication of 3D Structures by Holographic Lithography." [Online]. Available: https://hdl.handle.net/2152/89665. [Accessed: 11-Sept-2024].

[11] Kelly, B. E., Bhattacharya, I., Heidari, H., Shusteff, M., Spadaccini, C. M., and Taylor, H. K., 2019, "Volumetric Additive Manufacturing via Tomographic Reconstruction," Science, **363**(6431), pp. 1075–1079. https://doi.org/10.1126/science.aau7114.

[12] Ding, D., Pan, Z., Cuiuri, D., and Li, H., 2015, "Wire-Feed Additive Manufacturing of Metal Components: Technologies, Developments and Future Interests," Int J Adv Manuf Technol, **81**(1), pp. 465–481. https://doi.org/10.1007/s00170-015-7077-3.

[13] "3D Printing Guide: Types of 3D Printers, Materials, and Applications | Formlabs." [Online]. Available: https://formlabs.com/3d-printers/. [Accessed: 30-Sept-2024].

[14] 2024, "SpaceX Debuts Raptor 3 Engine, Further Enhanced with Metal AM," Metal Additive Manufacturing. [Online]. Available: https://www.metal-am.com/spacex-debuts-raptor-3-engine-further-enhanced-with-metal-additive-manufacturing/. [Accessed: 03-Sept-2024].

[15] 2024, "LEAP 71 Hot Fires Additively Manufactured Rocket Engine Designed without Human Intervention," Metal Additive Manufacturing. [Online]. Available: https://www.metal-am.com/leap-71-hot-fires-additively-manufactured-rocket-engine-designed-without-human-intervention/. [Accessed: 13-Sept-2024].

[16] "Additive Manufacturing Subtracts from Rocket Build Time | NASA Spinoff." [Online]. Available: https://spinoff.nasa.gov/Additive_Manufacturing_Subtracts_from_Rocket_Build_Time. [Accessed: 13-Sept-2024].

[17] DebRoy, T., Wei, H. L., Zuback, J. S., Mukherjee, T., Elmer, J. W., Milewski, J. O., Beese, A. M., Wilson-Heid, A., De, A., and Zhang, W., 2018, "Additive Manufacturing of Metallic Components –

Process, Structure and Properties," Progress in Materials Science, **92**, pp. 112–224. https://doi.org/10.1016/j.pmatsci.2017.10.001.
[18] Zwawi, M., 2022, "Recent Advances in Bio-Medical Implants; Mechanical Properties, Surface Modifications and Applications," Eng. Res. Express, **4**(3), p. 032003. https://doi.org/10.1088/2631-8695/ac8ae2.
[19] Arivazhagan, A., Mani, K., Kamarajan, B. P., A g s, S. A., S, V., Riju, A. A., and G, R., 2025, "Mechanical and Biocompatibility Studies on Additively Manufactured Ti6Al4V Porous Structures Infiltrated with Hydroxyapatite for Implant Applications," Journal of Alloys and Compounds, **1010**, p. 177966. https://doi.org/10.1016/j.jallcom.2024.177966.
[20] Thompson, M. K., Moroni, G., Vaneker, T., Fadel, G., Campbell, R. I., Gibson, I., Bernard, A., Schulz, J., Graf, P., Ahuja, B., and Martina, F., 2016, "Design for Additive Manufacturing: Trends, Opportunities, Considerations, and Constraints," CIRP Annals, **65**(2), pp. 737–760. https://doi.org/10.1016/j.cirp.2016.05.004.
[21] Gao, W., Zhang, Y., Ramanujan, D., Ramani, K., Chen, Y., Williams, C. B., Wang, C. C. L., Shin, Y. C., Zhang, S., and Zavattieri, P. D., 2015, "The Status, Challenges, and Future of Additive Manufacturing in Engineering," Computer-Aided Design, **69**, pp. 65–89. https://doi.org/10.1016/j.cad.2015.04.001.
[22] Duran, P., 2025, "Aconity3D Successfully Completes First Hot-Fire Test of LEAP 71's AI-Designed Cryogenic Aerospike Engine," 3D Printing Industry. [Online]. Available: https://3dprintingindustry.com/news/aconity3d-successfully-completes-first-hot-fire-test-of-leap-71s-ai-designed-cryogenic-aerospike-engine-242695/. [Accessed: 13-Jan-2026].
[23] Roehling, J. D., Khairallah, S. A., Shen, Y., Bayramian, A., Boley, C. D., Rubenchik, A. M., DeMuth, J., Duanmu, N., and Matthews, M. J., 2021, "Physics of Large-Area Pulsed Laser Powder Bed Fusion," Additive Manufacturing, **46**, p. 102186. https://doi.org/10.1016/j.addma.2021.102186.
[24] Materialgeeza, *SLS System Schematic*. [Online]. Available: https://commons.wikimedia.org/wiki/File:Selective_laser_melting_system_schematic.jpg. [Accessed: 19-Dec-2025].
[25] "Deed - Attribution-ShareAlike 3.0 Unported - Creative Commons." [Online]. Available: https://creativecommons.org/licenses/by-sa/3.0/deed.en. [Accessed: 13-Jan-2026].
[26] Heo, T. W., Khairallah, S. A., Shi, R., Berry, J., Perron, A., Calta, N. P., Martin, A. A., Barton, N. R., Roehling, J., Roehling, T., Fattebert, J.-L., Anderson, A., Nichols, A. L., Wopschall, S., King, W. E., McKeown, J. T., and Matthews, M. J., 2021, "A Mesoscopic Digital Twin That Bridges Length and Time Scales for Control of Additively Manufactured Metal Microstructures," J. Phys. Mater., **4**(3), p. 034012. https://doi.org/10.1088/2515-7639/abeef8.
[27] Hashemi, S. M., Parvizi, S., Baghbanijavid, H., Tan, A. T. L., Nematollahi, M., Ramazani, A., Fang, N. X., and Elahinia, M., 2021, "Computational Modelling of Process–Structure–Property–Performance Relationships in Metal Additive Manufacturing: A Review," International Materials Reviews, **0**(0), pp. 1–46. https://doi.org/10.1080/09506608.2020.1868889.
[28] Xu, X., Zhang, J., OUTEIRO, J., Xu, B., and ZHAO, W., 2020, "Multiscale Simulation of Grain Refinement Induced by Dynamic Recrystallization of Ti6Al4V Alloy during High Speed Machining," Journal of Materials Processing Technology, **286**(116834), pp. 1–16. https://doi.org/10.1016/j.jmatprotec.2020.116834.
[29] Stengel, C., Zielinski, J., and Schleifenbaum, J.-H., 2022, "The Effect of Polarized Laser Radiation on the Effective Absorption and Melt Pool Characteristics during the LPBF Process," Procedia CIRP, **111**, pp. 125–129. https://doi.org/10.1016/j.procir.2022.08.069.
[30] Yadroitsev, I., and Yadroitsava, I., 2015, "Evaluation of Residual Stress in Stainless Steel 316L and Ti6Al4V Samples Produced by Selective Laser Melting," Virtual and Physical Prototyping, **10**(2), pp. 67–76. https://doi.org/10.1080/17452759.2015.1026045.

[31] Levkulich, N. C., Semiatin, S. L., Gockel, J. E., Middendorf, J. R., DeWald, A. T., and Klingbeil, N. W., 2019, "The Effect of Process Parameters on Residual Stress Evolution and Distortion in the Laser Powder Bed Fusion of Ti-6Al-4V," Additive Manufacturing, **28**, pp. 475–484. https://doi.org/10.1016/j.addma.2019.05.015.
[32] Mishurova, T., Artzt, K., Rehmer, B., Haubrich, J., Ávila, L., Schoenstein, F., Serrano-Munoz, I., Requena, G., and Bruno, G., 2021, "Separation of the Impact of Residual Stress and Microstructure on the Fatigue Performance of LPBF Ti-6Al-4V at Elevated Temperature," International Journal of Fatigue, **148**, p. 106239. https://doi.org/10.1016/j.ijfatigue.2021.106239.
[33] Smudde, C. M., D'Elia, C. R., San Marchi, C. W., Hill, M. R., and Gibeling, J. C., 2022, "The Influence of Residual Stress on Fatigue Crack Growth Rates of Additively Manufactured Type 304L Stainless Steel," International Journal of Fatigue, **162**, p. 106954. https://doi.org/10.1016/j.ijfatigue.2022.106954.
[34] Gockel, J., Sheridan, L., Koerper, B., and Whip, B., 2019, "The Influence of Additive Manufacturing Processing Parameters on Surface Roughness and Fatigue Life," International Journal of Fatigue, **124**, pp. 380–388. https://doi.org/10.1016/j.ijfatigue.2019.03.025.
[35] Dzogbewu, T. C., and du Preez, W. B., 2021, "Additive Manufacturing of Ti-Based Intermetallic Alloys: A Review and Conceptualization of a Next-Generation Machine," Materials, **14**(15), p. 4317. https://doi.org/10.3390/ma14154317.
[36] Cuellar, S. D., Hill, M. R., DeWald, A. T., and Rankin, J. E., 2012, "Residual Stress and Fatigue Life in Laser Shock Peened Open Hole Samples," International Journal of Fatigue, **44**, pp. 8–13. https://doi.org/10.1016/j.ijfatigue.2012.06.011.
[37] Wang, Z., Palmer, T. A., and Beese, A. M., 2016, "Effect of Processing Parameters on Microstructure and Tensile Properties of Austenitic Stainless Steel 304L Made by Directed Energy Deposition Additive Manufacturing," Acta Materialia, **110**, pp. 226–235. https://doi.org/10.1016/j.actamat.2016.03.019.
[38] Dryepondt, S., Nandwana, P., Fernandez-Zelaia, P., and List, F., 2021, "Microstructure and High Temperature Tensile Properties of 316L Fabricated by Laser Powder-Bed Fusion," Additive Manufacturing, **37**, p. 101723. https://doi.org/10.1016/j.addma.2020.101723.
[39] Starr, T. L., Rafi, K., Stucker, B., and Scherzer, C. M., "CONTROLLING PHASE COMPOSITION IN SELECTIVE LASER MELTED STAINLESS STEELS," p. 8.
[40] Bandyopadhyay, R., Prithivirajan, V., Peralta, A. D., and Sangid, M. D., 2020, "Microstructure-Sensitive Critical Plastic Strain Energy Density Criterion for Fatigue Life Prediction across Various Loading Regimes," Proc Math Phys Eng Sci, **476**(2236), p. 20190766. https://doi.org/10.1098/rspa.2019.0766.
[41] Lekakh, S. N., O'malley, R., Emmendorfer, M., and Hrebec, B., 2017, "Control of Columnar to Equiaxed Transition in Solidification Macrostructure of Austenitic Stainless Steel Castings," ISIJ International, **57**(5), pp. 824–832. https://doi.org/10.2355/isijinternational.ISIJINT-2016-684.
[42] Kobryn, P. A., and Semiatin, S. L., 2001, "The Laser Additive Manufacture of Ti-6Al-4V," JOM, **53**(9), pp. 40–42. https://doi.org/10.1007/s11837-001-0068-x.
[43] Liverani, E., Toschi, S., Ceschini, L., and Fortunato, A., 2017, "Effect of Selective Laser Melting (SLM) Process Parameters on Microstructure and Mechanical Properties of 316L Austenitic Stainless Steel," Journal of Materials Processing Technology, **249**, pp. 255–263. https://doi.org/10.1016/j.jmatprotec.2017.05.042.
[44] Raghavan, N., Stump, B. C., Fernandez-Zelaia, P., Kirka, M. M., and Simunovic, S., 2021, "Influence of Geometry on Columnar to Equiaxed Transition during Electron Beam Powder Bed Fusion of IN718," Additive Manufacturing, **47**, p. 102209. https://doi.org/10.1016/j.addma.2021.102209.

[45] Boissier, M., Allaire, G., and Tournier, C., 2020, “Additive Manufacturing Scanning Paths Optimization Using Shape Optimization Tools,” Struct Multidisc Optim, **61**(6), pp. 2437–2466. https://doi.org/10.1007/s00158-020-02614-3.
[46] Sargent, N., Jones, M., Otis, R., Shapiro, A. A., Delplanque, J.-P., and Xiong, W., 2021, “Integration of Processing and Microstructure Models for Non-Equilibrium Solidification in Additive Manufacturing,” Metals, **11**(4), p. 570. https://doi.org/10.3390/met11040570.
[47] Liu, Z., Ji, J., Wang, Q., Guan, X., Wang, L., and Zhan, X., 2024, “Effect of Scanning Strategy on Microstructure Evolution and Stress Control in Laser Cladding Repair of Ti6Al4V Alloy,” Metals, **14**(7), p. 805. https://doi.org/10.3390/met14070805.
[48] Akram, J., Chalavadi, P., Pal, D., and Stucker, B., 2018, “Understanding Grain Evolution in Additive Manufacturing through Modeling,” Additive Manufacturing, **21**, pp. 255–268. https://doi.org/10.1016/j.addma.2018.03.021.
[49] Carter, L. N., Martin, C., Withers, P. J., and Attallah, M. M., 2014, “The Influence of the Laser Scan Strategy on Grain Structure and Cracking Behaviour in SLM Powder-Bed Fabricated Nickel Superalloy,” Journal of Alloys and Compounds, **615**, pp. 338–347. https://doi.org/10.1016/j.jallcom.2014.06.172.
[50] Saunders, R., Butler, C., Michopoulos, J., Lagoudas, D., Elwany, A., and Bagchi, A., 2021, “Mechanical Behavior Predictions of Additively Manufactured Microstructures Using Functional Gaussian Process Surrogates,” npj Comput Mater, **7**(1), pp. 1–11. https://doi.org/10.1038/s41524-021-00548-y.
[51] Matthews, M. J., Roehling, T. T., Khairallah, S. A., Guss, G., Wu, S. Q., Crumb, M. F., Roehling, J. D., and McKeown, J. T., 2018, “Spatial Modulation of Laser Sources for Microstructural Control of Additively Manufactured Metals,” Procedia CIRP, **74**, pp. 607–610. https://doi.org/10.1016/j.procir.2018.08.077.
[52] Dehoff, R. R., Kirka, M. M., Sames, W. J., Bilheux, H., Tremsin, A. S., Lowe, L. E., and Babu, S. S., 2015, “Site Specific Control of Crystallographic Grain Orientation through Electron Beam Additive Manufacturing,” Materials Science and Technology, **31**(8), pp. 931–938. https://doi.org/10.1179/1743284714Y.0000000734.
[53] Sofinowski, K., Wittwer, M., and Seita, M., 2022, “Encoding Data into Metal Alloys Using Laser Powder Bed Fusion,” Additive Manufacturing, **52**, p. 102683. https://doi.org/10.1016/j.addma.2022.102683.
[54] Lara, A. T., 2025, “Apple Details 3D Printing for Titanium Watch Enclosures,” 3D Printing Industry. [Online]. Available: https://3dprintingindustry.com/news/apple-details-3d-printing-for-titanium-watch-enclosures-246707/. [Accessed: 02-Dec-2025].
[55] “Mapping the Future with 3D-Printed Titanium Apple Watch Cases,” Apple Newsroom. [Online]. Available: https://www.apple.com/newsroom/2025/11/mapping-the-future-with-3d-printed-titanium-apple-watch-cases/. [Accessed: 02-Dec-2025].
[56] Shaikhnag, A., 2025, “Apple Debuts New iPhone Air and Watch Series 11 with 3D Printed Parts,” 3D Printing Industry. [Online]. Available: https://3dprintingindustry.com/news/apple-debuts-new-iphone-air-and-watch-series-11-with-3d-printed-parts-244339/. [Accessed: 02-Dec-2025].
[57] “The Role of Additive Manufacturing in the Era of Industry 4.0 - ScienceDirect.” [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2351978917303529?via%3Dihub. [Accessed: 22-Sept-2024].
[58] Pandita, P., Ghosh, S., Gupta, V. K., Meshkov, A., and Wang, L., 2021, “Application of Deep Transfer Learning and Uncertainty Quantification for Process Identification in Powder Bed Fusion,” ASCE-ASME J Risk and Uncert in Engrg Sys Part B Mech Engrg, **8**(1). https://doi.org/10.1115/1.4051748.

[59] Moges, T., Ameta, G., and Witherell, P., 2019, "A Review of Model Inaccuracy and Parameter Uncertainty in Laser Powder Bed Fusion Models and Simulations," J. Manuf. Sci. Eng, **141**(040801). https://doi.org/10.1115/1.4042789.
[60] Nath, P., 2020, "Uncertainty Quantification and Management in Additive Manufacturing." [Online]. Available: https://ir.vanderbilt.edu/handle/1803/15357. [Accessed: 30-Nov-2021].
[61] Cordero, Z. C., Knight, B. E., and Schuh, C. A., 2016, "Six Decades of the Hall–Petch Effect – a Survey of Grain-Size Strengthening Studies on Pure Metals," International Materials Reviews, **61**(8), pp. 495–512. https://doi.org/10.1080/09506608.2016.1191808.
[62] Li, J., Cao, Y., Gao, B., Li, Y., and Zhu, Y., 2018, "Superior Strength and Ductility of 316L Stainless Steel with Heterogeneous Lamella Structure," J Mater Sci, **53**(14), pp. 10442–10456. https://doi.org/10.1007/s10853-018-2322-4.
[63] Leuders, S., Thöne, M., Riemer, A., Niendorf, T., Tröster, T., Richard, H. A., and Maier, H. J., 2013, "On the Mechanical Behaviour of Titanium Alloy TiAl6V4 Manufactured by Selective Laser Melting: Fatigue Resistance and Crack Growth Performance," International Journal of Fatigue, **48**, pp. 300–307. https://doi.org/10.1016/j.ijfatigue.2012.11.011.
[64] Thijs, L., Montero Sistiaga, M. L., Wauthle, R., Xie, Q., Kruth, J.-P., and Van Humbeeck, J., 2013, "Strong Morphological and Crystallographic Texture and Resulting Yield Strength Anisotropy in Selective Laser Melted Tantalum," Acta Materialia, **61**(12), pp. 4657–4668. https://doi.org/10.1016/j.actamat.2013.04.036.
[65] Tiley, J. S., Kim, S. L., Parthasarathy, T. A., Loughnane, G. T., Kublik, R., and Salem, A. A., 2017, "Quantifying the Effect of Microstructure Variability on the Yield Strength Predictions of Ni-Base Superalloys," Materials Science and Engineering: A, **685**, pp. 178–186. https://doi.org/10.1016/j.msea.2016.12.068.
[66] Gunasegaram, D. R., and Steinbach, I., 2021, "Modelling of Microstructure Formation in Metal Additive Manufacturing: Recent Progress, Research Gaps and Perspectives," Metals, **11**(9), p. 1425. https://doi.org/10.3390/met11091425.
[67] Matthews, M. J., Roehling, T. T., Khairallah, S. A., Tumkur, T. U., Guss, G., Shi, R., Roehling, J. D., Smith, W. L., Vrancken, B. K., Ganeriwala, R. K., and McKeown, J. T., 2020, "Controlling Melt Pool Shape, Microstructure and Residual Stress in Additively Manufactured Metals Using Modified Laser Beam Profiles," Procedia CIRP, **94**, pp. 200–204. https://doi.org/10.1016/j.procir.2020.09.038.
[68] Roehling, T. T., Wu, S. S. Q., Khairallah, S. A., Roehling, J. D., Soezeri, S. S., Crumb, M. F., and Matthews, M. J., 2017, "Modulating Laser Intensity Profile Ellipticity for Microstructural Control during Metal Additive Manufacturing," Acta Materialia, **128**, pp. 197–206. https://doi.org/10.1016/j.actamat.2017.02.025.
[69] Gockel, J., and Beuth, J., 2013, "Understanding Ti-6Al-4V Microstructure Control in Additive Manufacturing via Process Maps." [Online]. Available: https://hdl.handle.net/2152/88657. [Accessed: 19-Dec-2025].
[70] Shi, R., Khairallah, S. A., Roehling, T. T., Heo, T. W., McKeown, J. T., and Matthews, M. J., 2020, "Microstructural Control in Metal Laser Powder Bed Fusion Additive Manufacturing Using Laser Beam Shaping Strategy," Acta Materialia, **184**, pp. 284–305. https://doi.org/10.1016/j.actamat.2019.11.053.
[71] "The Rapid Solidification Processing of Materials: Science, Principles, Technology, Advances, and Applications | Journal of Materials Science." [Online]. Available: https://link.springer.com/article/10.1007/s10853-009-3995-5. [Accessed: 21-Sept-2024].
[72] Liang, Y.-J., Cheng, X., and Wang, H.-M., 2016, "A New Microsegregation Model for Rapid Solidification Multicomponent Alloys and Its Application to Single-Crystal Nickel-Base Superalloys of Laser Rapid Directional Solidification," Acta Materialia, **118**, pp. 17–27. https://doi.org/10.1016/j.actamat.2016.07.008.

[73] Liu, P., Wang, Z., Xiao, Y., Horstemeyer, M. F., Cui, X., and Chen, L., 2019, "Insight into the Mechanisms of Columnar to Equiaxed Grain Transition during Metallic Additive Manufacturing," Additive Manufacturing, **26**, pp. 22–29. https://doi.org/10.1016/j.addma.2018.12.019.
[74] Bermingham, M. J., StJohn, D. H., Krynen, J., Tedman-Jones, S., and Dargusch, M. S., 2019, "Promoting the Columnar to Equiaxed Transition and Grain Refinement of Titanium Alloys during Additive Manufacturing," Acta Materialia, **168**, pp. 261–274. https://doi.org/10.1016/j.actamat.2019.02.020.
[75] Liu, W., and DuPont, J. N., 2004, "Effects of Melt-Pool Geometry on Crystal Growth and Microstructure Development in Laser Surface-Melted Superalloy Single Crystals: Mathematical Modeling of Single-Crystal Growth in a Melt Pool (Part I)," Acta Materialia, **52**(16), pp. 4833–4847. https://doi.org/10.1016/j.actamat.2004.06.041.
[76] Andreau, O., Koutiri, I., Peyre, P., Penot, J.-D., Saintier, N., Pessard, E., De Terris, T., Dupuy, C., and Baudin, T., 2019, "Texture Control of 316L Parts by Modulation of the Melt Pool Morphology in Selective Laser Melting," Journal of Materials Processing Technology, **264**, pp. 21–31. https://doi.org/10.1016/j.jmatprotec.2018.08.049.
[77] Sharifi, P., Jamali, J., Sadayappan, K., and Wood, J. T., 2018, "Grain Size Distribution and Interfacial Heat Transfer Coefficient during Solidification of Magnesium Alloys Using High Pressure Die Casting Process," Journal of Materials Sciences and Technology, **34**(2), p. 324. https://doi.org/10.1016/j.jmst.2016.09.004.
[78] Chan, K. S., 2020, "Mechanism-Based Models for Predicting the Microstructure and Stress–Strain Response of Additively Manufactured Superalloy 718Plus," J. of Materi Eng and Perform, **29**(3), pp. 2035–2045. https://doi.org/10.1007/s11665-020-04678-0.
[79] Khairallah, S. A., Anderson, A. T., Rubenchik, A., and King, W. E., 2016, "Laser Powder-Bed Fusion Additive Manufacturing: Physics of Complex Melt Flow and Formation Mechanisms of Pores, Spatter, and Denudation Zones," Acta Materialia, **108**, pp. 36–45. https://doi.org/10.1016/j.actamat.2016.02.014.
[80] King, W. E., Anderson, A. T., Ferencz, R. M., Hodge, N. E., Kamath, C., Khairallah, S. A., and Rubenchik, A. M., 2015, "Laser Powder Bed Fusion Additive Manufacturing of Metals; Physics, Computational, and Materials Challenges," Applied Physics Reviews, **2**(4), p. 041304. https://doi.org/10.1063/1.4937809.
[81] Khairallah, S. A., Anderson, A., Rubenchik, A. M., Florando, J., Wu, S., and Lowdermilk, H., 2015, "Simulation of the Main Physical Processes in Remote Laser Penetration with Large Laser Spot Size," AIP Advances, **5**(4), p. 047120. https://doi.org/10.1063/1.4918284.
[82] Hodge, N. E., Ferencz, R. M., and Solberg, J. M., 2014, "Implementation of a Thermomechanical Model for the Simulation of Selective Laser Melting," Comput Mech, **54**(1), pp. 33–51. https://doi.org/10.1007/s00466-014-1024-2.
[83] Khairallah, S. A., Martin, A. A., Lee, J. R. I., Guss, G., Calta, N. P., Hammons, J. A., Nielsen, M. H., Chaput, K., Schwalbach, E., Shah, M. N., Chapman, M. G., Willey, T. M., Rubenchik, A. M., Anderson, A. T., Wang, Y. M., Matthews, M. J., and King, W. E., 2020, "Controlling Interdependent Meso-Nanosecond Dynamics and Defect Generation in Metal 3D Printing," Science, **368**(6491), pp. 660–665. https://doi.org/10.1126/science.aay7830.
[84] Yang, J., Schlenger, L. M., Nasab, M. H., Van Petegem, S., Marone, F., Logé, R. E., and Leinenbach, C., 2024, "Experimental Quantification of Inward Marangoni Convection and Its Impact on Keyhole Threshold in Laser Powder Bed Fusion of Stainless Steel," Additive Manufacturing, **84**, p. 104092. https://doi.org/10.1016/j.addma.2024.104092.
[85] Cunningham, R., Zhao, C., Parab, N., Kantzos, C., Pauza, J., Fezzaa, K., Sun, T., and Rollett, A. D., 2019, "Keyhole Threshold and Morphology in Laser Melting Revealed by Ultrahigh-Speed x-Ray Imaging," Science, **363**(6429), pp. 849–852. https://doi.org/10.1126/science.aav4687.

[86] Wolfer, A. J., 2020, “Physics-Based Surrogate Modeling of Laser Powder Bed Fusion Additive Manufacturing,” Ph.D., University of California, Davis. [Online]. Available: https://www.proquest.com/docview/2503475225/abstract/E4D17D5F6A484678PQ/1. [Accessed: 05-Dec-2021].
[87] Ridolfi, M. R., Folgarait, P., and Di Schino, A., 2020, “Modelling Selective Laser Melting of Metallic Powders,” Metallurgist, **64**(5), pp. 588–600. https://doi.org/10.1007/s11015-020-01031-7.
[88] Li, Y., Mojumder, S., Lu, Y., Amin, A. A., Guo, J., Xie, X., Chen, W., Wagner, G. J., Cao, J., and Liu, W. K., 2024, “Statistical Parameterized Physics-Based Machine Learning Digital Shadow Models for Laser Powder Bed Fusion Process,” Additive Manufacturing, **87**, p. 104214. https://doi.org/10.1016/j.addma.2024.104214.
[89] Teferra, K., and Rowenhorst, D. J., 2021, “Optimizing the Cellular Automata Finite Element Model for Additive Manufacturing to Simulate Large Microstructures,” Acta Materialia, **213**, p. 116930. https://doi.org/10.1016/j.actamat.2021.116930.
[90] Sweidan, F. B., and Ryu, H. J., 2021, “Kinetic Monte Carlo Simulations of the Sintering Microstructural Evolution in Density Graded Stainless Steel Fabricated by SPS,” Materials Today Communications, **26**, p. 101863. https://doi.org/10.1016/j.mtcomm.2020.101863.
[91] Saluja, R. S., Ganesh Narayanan, R., and Das, S., 2012, “Cellular Automata Finite Element (CAFE) Model to Predict the Forming of Friction Stir Welded Blanks,” Computational Materials Science, **58**, pp. 87–100. https://doi.org/10.1016/j.commatsci.2012.01.036.
[92] Baumard, A., Ayrault, D., Fandeur, O., Bordreuil, C., and Deschaux-Beaume, F., 2021, “Numerical Prediction of Grain Structure Formation during Laser Powder Bed Fusion of 316 L Stainless Steel,” Materials & Design, **199**, p. 109434. https://doi.org/10.1016/j.matdes.2020.109434.
[93] Rodgers, T. M., Moser, D., Abdeljawad, F., Jackson, O. D. U., Carroll, J. D., Jared, B. H., Bolintineanu, D. S., Mitchell, J. A., and Madison, J. D., 2021, “Simulation of Powder Bed Metal Additive Manufacturing Microstructures with Coupled Finite Difference-Monte Carlo Method,” Additive Manufacturing, **41**, p. 101953. https://doi.org/10.1016/j.addma.2021.101953.
[94] Li, W., and Soshi, M., 2019, “Modeling Analysis of the Effect of Laser Transverse Speed on Grain Morphology during Directed Energy Deposition Process,” Int J Adv Manuf Technol, **103**(9), pp. 3279–3291. https://doi.org/10.1007/s00170-019-03690-6.
[95] Li, W., and Soshi, M., 2019, “Modeling Analysis of Grain Morphologies in Directed Energy Deposition (DED) Coating with Different Laser Scanning Patterns,” Materials Letters, **251**, pp. 8–12. https://doi.org/10.1016/j.matlet.2019.05.027.
[96] Ma, L., Liu, H., Williams, M., Levine, L., and Ramazani, A., 2024, “Integrating Phase Field Modeling and Machine Learning to Develop Process-Microstructure Relationships in Laser Powder Bed Fusion of IN718,” Metallogr. Microstruct. Anal., **13**(5), pp. 983–995. https://doi.org/10.1007/s13632-024-01130-w.
[97] Sieradzki, L., and Madej, L., 2013, “A Perceptive Comparison of the Cellular Automata and Monte Carlo Techniques in Application to Static Recrystallization Modeling in Polycrystalline Materials,” Computational Materials Science, **67**, pp. 156–173. https://doi.org/10.1016/j.commatsci.2012.08.047.
[98] Gäumann, M., Bezençon, C., Canalis, P., and Kurz, W., 2001, “Single-Crystal Laser Deposition of Superalloys: Processing–Microstructure Maps,” Acta Materialia, **49**(6), pp. 1051–1062. https://doi.org/10.1016/S1359-6454(00)00367-0.
[99] Anderson, T. D., DuPont, J. N., and DebRoy, T., 2010, “Origin of Stray Grain Formation in Single-Crystal Superalloy Weld Pools from Heat Transfer and Fluid Flow Modeling,” Acta Materialia, **58**(4), pp. 1441–1454. https://doi.org/10.1016/j.actamat.2009.10.051.

[100] Oommen, V., Shukla, K., Desai, S., Dingreville, R., and Karniadakis, G. E., 2024, "Rethinking Materials Simulations: Blending Direct Numerical Simulations with Neural Operators," npj Comput Mater, **10**(1), pp. 1–14. https://doi.org/10.1038/s41524-024-01319-1.
[101] Murphy, K. P., 2012, *Machine Learning: A Probabilistic Perspective*, MIT Press.
[102] 2021, "What Are Large Language Models (LLMs)? | IBM." [Online]. Available: https://www.ibm.com/think/topics/large-language-models. [Accessed: 19-Dec-2025].
[103] "Weights & Biases," W&B. [Online]. Available: httpss://wandb.ai/gladiator/LLMs-as-classifiers/reports/LLMs-are-machine-learning-classifiers--VmlldzoxMTEwNzUyNA. [Accessed: 19-Dec-2025].
[104] Schäfer, A. M., and Zimmermann, H. G., 2006, "Recurrent Neural Networks Are Universal Approximators," *Artificial Neural Networks – ICANN 2006*, S.D. Kollias, A. Stafylopatis, W. Duch, and E. Oja, eds., Springer, Berlin, Heidelberg, pp. 632–640. https://doi.org/10.1007/11840817_66.
[105] Sonoda, S., and Murata, N., 2017, "Neural Network with Unbounded Activation Functions Is Universal Approximator," Applied and Computational Harmonic Analysis, **43**(2), pp. 233–268. https://doi.org/10.1016/j.acha.2015.12.005.
[106] Leshno, M., Lin, V. Ya., Pinkus, A., and Schocken, S., 1993, "Multilayer Feedforward Networks with a Nonpolynomial Activation Function Can Approximate Any Function," Neural Networks, **6**(6), pp. 861–867. https://doi.org/10.1016/S0893-6080(05)80131-5.
[107] Tran, D., Ranganath, R., and Blei, D. M., 2016, "The Variational Gaussian Process." https://doi.org/10.48550/arXiv.1511.06499.
[108] Rasmussen, C. E., 1997, "Evaluation of Gaussian Processes and Other Methods for Non-Linear Regression," Thesis. [Online]. Available: https://tspace.library.utoronto.ca/handle/1807/10840. [Accessed: 01-July-2024].
[109] Rosenblatt, F., *PRINCIPLES OF NEURODYNAMICS. PERCEPTRONS AND THE THEORY OF BRAIN MECHANISMS*. https://doi.org/10.21236/AD0256582.
[110] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R., 2014, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," Journal of Machine Learning Research, **15**(56), pp. 1929–1958.
[111] Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P., 1998, "Gradient-Based Learning Applied to Document Recognition," Proceedings of the IEEE, **86**(11), pp. 2278–2324. https://doi.org/10.1109/5.726791.
[112] Krizhevsky, A., Sutskever, I., and Hinton, G. E., 2012, "ImageNet Classification with Deep Convolutional Neural Networks," *Advances in Neural Information Processing Systems*, Curran Associates, Inc. [Online]. Available: https://papers.nips.cc/paper_files/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html. [Accessed: 10-Oct-2024].
[113] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L., 2015, "ImageNet Large Scale Visual Recognition Challenge," Int J Comput Vis, **115**(3), pp. 211–252. https://doi.org/10.1007/s11263-015-0816-y.
[114] Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., and Xie, S., 2022, "A ConvNet for the 2020s." https://doi.org/10.48550/arXiv.2201.03545.
[115] Tolstikhin, I., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., Lucic, M., and Dosovitskiy, A., 2021, "MLP-Mixer: An All-MLP Architecture for Vision." [Online]. Available: http://arxiv.org/abs/2105.01601. [Accessed: 03-Dec-2022].
[116] Mentzer, K. L., and Peterson, J. L., 2023, "Neural Network Surrogate Models for Equations of State," Physics of Plasmas, **30**(3), p. 032704. https://doi.org/10.1063/5.0126708.
[117] Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., Bridgland, A., Meyer, C., Kohl, S. A. A., Ballard, A. J., Cowie, A.,

Romera-Paredes, B., Nikolov, S., Jain, R., Adler, J., Back, T., Petersen, S., Reiman, D., Clancy, E., Zielinski, M., Steinegger, M., Pacholska, M., Berghammer, T., Bodenstein, S., Silver, D., Vinyals, O., Senior, A. W., Kavukcuoglu, K., Kohli, P., and Hassabis, D., 2021, "Highly Accurate Protein Structure Prediction with AlphaFold," Nature, **596**(7873), pp. 583–589. https://doi.org/10.1038/s41586-021-03819-2.

[118] Anandan Kumar, H., Kumaraguru, S., Paul, C., and Bindra, K., 2021, "Faster Temperature Prediction in the Powder Bed Fusion Process through the Development of a Surrogate Model," Optics & Laser Technology, **141**, p. 107122. https://doi.org/10.1016/j.optlastec.2021.107122.

[119] Huang, X., Xie, T., Wang, Z., Chen, L., Zhou, Q., and Hu, Z., 2021, "A Transfer Learning-Based Multi-Fidelity Point-Cloud Neural Network Approach for Melt Pool Modeling in Additive Manufacturing," ASCE-ASME J Risk and Uncert in Engrg Sys Part B Mech Engrg, **8**(1). https://doi.org/10.1115/1.4051749.

[120] Huang, X., Hu, Z., Xie, T., Wang, Z., Chen, L., and Zhou, Q., 2021, "Point-Cloud Neural Network Using Transfer Learning-Based Multi-Fidelity Method for Thermal Field Prediction in Additive Manufacturing," American Society of Mechanical Engineers Digital Collection. https://doi.org/10.1115/DETC2021-67963.

[121] Bhutada, A., Kumar, S., Gunasegaram, D., and Alankar, A., 2021, "Machine Learning Based Methods for Obtaining Correlations between Microstructures and Thermal Stresses," Metals, **11**(8), p. 1167. https://doi.org/10.3390/met11081167.

[122] Boillat, E., Kolossov, S., Glardon, R., Loher, M., Saladin, D., and Levy, G., 2004, "Finite Element and Neural Network Models for Process Optimization in Selective Laser Sintering," Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, **218**(6), pp. 607–614. https://doi.org/10.1243/0954405041167121.

[123] Bauhofer, A., and Daraio, C., 2020, "Neural Networks for Trajectory Evaluation in Direct Laser Writing," Int J Adv Manuf Technol, **107**(5), pp. 2563–2577. https://doi.org/10.1007/s00170-020-05086-3.

[124] Tapia, G., Khairallah, S., Matthews, M., King, W. E., and Elwany, A., 2018, "Gaussian Process-Based Surrogate Modeling Framework for Process Planning in Laser Powder-Bed Fusion Additive Manufacturing of 316L Stainless Steel," Int J Adv Manuf Technol, **94**(9), pp. 3591–3603. https://doi.org/10.1007/s00170-017-1045-z.

[125] Knapp, G. L., Stump, B., Scime, L., Márquez Rossy, A., Joslin, C., Halsey, W., and Plotkowski, A., 2023, "Leveraging the Digital Thread for Physics-Based Prediction of Microstructure Heterogeneity in Additively Manufactured Parts," Additive Manufacturing, **78**, p. 103861. https://doi.org/10.1016/j.addma.2023.103861.

[126] Mitchell, J. A., Abdeljawad, F., Battaile, C., Garcia-Cardona, C., Holm, E. A., Homer, E. R., Madison, J., Rodgers, T. M., Thompson, A. P., Tikare, V., Webb, E., and Plimpton, S. J., 2023, "Parallel Simulation via SPPARKS of On-Lattice Kinetic and Metropolis Monte Carlo Models for Materials Processing," Modelling Simul. Mater. Sci. Eng., **31**(5), p. 055001. https://doi.org/10.1088/1361-651X/accc4b.

[127] Pan, Y., Kuo, K.-S., Rilee, M. L., and Yu, H., 2021, "Assessing Deep Neural Networks as Probability Estimators," *2021 IEEE International Conference on Big Data (Big Data)*, pp. 1083–1091. https://doi.org/10.1109/BigData52589.2021.9671328.

[128] Khoussi, S., Heckert, N. A., Battou, A., and Bensalem, S., 2021, "Neural Networks for Classifying Probability Distributions," NIST. [Online]. Available: https://www.nist.gov/publications/neural-networks-classifying-probability-distributions. [Accessed: 17-Sept-2024].

[129] Clare, M. C. A., Jamil, O., and Morcrette, C. J., 2021, "Combining Distribution-Based Neural Networks to Predict Weather Forecast Probabilities," Quarterly Journal of the Royal Meteorological Society, **147**(741), pp. 4337–4357. https://doi.org/10.1002/qj.4180.

[130] Bouallègue, Z. B., Clare, M. C. A., Magnusson, L., Gascón, E., Maier-Gerber, M., Janoušek, M., Rodwell, M., Pinault, F., Dramsch, J. S., Lang, S. T. K., Raoult, B., Rabier, F., Chevallier, M., Sandu, I., Dueben, P., Chantry, M., and Pappenberger, F., 2024, "The Rise of Data-Driven Weather Forecasting: A First Statistical Assessment of Machine Learning–Based Weather Forecasts in an Operational-Like Context." https://doi.org/10.1175/BAMS-D-23-0162.1.
[131] Opitz, D., and Maclin, R., 1999, "Popular Ensemble Methods: An Empirical Study," jair, **11**, pp. 169–198. https://doi.org/10.1613/jair.614.
[132] Wolfer, A. J., Aires, J., Wheeler, K., Delplanque, J.-P., Rubenchik, A., Anderson, A., and Khairallah, S., 2019, "Fast Solution Strategy for Transient Heat Conduction for Arbitrary Scan Paths in Additive Manufacturing," Additive Manufacturing, **30**, p. 100898. https://doi.org/10.1016/j.addma.2019.100898.
[133] Gao, J., and Thompson, R. G., 1996, "Real Time-Temperature Models for Monte Carlo Simulations of Normal Grain Growth," Acta Materialia, **44**(11), pp. 4565–4570. https://doi.org/10.1016/1359-6454(96)00079-1.
[134] Garcia Cardona, C., Wagner, G. J., Tikare, V., Holm, E. A., Plimpton, S. J., Thompson, A. P., Slepoy, A., Zhou, X. W., Battaile, C. C., and Chandross, M. E., 2009, *Crossing the Mesoscale No-Mans Land via Parallel Kinetic Monte Carlo.*, SAND2009-6226, Sandia National Laboratories (SNL), Albuquerque, NM, and Livermore, CA (United States). https://doi.org/10.2172/966942.
[135] "Trodger-Snl/Spparks at Am_finitedifference." [Online]. Available: https://github.com/trodger-snl/spparks/tree/am_finitedifference. [Accessed: 08-Aug-2024].
[136] 2024, "Off-by-One Error," Wikipedia. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Off-by-one_error&oldid=1256078528. [Accessed: 15-Nov-2024].
[137] "Skimage.Measure — Skimage 0.24.0 Documentation." [Online]. Available: https://scikit-image.org/docs/stable/api/skimage.measure.html#skimage.measure.label. [Accessed: 20-Aug-2024].
[138] Roehling, T. T., Shi, R., Khairallah, S. A., Roehling, J. D., Guss, G. M., McKeown, J. T., and Matthews, M. J., 2020, "Controlling Grain Nucleation and Morphology by Laser Beam Shaping in Metal Additive Manufacturing," Materials & Design, **195**, p. 109071. https://doi.org/10.1016/j.matdes.2020.109071.
[139] "Deed - Attribution-NonCommercial-NoDerivatives 4.0 International - Creative Commons." [Online]. Available: https://creativecommons.org/licenses/by-nc-nd/4.0/. [Accessed: 13-Jan-2026].
[140] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N., 2021, "An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale," arXiv:2010.11929 [cs]. [Online]. Available: http://arxiv.org/abs/2010.11929. [Accessed: 26-Feb-2022].
[141] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B., 2021, "Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows," arXiv:2103.14030 [cs]. [Online]. Available: http://arxiv.org/abs/2103.14030. [Accessed: 26-Feb-2022].
[142] Yao, Z., Cao, Y., Lin, Y., Liu, Z., Zhang, Z., and Hu, H., 2021, "Leveraging Batch Normalization for Vision Transformers," *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pp. 413–422. https://doi.org/10.1109/ICCVW54120.2021.00050.
[143] Ioffe, S., 2017, "Batch Renormalization: Towards Reducing Minibatch Dependence in Batch-Normalized Models." https://doi.org/10.48550/arXiv.1702.03275.
[144] Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. R., 2012, "Improving Neural Networks by Preventing Co-Adaptation of Feature Detectors." https://doi.org/10.48550/arXiv.1207.0580.

[145] Neils, A., Dong, L., and Wadley, H., 2022, "The Small-Scale Limits of Electron Beam Melt Additive Manufactured Ti–6Al–4V Octet-Truss Lattices," AIP Advances, **12**(9), p. 095021. https://doi.org/10.1063/5.0094155.
[146] Soltani, H., Ngomesse, F., Reinhart, G., Benoudia, M. C., Zahzouh, M., and Nguyen-Thi, H., 2021, "Impact of Gravity on Directional Solidification of Refined Al-20wt.%Cu Alloy Investigated by *in Situ* X-Radiography," Journal of Alloys and Compounds, **862**, p. 158028. https://doi.org/10.1016/j.jallcom.2020.158028.
[147] Goins, P. E., Murdoch, H. A., Hernández-Rivera, E., and Tschopp, M. A., 2018, "Effect of Magnetic Fields on Microstructure Evolution," Computational Materials Science, **150**, pp. 464–474. https://doi.org/10.1016/j.commatsci.2018.04.034.
[148] O'Brien, D., McShane, P., and Thornton, S., "The Group of Symmetries of the Cube."
[149] Kukačka, J., Golkov, V., and Cremers, D., 2017, "Regularization for Deep Learning: A Taxonomy." https://doi.org/10.48550/arXiv.1710.10686.
[150] Tian, Y., and Zhang, Y., 2022, "A Comprehensive Survey on Regularization Strategies in Machine Learning," Information Fusion, **80**, pp. 146–166. https://doi.org/10.1016/j.inffus.2021.11.005.
[151] Loshchilov, I., and Hutter, F., 2019, "Decoupled Weight Decay Regularization." https://doi.org/10.48550/arXiv.1711.05101.
[152] "AdamW — PyTorch 2.1 Documentation." [Online]. Available: https://pytorch.org/docs/stable/generated/torch.optim.AdamW.html. [Accessed: 24-Oct-2023].
[153] "CosineAnnealingWarmRestarts — PyTorch 2.10 Documentation." [Online]. Available: https://docs.pytorch.org/docs/stable/generated/torch.optim.lr_scheduler.CosineAnnealingWarmRestarts.html. [Accessed: 18-Mar-2026].
[154] Akiba, T., Sano, S., Yanase, T., Ohta, T., and Koyama, M., 2019, "Optuna: A Next-Generation Hyperparameter Optimization Framework," *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Association for Computing Machinery, New York, NY, USA, pp. 2623–2631. https://doi.org/10.1145/3292500.3330701.
[155] Balandat, M., Karrer, B., Jiang, D. R., Daulton, S., Letham, B., Wilson, A. G., and Bakshy, E., 2020, "BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization." https://doi.org/10.48550/arXiv.1910.06403.
[156] Yang, L., Zhang, Z., Song, Y., Hong, S., Xu, R., Zhao, Y., Zhang, W., Cui, B., and Yang, M.-H., 2023, "Diffusion Models: A Comprehensive Survey of Methods and Applications," ACM Comput. Surv., **56**(4), p. 105:1-105:39. https://doi.org/10.1145/3626235.
[157] Lehtinen, J., Munkberg, J., Hasselgren, J., Laine, S., Karras, T., Aittala, M., and Aila, T., 2018, "Noise2Noise: Learning Image Restoration without Clean Data." https://doi.org/10.48550/arXiv.1803.04189.
[158] Krull, A., Buchholz, T.-O., and Jug, F., 2019, "Noise2Void - Learning Denoising from Single Noisy Images." https://doi.org/10.48550/arXiv.1811.10980.
[159] Laine, S., Karras, T., Lehtinen, J., and Aila, T., 2019, "High-Quality Self-Supervised Deep Image Denoising." https://doi.org/10.48550/arXiv.1901.10277.
[160] Simonyan, K., Vedaldi, A., and Zisserman, A., 2013, "Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps." [Online]. Available: https://openreview.net/forum?id=cO4ycnpqxKcS9. [Accessed: 24-Sept-2025].
[161] Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D., 2020, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization," Int J Comput Vis, **128**(2), pp. 336–359. https://doi.org/10.1007/s11263-019-01228-7.
[162] Schubert, E., and Gertz, M., 2018, "Numerically Stable Parallel Computation of (Co-)Variance," *Proceedings of the 30th International Conference on Scientific and Statistical Database*

Management, Association for Computing Machinery, New York, NY, USA, pp. 1–12. https://doi.org/10.1145/3221269.3223036.
[163] Efanov, A. A., Ivliev, S. A., and Shagraev, A. G., 2021, "Welford's Algorithm for Weighted Statistics," *2021 3rd International Youth Conference on Radio Electronics, Electrical and Power Engineering (REEPE)*, pp. 1–5. https://doi.org/10.1109/REEPE51337.2021.9387973.
[164] Fodor, I. K., 2002, *A Survey of Dimension Reduction Techniques*, UCRL-ID-148494, 15002155. https://doi.org/10.2172/15002155.
[165] Hoover, B. G., and Ornelas-Rascon, C. H., 2025, "Quantitative Large-Area Polarized-Light Microscopy," Micros Today, **33**(1), pp. 26–30. https://doi.org/10.1093/mictod/qaae106.
[166] DeMott, R., Haghdadi, N., Kong, C., Gandomkar, Z., Kenney, M., Collins, P., and Primig, S., 2021, "3D Electron Backscatter Diffraction Characterization of Fine α Titanium Microstructures: Collection, Reconstruction, and Analysis Methods," Ultramicroscopy, **230**, p. 113394. https://doi.org/10.1016/j.ultramic.2021.113394.