Joint Special Interest Group meeting between euspen and ASPE
Advancing Precision in Additive Manufacturing
KU Leuven, Belgium, September 2023
www.euspen.eu

Enhancing keyhole porosity detection in laser powder bed fusion using a hybrid approach of machine learning and physics-informed knowledge

Zhengrui Tao[1], Aditi Thanki[1,2], Louca Goossens[1], Ann Witvrouw[1,2], Bey Vrancken[1,2], Wim Dewulf[1,2]

[1]*KU Leuven, Department of Mechanical Engineering, Belgium*
[2]*Flanders Make@KU Leuven*

zhengrui.tao@kuleuven.be

Abstract

The increasing use of Laser Powder Bed Fusion (L-PBF) in various industrial sectors is driven by its ability to produce parts with greater control over dimensional accuracy and surface roughness compared to other Additive Manufacturing (AM) techniques. However, keyholes - narrow and deep melt pools - may form during the process, leading to porosity in the final product and highlighting the need for real-time detection and prevention of keyhole porosity. In this paper, physics-informed signatures of the meltpool were extracted from in-situ optical meltpool images, including morphology and spatial distribution of spatter ejecta and meltpool. The extracted signatures were then analyzed using computationally feasible machine learning (ML) methods. The proposed approach uses a high-speed CMOS camera to record the L-PBF process zone, enabling real-time detection of keyhole porosity without complicated computations. The investigation involved the production of 12 scan tracks on the top surface of a cuboid made of Ti6Al4V with controlled keyhole pores by doubling the laser power or halving the scanning speed. Simple ML models such as Support Vector Machine were used to identify the occurrence of keyhole pores, achieving high statistical fidelities with Area Under Receiver Operating Characteristic (AUROC) and Area Under Precision-Recall Curve (AUPRC) reaching 93% and 90%, respectively. Overall, using a high-speed CMOS camera, the proposed process-informed dataset building, physics-informed feature engineering, and simple ML methods demonstrate high-accuracy production quality monitoring. This approach provides a more straightforward and more accessible solution for real-time detection of keyhole porosity during L-PBF, which can lead to timely adjustments and improvements in the quality of printed parts.

In-process monitoring, X-ray computed tomography, Laser Powder Bed Fusion

1. Introduction

During the L-PBF process, the meltpool is generated by directing a laser beam on the metal powders, and then solidifies to the consolidated structure. Consequently, the meltpool is a fundamental feature of this process, and its stability, dimensions, and behavior play a critical role in determining the quality of the process. Co-axial meltpool monitoring (MPM) is commonly deployed in situ for L-PBF AM processes. Unlike off-axis imaging, co-axial MPM offers much greater spatial resolution across the part surface by utilizing the laser-galvo system to 'scan' the camera field of view across the build plane in synchronization with the laser [1]. Multiple algorithmic methods for correlation between meltpool images and porosities exist [2,3]. There are misalignments between the locations of keyhole pores in X-ray computed tomography (X-CT) volume and the positions of the sensor signaled when pores were initially generated. The keyhole pores can move within the meltpool after pinching off the keyhole, leading to substantial mislabeling. Researchers at KU Leuven have recently focused on a method of many co-axial MPM images to one keyhole pore in scan track experiments. Generally, this method takes sequential co-axial MPM images considering the camera's frame rate and thermal-affect duration time, and the meltpool length and scan speed determine the latter. Then, the images can be stacked and registered to one location in X-CT volume, corresponding to the laser spot position.

The physics-informed MPM image features that target L-PBF keyhole porosity are extracted. When used with simple ML models, a small set of pragmatic, physics-informed features will detect pores at par with a complex and computationally intensive deep-learning model using raw meltpool images. Besides, the process-informed many-to-one dataset and the physics-informed featuring engineering make the model training more interpretable. This paper aims to explore and demonstrate the idea and build upon the scan track experiment.

2. Methodologies

This paper describes the process for collecting data, labeling single-track quality based on X-CT, extracting features from MPM data, and developing ML models. Our methodology can be applied to other metal AM systems, not just the L-PBF hardware used in our study. The collected MPM video can also be obtained using different sensing modalities.

2.1. Experiment design

A dense cuboid part (10 mm×10 mm×5 mm) was built with 12 scan tracks on the top surface. The feedstock material used is Ti-6Al-4V grade 23 powder, with a particle size ranging between 15-45 µm. The layer thickness is 30 µm, and the hatch space is 70 µm for the substrates and the single-line tracks. The parameters are listed in Table 1, and the in-situ MPM image and the X-CT volume are shown in Figure 1. The image size is 80×80 pixels, with each pixel having an 8-bit grayscale depth. The calibrated pixel size is 14 µm, resulting in a field of view of 1120 µm×1120 µm.

Table 1 L-PBF process parameters

Process Step	Laser Power P (W)	Scan Speed v (mm/s)	Length L (mm)	Line Number
Cuboid	170	1000	10	/
C1	340	1000	6	1-6
C2	170	500	8	7-8

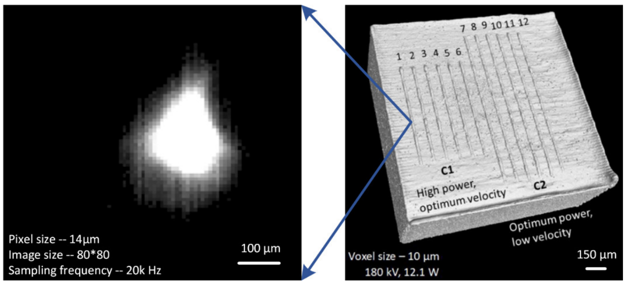


Figure 1. An MPM image depicting a meltpool moving downwards and X-CT volume.

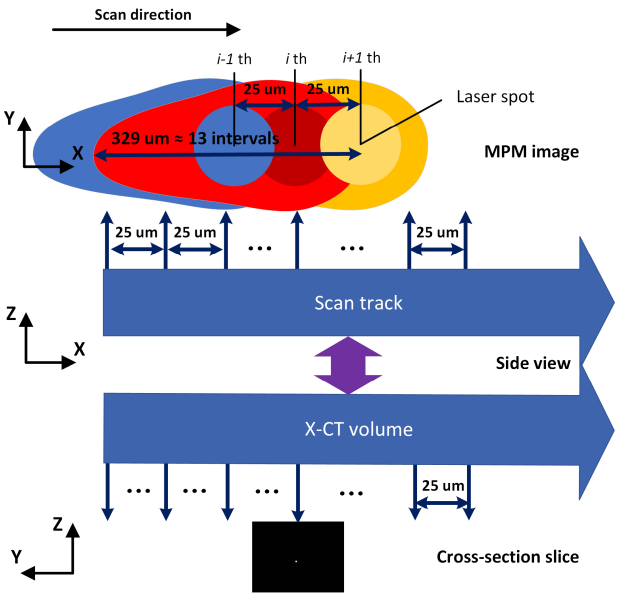
2.2. Porosity analysis

The part was scanned on a Nikon XT H 225 ST. A magnification factor of 20 was applied, resulting in a voxel size (resolution) of 10 µm. Reconstruction of volumetric data from the individual projection images was completed in CT-Pro 3D with a Feldkamp-David-Kress algorithm using a Shepp-Logan digital filter. The reconstructed volume represented as unsigned 16-bit integers were then imported into VGStudio MAX 2022.1 as a .raw file. To facilitate registration and fit geometric primitives to the sample, a thresholding operation was performed on the volume. This operation involved two stages: first, the volume was segmented using the ISO50 method, and then the surface was refined using VG's "advanced mode" with a search distance of 8 voxels to eliminate all particles and voids. This advanced method allows for surface determination at a sub-voxel level. Following thresholding, the part coordinate system was established using a 3-2-1 registration, aligning it with the component's orientation in the baseplate. Each scan track, consisting of 6 layers beneath the line scan, was segmented from the X-CT volume for porosity analysis.

The VGEasypores relative algorithm [4] was employed for porosity analysis, in which the contrast (%) is setting 15, and the local area size (%) is using 10 and checking the refinement and setting the search distance as 3 voxels. The pores below 8 voxels are filtered out, and keep the larger pores. The cross-section slices, spaced at 5 µm intervals along the scan direction, were exported as 16-bit TIF images, with pores marked in the dark. The slices were then processed in MATLAB, such as filtering, gray scaling, and binarization with pores marked in white. In one cross-section slice, if one pixel is white, the label for the cross-section is 1. Thus, the in-process keyhole porosity detection can be treated as a binary classification problem.

2.3. L-PBF process-informed knowledge

In this work, meltpool dynamics are modeled as a time series problem. Multiple MPM frames across one location impact the status of that position, as shown in Figure 2. Considering frame rate, scan speed, and the average meltpool length, the camera takes 13 images for Lines 6-10 and 11 images for Lines 1-5 when the meltpool passes by one position in the scan track. Here we choose 13 images for both scan track groups to facilitate unified processing, which means 13 neighboring MPM images may contain information about one position's porosity formation.



i th slice is determined by the 13 neighboring MPM images

Figure 2. Schematic diagram of L-PBF process-informed dataset

2.4. Physics-informed feature engineering

Physically intuitive signatures were extracted from MPM frames, such as meltpool morphology, spatter morphology, temperature distribution, etc. These features are outlined in Table 2. Deep Autoencoders (DAE) are designed to compress input data into a compact feature vector using an encoder and then reconstruct the original data using a decoder. The DAE decomposes the MPM image into interpretable representations, offering a proper feature space for ML models to learn the boundary between the Non-keyhole pore and the Keyhole pore. Thus, 104 features can be obtained from one MPM image. Considering one sample with 13 MPM images, a feature matrix with a dimension of (13, 104) for each sample. To enable ML models to process the features, the 13 rows of the matrix are fused into a single row, resulting in a vector with 104 elements. The simple feature fusion method, Mean operation, is applied along the row direction.

Table 2 Physics-based Features from one MPM image

Features	Dimension	Features	Dimension
Meltpool length	(1,)	Temperature gradient	(80,)
Meltpool width	(1,)	#Spatter	(1,)
Meltpool eccentricity	(1,)	Spatter area	(1,)
Meltpool area	(1,)	Spatter ratio	(1,)
Meltpool perimeter	(1,)	Latent vector	(16,)

Next, in Figure 3, we test the ability of the extracted features to differentiate between the formation of keyholes without the need for an ML model. It is noticed that the meltpool features and latent vectors ((a) and (d)) show more discernable clustering compared to the temperature gradient and spatter-related features ((b) and (c)). However, there are significant overlaps between Non-keyhole pore and Keyhole pore regimes. The significant overlap between these clusters and the nonlinear interaction between features necessitates the need for ML models.

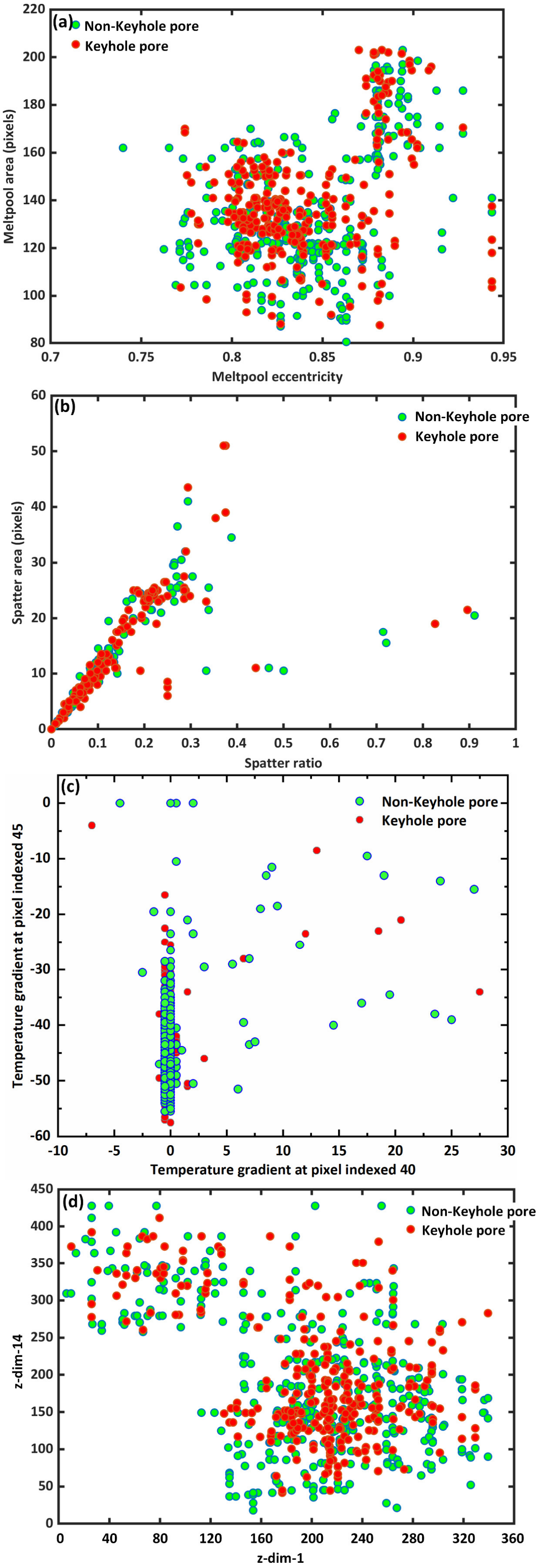


Figure 3. Correlation between features, where (a) and (b) depict Meltpool and Spatter-related features, and (c) and (d) depict the temperature gradient and latent vector.

3. Results and discussions

All the ML models are implemented in Scikit-learn [5] or Keras [6]. The setting parameters and the parameters to be tuned using a five-fold grid search are both listed in Table 3. Other parameters needed to establish the models are default.

Table 3 Parameters used in simple ML models

Models	Setting parameters	Parameters to be tuned

Logistic Regression (LR) [7]	solver=’saga’,
	max_iter=300
		C=[10^-5, 10^-3, 10^-1, 1,10, 1000, 10^5],
		Penalty={‘l1’, ‘l2’}

K-Nearest Neighbors (KNN) [8]	n_neighbors={3, 5, 11, 19},
	weights={'uniform', 'distance'}	

Gaussian Naive Bayes (GNB) [9]		

Support Vector Machine (SVM) [10]	probability=True	C={0.001, 0.01, 0.1, 1, 10},
		gamma={0.0001, 0.001, 0.01, 0.1, 1}

Decision Tree (DT) [11]	max_features={1,3,10},
	min_samples_split={2,3,10},
	min_samples_leaf={1,3,10},
	criterion={'entropy', 'gini',' log_loss'}	

Random Forest (RF) [12]	max_features={1,3,10},
	min_samples_split={2,3,10},
	min_samples_leaf={1,3,10},
	n_estimators={20,50,100},
	criterion={'entropy', 'gini',' log_loss'}	

Multilayer Perceptron (MLP) [13]	model.add(Dense(64, input_dim=104, activation='relu')), model.add(Dropout(0.7)), model.add(Dense(1, activation='sigmoid', bias_initializer=output_bias))	

Table 4 presents the performance of ML models. The green-marked rows indicate the top two models: RF and SVM. The results demonstrate that utilizing 104 features as predictors achieved excellent capability for identifying keyhole pores, with the RF model reaching an AUROC of 0.93 and AUPRC of 0.9.

Table 4 Results of keyhole pore detection using simple ML models (refer to Table 3 for the acronyms)

Models	AUROC	AUPRC
LR	0.76	0.7
KNN	0.89	0.85
GNB	0.63	0.52
SVM	0.91	0.86
DT	0.78	0.64
RF	0.93	0.9
MLP	0.9	0.84

5. Conclusion

This study demonstrates a method for detecting keyhole formation in L-PBF by utilizing physically intuitive meltpool signatures and simple ML models. This study's main contributions are twofold: firstly, establishing a many-to-one dataset considers the interaction between meltpools during laser spot moves, resulting in a more physics-interpretable and label-aligned method than the traditional one-to-one approach.

Secondly, combining interpretable physics-informed signatures with readily implementable ML models allows for rapid detection of pores and eliminates the delay inherent in complex data-driven defect detection algorithms. This study represents a significant advancement toward the online, real-time detection of defects during the L-PBF building process.

References

[1] L. Mazzoleni, A.G. Demir, L. Caprio, M. Pacher, B. Previtali, Real-Time Observation of Melt Pool in Selective Laser Melting: Spatial, Temporal, and Wavelength Resolution Criteria, IEEE Trans. Instrum. Meas. 69 (2020) 1179–1190. https://doi.org/10.1109/TIM.2019.2912236.

[2] Z. Smoqi, A. Gaikwad, B. Bevans, M.H. Kobir, J. Craig, A. Abul-Haj, A. Peralta, P. Rao, Monitoring and prediction of porosity in laser powder bed fusion using physics-informed meltpool signatures and machine learning, J. Mater. Process. Technol. 304 (2022) 117550. https://doi.org/10.1016/j.jmatprotec.2022.117550.

[3] H. Yeung, Z. Yang, L. Yan, A meltpool prediction based scan strategy for powder bed fusion additive manufacturing, Addit. Manuf. 35 (2020) 101383. https://doi.org/10.1016/j.addma.2020.101383.

[4] VGStudio MAX 3.2 software, https://www.volumegraphics.com/en/products/vgstudio-max.html, (n.d.).

[5] scikit-learn: machine learning in Python — scikit-learn 1.2.1 documentation, (n.d.). https://scikit-learn.org/stable/index.html (accessed February 6, 2023).

[6] Keras: Deep Learning for humans, (n.d.). https://keras.io/ (accessed June 23, 2023).

[7] sklearn.linear_model.LogisticRegression, Scikit-Learn. (n.d.). https://scikit-learn/stable/modules/generated/sklearn.linear_model.LogisticRegression.html (accessed February 13, 2023).

[8] sklearn.neighbors.KNeighborsClassifier, Scikit-Learn. (n.d.). https://scikit-learn/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html (accessed February 13, 2023).

[9] sklearn.naive_bayes.GaussianNB, Scikit-Learn. (n.d.). https://scikit-learn/stable/modules/generated/sklearn.naive_bayes.GaussianNB.html (accessed February 13, 2023).

[10] sklearn.svm.SVC, Scikit-Learn. (n.d.). https://scikit-learn/stable/modules/generated/sklearn.svm.SVC.html (accessed February 13, 2023).

[11] sklearn.tree.DecisionTreeClassifier, Scikit-Learn. (n.d.). https://scikit-learn/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html (accessed February 13, 2023).

[12] sklearn.ensemble.RandomForestClassifier, Scikit-Learn. (n.d.). https://scikit-learn/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html (accessed February 13, 2023).

[13] K. Team, Keras documentation: Imbalanced classification: credit card fraud detection, (n.d.). https://keras.io/examples/structured_data/imbalanced_classification/ (accessed February 14, 2023).