Classification of Microstructural Defects in Selective Laser Melted Inconel 713C alloy using Convolutional Neural Networks

Ethan L. Edmunds and Meurig Thomas*

1 Interdisciplinary Programmes in Engineering, University of Sheffield, Mappin Street, Sheffield, S1 3JD, United Kingdom

*Corresponding Author (meurig.thomas@sheffield.ac.uk)

Abstract

Microstructural defects are commonly found in additively manufactured materials and can have significant effects on the material's bulk properties. This warrants defect detection and classification during microstructural evaluation, which is often laborious, costly, and can yield sub-optimal results if done manually. Previous attempts to facilitate automated classification in additively manufactured nickel-alloys have used supervised machine learning methods, such as kth-nearest neighbour classification and decision trees. This study proposes and evaluates the use of convolutional neural networks for microstructural defect classification in selective laser melted Inconel 713C samples. It outlines the process used to create and augment the dataset, as well as hyperparameter selection of the neural network for optimal classification performance.

Keywords: Additive Manufacturing; Nickel Alloys; Defects; Machine Learning; Deep Learning

Introduction

Additive manufacturing (AM) is a class of manufacturing processes, that enables the creation of components, layer-by-layer using a high-energy heat source, from three-dimensional models. The unique characteristics of AM lead to various advantages when compared to traditional manufacturing processes [1], [2], [3]. Despite these advantages, there is a lack of understanding surrounding the "process-microstructure-property" relationship in AM, leading

to lack of opportunities in the design and optimisation of materials and manufacturing processes [4]. In particular, components produced using AM methods are typically prone to the presence of microstructural defects, such as lack of fusion defects, gas porosities, solidification cracking and keyhole collapses, which pose risks in structural applications, especially when under cyclic loading conditions [5], [6]. For these reasons, studying the "process-microstructure-property" relationship is vital to advancing the performance of engineering components. A part of the investigation of these relationships is correctly detecting and classifying microstructural defects from a variety of microstructural images. However, when done manually, the detection and classification of microstructural defects can often be laborious, costly and yield suboptimal results, suggesting the need for an alternative strategy to detection and classification [7].

Machine Learning (ML) is a brand of Artificial Intelligence that offers a solution to data analysis in the field of materials science. It focuses on pattern identification and cognitive acquisition concepts, which optimise performance criteria based on example data or past experiences [8], [9]. In the field of AM, supervised and unsupervised machine learning methods have been used for a variety of microstructural detection and classification applications. For instance, it has been used to classify three types of pores in titanium and nickel-alloys, to classify powder-bed anomalies, and to detect porosities in aluminium alloy thin-walled structures [10], [11], [12]. Deep Learning (DL), a relatively new subset of ML algorithms, have been pivotal in advancing the state-of-the-art across various domains, including speech and visual object detection and classification [13], [14]. DL algorithms are more capable at learning complicated patterns from more complex datasets, leading to their increased use in materials science applications. For instance, it has been used to distinguish between metallic morphologies with dendrites and those without [15]. It has also been used for lath-bainite segmentation in complex-phase steel, achieving an accuracy of 90% which rivals the accuracy of expert segmentations [16]. Deep learning has also found use in the design of new materials and to link microstructural characteristics to the material's bulk properties [17], [18]. For an overview on the previous use of deep learning approaches in microstructural characterisation, the reader is directed to these papers for more information [16], [19].

In this study, convolutional neural networks (CNNs) are used to automatically classify the different types of defects that are present in Inconel 713C alloy.

This paper builds on the work of Aziz et al. who used k-th Nearest Neighbours (kNN) and decision tree algorithms to classify defects found in a variety of nickel-alloys with 89.8% and 90-92% accuracy respectively [20]. It will outline the process to sourcing binary images of individual defects from metallographs of the alloy samples, the preprocessing that the dataset underwent, as well as the methodologies used for CNN optimisation. It will then evaluate the efficacy of neural networks as a method to classifying defects, comparing the network's performance to that of the supervised machine learning methods used by Aziz. et al.

Methodology

Dataset Creation and Preparation

A range of micrographs of the IN713C alloy samples were produced using selective laser melting (SLM) during a research program at the University of Sheffield, with varying settings for power, beam velocity and hatch spacing [21]. The micrographs, as shown in Fig. 1, were then imported into MATLAB and thresholded yielding a binary image of the entire micrograph, clearly showing the microstructural defects in white with a black background. The scale, shown on the bottom, along with defects that intersected the border of the micrograph were removed, to ensure the quality of the dataset. In addition, defects that were too small (below 10 pixels in width or height), were excluded as low-resolution images made manual classification challenging. After filtering unwanted data, individual binary images of the defects in the micrographs were created, as illustrated in Fig 1b. The images were categorised into specific classes including 'crack', 'pore', 'lack of fusion', and 'pore with crack' defects. This process was applied to 18 micrographs, yielding a dataset containing 4800 binary images of microstructural defects.

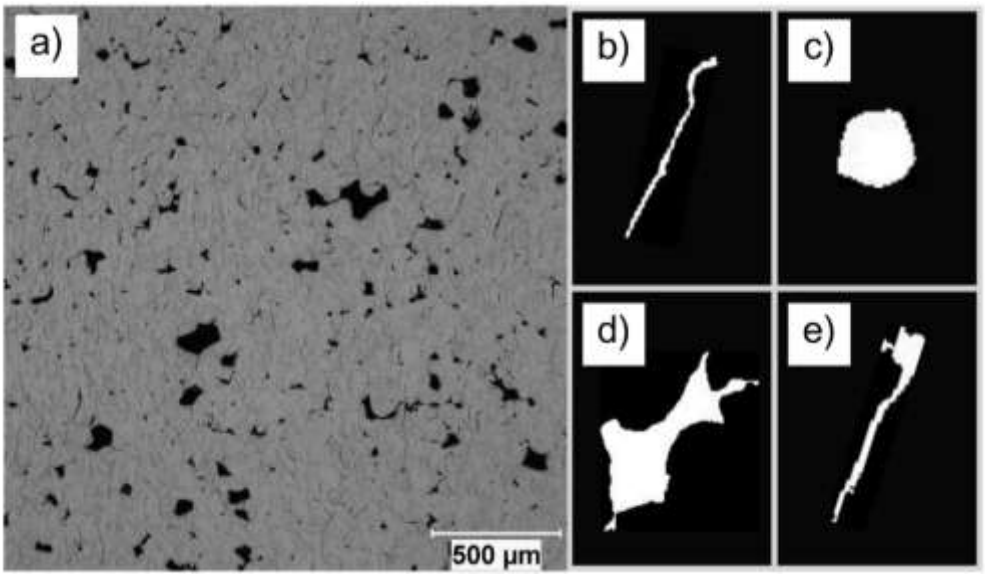

Figure 1: Example micrograph used to create the dataset, with example binary images of each defect type that were found in the micrographs. This includes b) 'crack', c) 'pore', d) 'lack of fusion, and e) 'pore with crack' defects.

The defect dataset images were standardised to a size of 200x200 pixels and underwent data augmentation. Deep learning models typically demand extensive datasets to gather sufficient information and patterns for effective training and performance. Data augmentation can artificially inflate the training set while preserving labels during transformations, leading to a larger dataset for effective training and performance [22]. Furthermore, data augmentation can support with avoiding overfitting to training data, which would result in the model's poor generalisation performance on unseen samples [23], [24]. In this work, augmentation involved a random reflection on the X and Y axis of the image, a random rotation of the image from 20 to -20 degrees around its centre, and a random scaling of 0.9 to 1.1. Following augmentation, the dataset was compiled into a training dataset (70%) and a validation dataset (30%).

Baseline Model

To begin, a baseline neural network was created as a foundation model for optimisation. It used a filter size of 5x5 which was kept consistent for all subsequent model training. The baseline model consisted of 4 blocks and 10 filters per batches. Each block is of a standard architecture, containing a 2d convolution layer, batch normalization layer, RELU layer and a max pooling layer. By varying the number of batches and stacking them on top of each other, we achieve a network with different architectures, modifying its ability to learn the patterns from the training data and improving overall performance.

Hyperparameter Selection and Optimisation

An important feature of designing and optimising neural networks is hyperparameter selection. Hyperparameters are settings or configurations of the models, which are freely selectable within a certain range and influence model performance. This differs to model parameters which are chosen during the learning process by the model itself [25]. In this work, two methods of hyperparameter selection are considered. The reason for performing hyperparameter optimisation, is because it considerable reduces the human effort in identifying the best combination of hyperparameters for best performance, improves the performance of the algorithm by customizing it to the given dataset/application, and it improves reproducibility and facilitates comparison between models [26]. In this work, two methods of hyperparameter optimisation were performed: grid search and Bayesian optimisation.

Grid search is a basic hyperparameter tuning method where a model is built using each combination of hyperparameters, trained, and then evaluated based on its performance [27]. In this grid search, the number of batches and filters in the CNN were explored, to evaluate which combination of these two CNN characteristics yields the optimal performance. A disadvantage to grid search is that as the range of values being explored increases, the computational efficiency of the algorithm drastically decreases. To combat this, each model trained during grid search used 25% of the baseline training dataset, improving computational efficiency of the algorithm. The optimal CNN architecture that yielded the optimal results in grid search was then trained with the full baseline training dataset, to evaluate its performance on a noisy dataset. During grid search, CNN architectures with 2-7 blocks and 5 to 25 filters (with increments of 5) were explored.

Bayesian optimisation is a more advanced approach to hyperparameter optimisation. The algorithm builds a response surface model using the mean and uncertainty predictions to guide the selection of subsequent data collection [28]. Due to its ability to learn from previous iterations, Bayesian optimisation offers a more efficient approach to hyperparameter selection and Neural Architecture Search when compared to grid search [29]. Due to this improvement compared to grid search, the scope of hyperparameters explored during optimisation was widened. This included varying the number of batches, number of filters, initial learn rate, mini batch size, learn rate drop factor, and learn rate drop period.

Results and Discussion

Confusion matrices describing the performance of the baseline model and models produced using optimisation methods are shown in Figure 2. Table 1 presents the performance of each model in accurately classifying various defects within the validation dataset along with its total accuracy. There was a steady increase in model classification performance as different optimisation methods were used to vary the hyperparameters of the model, with the Bayesian optimised model yielding the best classification performance. The network produced by gridsearch optimisation contained 7 batches and 10 filters, whereas Bayesian optimisation created a model with 8 batches and 26 filters. This suggests that CNNs with more complex architectures yield better results, as they are better able to capture distinctive features between different types of microstructural defects. Similar to supervised ML models produced by Aziz et al., the CNNs faced consistent challenges in accurately classifying 'Pore with Crack' defects. These defects combine the characteristics of 'Pore' and 'Crack' defects, which already exist as separate classes in the dataset. The CNNs struggle due to overlapping features between these classes, making it difficult to distinguish and classify them effectively. Moreover, this disparity could also stem from the imbalance in the dataset, where there were only 254 instances of 'Pore with Crack' defects out of a total of 4800 defects. This imbalance may suggest that the model is overfitting to the more prevalent defect types. Despite it's poor performance in correctly classifying 'Pore with Crack' defects, the models showed excpetional capability at distinguishing between 'Lack of Fusion', 'Pore' and 'Crack' defects. The final model produced by Bayesian Optimisation showed a overall performance of 91.6%, which rivals that of models produced by Aziz et al. [20]. It should be noted that the performance of the optimal models do vary each time the model was trained. This variability stems from factors such as the stochastic nature of training algorithms, such as stochastic gradient descent, which randomly selects mini-batches and adjusts learning rates based on varying gradients. Additionally, data shuffling between training epochs and random weight initialisation leads to diverse starting points and optimisation paths during training sessions. Baseline models converged to usually 85-88% in classification accuracy, grid search optimised models to 88-91% and Bayesian optimised models to 90-92%. It should be noted that other forms of performance metrics, such as sensitivity, precision, recall and the kappa (K) statistic are also available for model performance evaluation [20].

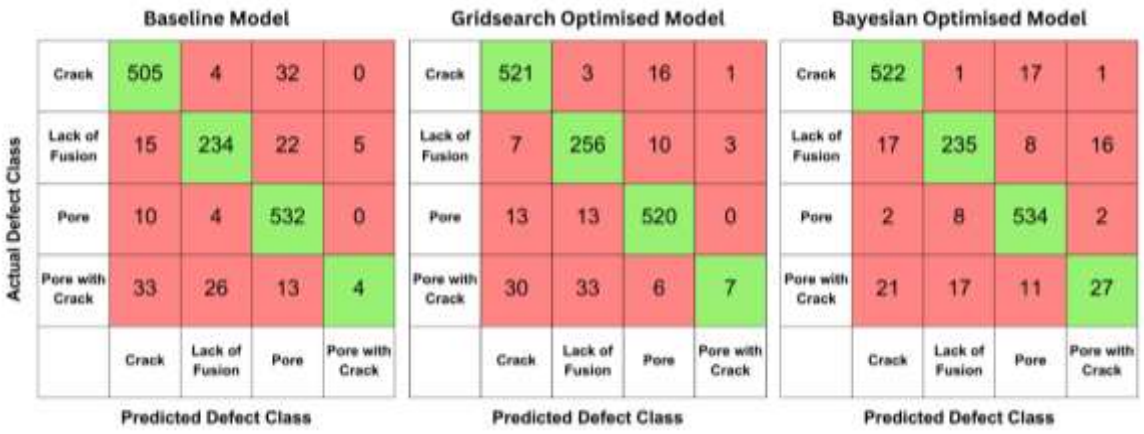

Figure 2: Confusion matrices showing the performance of the baseline, grid search optimised, and Bayesian optimised models.

Defect Type	**Baseline Model**	**Gridsearch Optimised Model**	**Bayesian Optimised Model**
Crack	93.3%	96.3%	96.5%
Lack of Fusion	84.8%	92.8%	85.1%
Pore	97.4%	95.2%	97.8%
Pore with Crack	5.3%	9.2%	35.5%
Total	**88.6%**	**90.6%**	**91.6%**

Table 1: Summary of accuracy for various models in correctly classifying defects in the validation dataset.

Conclusions

The networks that have been trained have shown good capability at classifying pores, cracks and lack of fusion defects. However, they have shown significant challenge in correctly classifying pore with crack defects. This is most likely since they share features of both ‘Pore’ and ‘Crack’ defects, as well as the sparse number of instances of this class.

The MATLAB scripts and the dataset are available on GitHub:

Acknowledgements

Declaration of Interests

None

References (Zotero)

[1] A. Mumith, M. Thomas, Z. Shah, M. Coathup, and G. Blunn, ‘Additive manufacturing current concepts, future trends’, *The bone & joint journal*, vol. 100B, no. 4, pp. 455–460, 2018, doi: 10.1302/0301-620X.100B4.BJJ-2017-0662.R2.

[2] S. Brischetto, P. Maggiore, and C. G. Ferro, ‘Special Issue on “Additive Manufacturing Technologies and Applications”’, *Technologies*, vol. 5, no. 3, 2017, doi: 10.3390/technologies5030058.

[3] J. ZHU, H. ZHOU, C. WANG, L. ZHOU, S. YUAN, and W. ZHANG, ‘A review of topology optimization for additive manufacturing: Status and challenges’, *Chinese Journal of Aeronautics*, vol. 34, no. 1, pp. 91–110, 2021, doi: https://doi.org/10.1016/j.cja.2020.09.020.

[4] M. M. Mojtaba Khanzadeh Sudipta Chowdhury, Mark A. Tschopp, Haley R. Doude and L. Bian, ‘In-situ monitoring of melt pool images for porosity prediction in directed energy deposition processes’, *IISE Transactions*, vol. 51, no. 5, pp. 437–455, 2019, doi: 10.1080/24725854.2017.1417656.

[5] M. Gorelik, ‘Additive manufacturing in the context of structural integrity’, *International Journal of Fatigue*, vol. 94, pp. 168–177, 2017, doi: https://doi.org/10.1016/j.ijfatigue.2016.07.005.

[6] M. C. Brennan, J. S. Keist, and T. A. Palmer, ‘Defects in Metal Additive Manufacturing Processes’, *Journal of Materials Engineering and Performance*, vol. 30, no. 7, pp. 4808–4818, Jul. 2021, doi: 10.1007/s11665-021-05919-6.

[7] G. Lorena, G. Robinson, P. Stefania, C. Pasquale, B. Fabiano, and M. Franco, ‘Automatic Microstructural Classification with Convolutional Neural Network’, in *Information and Communication Technologies of Ecuador (TIC.EC)*, M. Botto-Tobar, L. Barba-Maggi, J. González-Huerta, P. Villacrés-Cevallos, O. S. Gómez, and M. I. Uvidia-Fassler, Eds., Cham: Springer International Publishing, 2019, pp. 170–181.

[8] B. Lantz, *Machine Learning with R*, 1st edition. Birmingham, UNITED KINGDOM: Packt Publishing, Limited, 2013. [Online]. Available: http://ebookcentral.proquest.com/lib/sheffield/detail.action?docID=1343653

[9] E. Alpaydin, *Introduction to machine learning*, 3rd edition. in Adaptive computation and machine learning. Cambridge, Massachusetts: MIT Press, 2014.

[10] R. Snell *et al.*, ‘Methods for Rapid Pore Classification in Metal Additive Manufacturing’, *JOM*, vol. 72, no. 1, pp. 101–109, Jan. 2020, doi: 10.1007/s11837-019-03761-9.

[11] L. Scime and J. Beuth, ‘Anomaly detection and classification in a laser powder bed additive manufacturing process using a trained computer vision algorithm’, *Additive Manufacturing*, vol. 19, pp. 114–126, 2018, doi: https://doi.org/10.1016/j.addma.2017.11.009.

[12] P. K. Nalajam and R. V, ‘Microstructural porosity segmentation using machine learning techniques in wire-based direct energy deposition of AA6061’, *Micron*, vol. 151, p. 103161, 2021, doi: https://doi.org/10.1016/j.micron.2021.103161.

[13] N. Ketkar, *Deep Learning with Python : A Hands-On Introduction*. Berkeley, CA, UNITED STATES: Apress L. P., 2017. [Online]. Available: http://ebookcentral.proquest.com/lib/sheffield/detail.action?docID=6363117

[14] Y. LeCun, Y. Bengio, and G. Hinton, ‘Deep learning’, *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015, doi: 10.1038/nature14539.

[15] A. Chowdhury, E. Kautz, B. Yener, and D. Lewis, ‘Image driven machine learning methods for microstructure recognition’, *Computational Materials Science*, vol. 123, pp. 176–187, 2016, doi: https://doi.org/10.1016/j.commatsci.2016.05.034.

[16] A. R. Durmaz *et al.*, ‘A deep learning approach for complex microstructure inference’, *Nature Communications*, vol. 12, no. 1, p. 6272, Nov. 2021, doi: 10.1038/s41467-021-26565-5.
[17] R. K. Tan, N. L. Zhang, and W. Ye, ‘A deep learning–based method for the design of microstructural materials’, *Structural and Multidisciplinary Optimization*, vol. 61, no. 4, pp. 1417–1438, Apr. 2020, doi: 10.1007/s00158-019-02424-2.
[18] A. Beniwal, R. Dadhich, and A. Alankar, ‘Deep learning based predictive modeling for structure-property linkages’, *Materialia*, vol. 8, p. 100435, 2019, doi: https://doi.org/10.1016/j.mtla.2019.100435.
[19] E. A. Holm *et al.*, ‘Overview: Computer Vision and Machine Learning for Microstructural Characterization and Analysis’, *Metallurgical and Materials Transactions A*, vol. 51, no. 12, pp. 5985–5999, Dec. 2020, doi: 10.1007/s11661-020-06008-4.
[20] U. Aziz, A. Bradshaw, J. Lim, and M. Thomas, ‘Classification of defects in additively manufactured nickel alloys using supervised machine learning’, *Materials Science and Technology*, vol. 39, no. 16, pp. 2464–2468, 2023, doi: 10.1080/02670836.2023.2207337.
[21] C. Liu, ‘Selective laser melting of nickel superalloys for aerospace applications: defect analysis and material property optimisation’, Doctoral Thesis, University of Sheffield, Sheffield, 2021. [Online]. Available: https://etheses.whiterose.ac.uk/30067/
[22] L. Taylor and G. Nitschke, ‘Improving Deep Learning with Generic Data Augmentation’, in *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2018, pp. 1542–1547. doi: 10.1109/SSCI.2018.8628742.
[23] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, ‘Dropout: A Simple Way to Prevent Neural Networks from Overfitting’, *Journal of Machine Learning Research*, vol. 15, no. 56, pp. 1929–1958, 2014.
[24] J. Sietsma and R. J. F. Dow, ‘Creating artificial neural networks that generalize’, *Neural Networks*, vol. 4, no. 1, pp. 67–79, 1991, doi: https://doi.org/10.1016/0893-6080(91)90033-2.
[25] E. Bartz, T. Bartz-Beielstein, M. Zaefferer, and O. Mersmann, *Hyperparameter Tuning for Machine and Deep Learning with R : A Practical Guide.*, 1st ed. Singapore: Springer, 2023.
[26] M. Feurer and F. Hutter, ‘Hyperparameter Optimization’, in *Automated Machine Learning: Methods, Systems, Challenges*, F. Hutter, L. Kotthoff, and J. Vanschoren, Eds., Cham: Springer International Publishing, 2019, pp. 3–33. doi: 10.1007/978-3-030-05318-5_1.
[27] C. Antal-Vaida, ‘Basic Hyperparameters Tuning Methods for Classification Algorithms.’, *Informatica Economica*, vol. 25, no. 2, 2021.
[28] J. Ungredda and J. Branke, ‘Bayesian Optimisation for Constrained Problems’, *ACM Trans. Model. Comput. Simul.*, vol. 34, no. 2, Apr. 2024, doi: 10.1145/3641544.
[29] K. Kandasamy, W. Neiswanger, J. Schneider, B. Poczos, and E. Xing, ‘Neural Architecture Search with Bayesian Optimisation and Optimal Transport’, *arXiv.org*, 2018, [Online]. Available: https://www.proquest.com/working-papers/neural-architecture-search-with-bayesian/docview/2073873410/se-2?accountid=13828