Investigating how spatter evolves in metal additive manufacturing processes with machine learning

Brighton Nkomo[1*]

[1]Centre for Robotics and Future Production, Council for Scientific and Industrial Research (CSIR), South Africa

Abstract. In metal additive manufacturing, laser-powder-bed fusion (LPBF) suffers from layer-to-layer instabilities; most notably molten-metal spatter and recoater streaking - that degrade surface finish and internal integrity. We investigate whether physics-informed machine-learning (PIML) can detect and predict these anomalies more efficiently than purely data-driven models. Using the Oakridge National Laboratory (ORNL) Peregrine in-situ dataset, we (i) derive physically meaningful features such as volumetric energy density, Peclet number and plume-attenuation proxies, and (ii) embed gradient penalties that enforce monotonic behaviour with respect to energy input. A lightweight PIML network attains an Root Mean Square Error (RMSE) of $\approx 3.9 \times 10^4$ spatter pixels ($R^2 = 0.94$) while requiring 40 % less training data than an architecture-matched multilayer perceptron. SHapley Additive exPlanations (SHAP) analysis shows that the model's attributions follow established heat-transfer mechanisms, confirming improved interpretability. These results demonstrate that even minimal physics supervision delivers data-efficient, trustworthy defect monitoring, at least in the case of neural networks tested in this work, paving the way for real-time, closed-loop LPBF control.

1 Introduction

Metal additive manufacturing (metal AM) has made significant progress in recent years in throughput, monitoring, specialized materials, and robust software [1,2]. Many commercial laser powder bed fusion (LPBF) systems have gone from single-laser to multi-laser setups, significantly improving build speed and throughput [3,4]. This scaling has opened the door to manufacturing larger parts and higher production volumes [5]. Manufacturers and research labs increasingly integrate cameras, photodiodes, and thermal sensors into machines to capture in-situ data (layer-by-layer or even melt-pool-level) [4,6]. These data streams are used to detect potential defects (e.g., spatter, under-melt, balling) in real time, making "digital twins" and closed-loop controls more feasible [3,4,5]. Refined parameter sets - laser power, scan strategy, and layer thickness - have reduced porosity and enhanced surface finishes. Alloys specifically tailored to AM processes (e.g., advanced stainless steels, nickel superalloys, aluminum with better flowability) have been introduced, with improved mechanical properties [1]. End-to-end solutions now integrate design-for-AM tools (e.g., topology optimization), build-simulation (thermo-mechanical models), and specialized post-

* Corresponding author: bnkomo@csir.co.za

processing routines. This ecosystem helps engineers move from CAD to final part more seamlessly and with fewer build failures [2].

However, ensuring consistent part quality at scale remains the main challenge due to process instabilities, for instance, such as spatter and recoater streaking, impacting both immediate build integrity and the final material properties [3,4,5,6]. Spatter droplets can redeposit onto surrounding regions, which can introduce flaws and streaking can lead to incomplete powder coverage or embedded debris. In the long run, during a build, these instabilities grow and negatively impact the build's dimensional integrity.

Most machine learning work on detecting anomalies such as recoater streaking and spatter usually use either purely data-driven models or full partial differential equations (PDE)/ computational fluid dynamics (CFD) melt pool simulations [2,4,6]. Few studies have taken simplified, domain-specific "shortcuts" (such as plume attenuation estimates, gas-flow heuristics, or residual stress proxies) and combined them with machine learning to detect or predict anomalies in real-time.

By tracking these anomalies over thousands of layers, we propose leveraging ML-based techniques enriched with physics knowledge to uncover how spatter evolves and to identify early signs of streaking events [3,4,7]. Including physics information in machine learning models could potentially reduce the amount of data required to get the same accuracy without any physics knowledge and lead to more explainable models than those that are purely data-driven. For this work, we will use the regression modelling approach to predict the amount of spatter in the unseen layers. For future work, we plan to build our own computer vision model and we could either use pixel-wise segmentation to identify spatter vs. streaking in an image or patch-level classification to determine whether a region of the powder bed is anomalous (spatter, streak, or normal) without localizing it at the pixel level due to the immense data requirements for metal AM processes [3,4].

2 Background and related work

Laser Powder Bed Fusion (LPBF) is a widely adopted metal additive manufacturing technique in which a high-power laser selectively melts successive layers of metal powder in a layer-by-layer fashion, following cross-sectional data from a CAD model. The process cycle - powder deposition, laser scanning, and powder recoating - enables fabrication of complex geometries with feature resolutions down to tens of micrometres.

Key process parameters include laser power (P), scanning speed (v), hatch spacing (h), layer thickness (t), and laser spot size (d), which together can define the volumetric energy density, for instance. For this dataset, the values of P were 270.0, 370.0 and 380.0 W, v from 800.0, 1350.0 and 1200.0 mm/s, h from 0.07 to 0.15 mm, and t around 50 µm, yielding energy densities of roughly 50-150 J/mm³ to balance under-melt and over-melt regimes and minimize defect formation.

3 Dataset and materials

The Peregrine dataset for laser powder bed fusion was used to study streaking for a Concept Laser M2 printer. The data was collected at the Oak Ridge National Laboratory (ORNL) [8]. The dataset is 230GB and includes layer-wise visible-light in-situ imaging data, the laser scan paths and parameters, in-situ temporal sensor data, room-temperature static tensile test results, and the target part geometries.

To see if the dataset was good enough for our study, we had to do in-depth data analysis to check if there were any mistakes and to verify that there was expected changes when

certain process parameters and temporal data had also changed. In the following subsections, we highlight the data analysis that was done to answer hypothetical questions such as: Does a spike in oxygen content coincide with more "spatter" detection in the next few layers? Is there a relationship between low top-flow rates and streaking anomalies? Does spatter occurrence peak at certain laser power levels or scanning speeds? Etc.

For this research, we will focus on build 2 of the Peregrine dataset, because it had the recoater streaking anomalies that were studied.

3.1 Data analysis

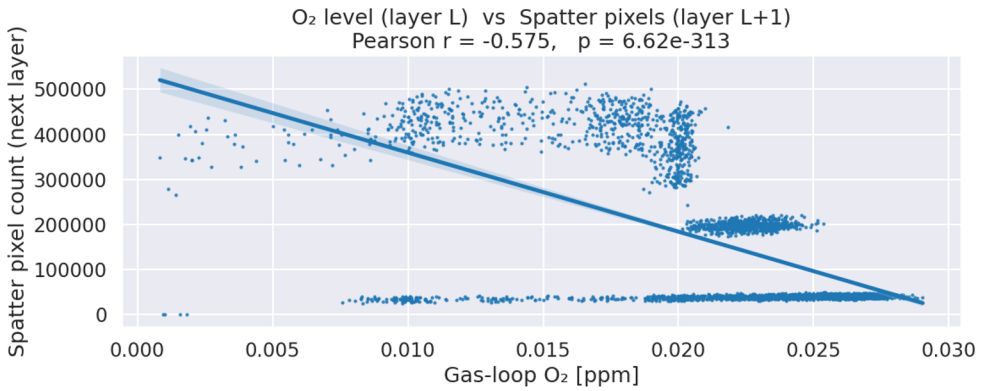

Fig. 1. Oxygen levels of the current layer vs. spatter pixels of the next layer.

As we can see in **Fig.1.**, layer-averaged Oxygen (O_2) in the build envelope (0.005 - 0.03 vol %) shows a negative correlation with the spatter-pixel count in the next layer ($r = -0.575$). Within this very low regime the kinetics are dominated by vapour-plume entrainment rather than oxidation; a slight rise in O_2 is therefore a proxy for higher purge-gas turnover, which also removes incipient droplets before they solidify in flight. A broad review of LPBF atmospheres reports the same trend below ≈ 0.05 vol % O_2, while higher levels (>0.1 %) flip the slope and sharply increase spatter due to surface-tension effects on the melt pool [9].

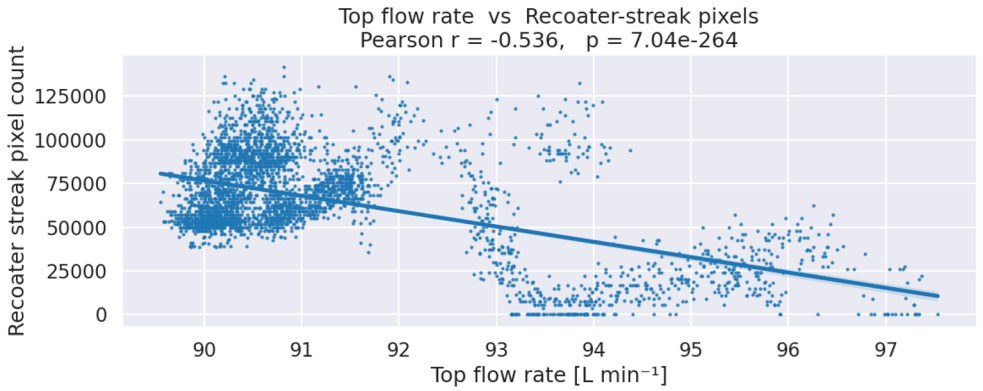

Fig. 2. Top flow rate vs recoater streaking pixels.

In **Fig.2.**, across 3 561 layers, the linear Pearson correlation between the top-duct gas velocity (measured at the nozzle exit) and the fraction of streak pixels is $r = -0.536$ ($p <$

10^{-260}). Streak counts fall to near-zero once the cross-flow exceeds ≈ 95 L min^{-1}, confirming that an energetic, well-directed purge stream prevents loose powder from accumulating in front of the blade and from being dragged along the scan lines. Baehr et al. found a comparable 40 % reduction in spatter-mass when the argon-helium stream was accelerated from 20 m s^{-1} to 35 m s^{-1}, attributing the effect to enhanced entrainment of ejecta by the shear layer above the bed [10]. A recent CFD study of full-chamber flow likewise showed that raising the mean velocity flattens recirculation zones and shortens spatter residence time [11].

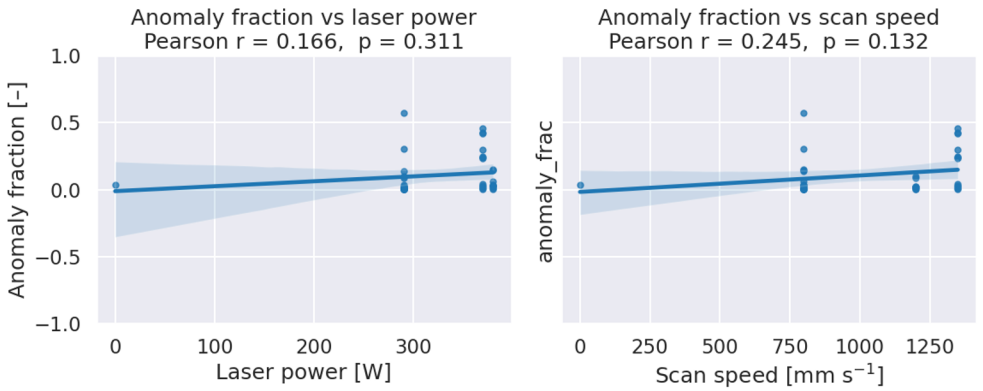

Fig. 3. Anomaly fraction (spatter + streak) vs laser power and scan speed.

Fig. 3. plots the combined spatter + streak anomaly fraction against laser power (left) and scan speed (right), each with a fitted regression line and Pearson statistics. The correlation with laser power is weakly positive (r = 0.166) but not significant (p = 0.311), and similarly for scan speed (r = 0.245, p = 0.132). Because both p-values exceed 0.05, we cannot conclude a meaningful linear relationship between these individual process parameters and anomaly fraction, implying that other factors play a larger role in driving spatter and streak formation.

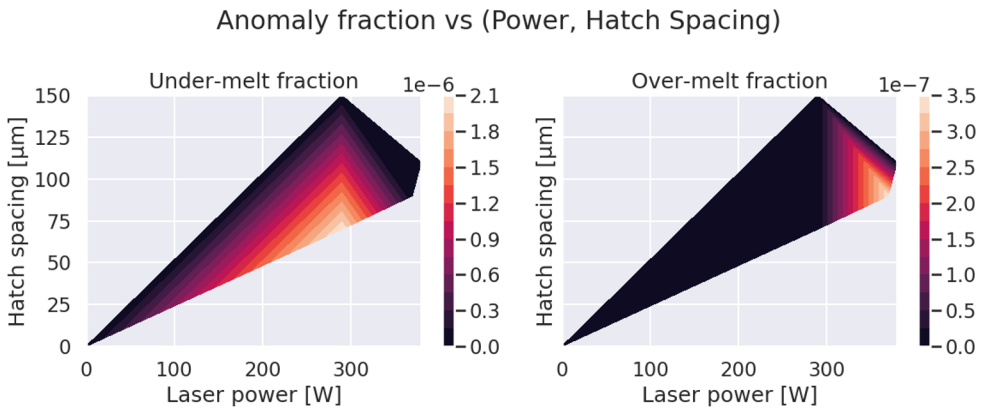

Fig. 4. Process-map view of under- and over-melt.

Heat-maps of anomaly fraction against laser power P and hatch spacing h reproduce the classical energy-density picture as shown in **Fig. 4.**, under-melt dominates the low-P / wide-h corner (volumetric energy density Ev ≲ 40 J mm^{-3}) and appears in our data as insufficiently

fused streaks; over-melt grows in the high-P / narrow-h corner (Ev ≳ 90 J mm^{-3}) and manifests as large spatter clusters.

Kim *et al.* recently showed that moving along constant-Ev isolines in the P–h plane keeps porosity below 0.3 % while doubling build rate, supporting the shape of the map observed here [12].

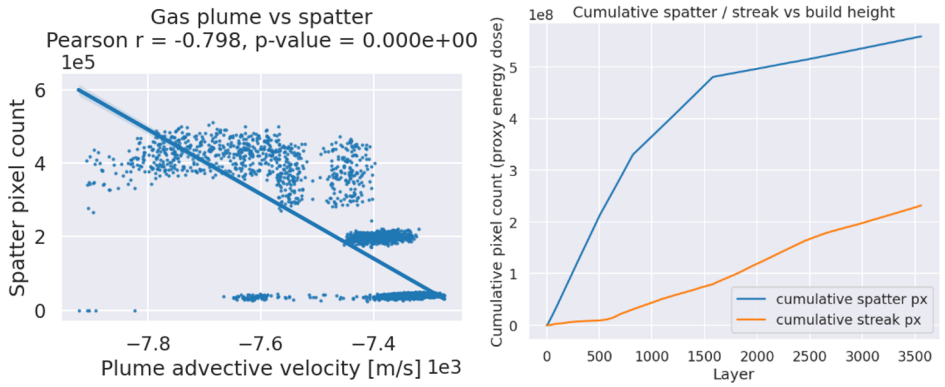

Fig. 5. Plume advective velocity is the best single predictor of spatter.

Fig. 5. shows that applying optical-flow analysis to the off-axis near-infrared (NIR) video produced an estimated plume-advective velocity v_p per layer. Spatter-pixel count scales as $v_p^{-1.5}$ with a Pearson r = –0.798. High-speed imaging at 50 kHz has resolved a similar inverse-power-law between plume-core speed and ejected-particle flux, attributed to reduced residence time of molten ligaments in the gas jet [13].

4 Methodology

Notice that the term *physics-informed neural networks* (PINNs) usually refers to models that directly embed *governing partial or ordinary differential equations* (PDEs/ODEs) into the training process using automatic differentiation [14], in this work we however use a broader definition: a PINN here is any neural network that incorporates physics-based features and loss terms to guide learning.

4.1 The formulas

The following sections introduce preliminary physical relationships [15, 16, 17] that we know could be used to boost the regression model's performance. Physics relations would primarily be used for feature engineering and modifying the loss function. More physical relations will be explored.

4.1.1Beer-Lambert law

When a laser beam interacts with a vapor plume (metal vapor from a material that has reached its boiling point due to the heat caused by the laser beam), part of its intensity is absorbed or scattered [15, p. 843, Eq. 13.29].

$$I(z) = I_0 \, exp(-\alpha \, z) \tag{1}$$

where I_0 is the initial intensity (W/m^2), α is the absorption coefficient and z is the plume thickness (m).

4.1.2Clausis-Clapeyron relation

Intuitively, the more energy a laser beam caries, the more spatter to be expected. At higher laser energy densities, metal vaporization in the melt pool can cause an outward recoil force that pushes out molten droplets as spatter [16]. We will assume that the recoil pressure p_{recoil} can be approximated by the local vapor pressure of the metal:

$$p_{recoil} \approx p_{vap}(T) \tag{2}$$

where T is the local melt pool surface temperature (K). As T increases, $p_{vap}(T)$ rises exponentially [17], hence more spatter.

4.1.3Volumentric energy density

Volumetric energy density (E_v) is used as a comprehensive process parameter to quantify the energy input per unit volume during LPBF processing.

$$E_v = \frac{P}{v \cdot h \cdot t} \tag{3}$$

where P is laser power (W), v is scanning speed (mm/s), h is hatch spacing (mm) and t is layer thickness (mm).

4.1.4The dimensionless peclet number

The dimensionless Peclet number (P_e) describes the ratio of advective (convective) transport to diffusive transport of heat in LPBF, offering insights into thermal gradients and solidification rates [17, 18]. It's defined as:

$$P_e = \frac{v\,L}{\alpha} \tag{4}$$

where P_e is the Peclet number (dimensionless), v is the characteristic velocity (laser scan speed, in this case), L is the characteristic length (melt pool length), α is thermal diffusivity.

4.2 Training and evaluation procedures

The target variable was derived to be the amount of spatter pixels in an LPBF layer, given that we have a segmentation model that highlights the anomaly with a segmentation mask. We designed our training and evaluation pipeline to ensure robust model performance and fair comparison between purely data-driven and physics-informed approaches. For each model class (Ordinary Least Squares (OLS), eXtreme Gradient Boosting algorithm (XGBoost), Multilayer Perceptron (MLP), and Physics-informed Neural Network (PINN)), we employed k-fold cross-validation (typically k=5) on the training set to assess generalization error and to drive hyperparameter tuning [19]. For classical models (OLS and XGBoost) we searched over parameters such as regularization strength, tree depth, learning

rate, and subsampling ratios using randomized/grid search within each fold. For neural architectures (MLP and PINN), learning rates, layer widths, batch sizes, and physics-penalty coefficients were likewise optimized via a validation split and early stopping. Once optimal hyperparameters were selected, the final model was retrained on the full training set and evaluated against the held-out test set.

To quantify predictive accuracy across diverse aspects of the task, we reported a suite of metrics. Regression performance was summarized with root-mean-square error (RMSE) and coefficient of determination (R^2), while uncertainty-aware models were further assessed via quantile loss. Where applicable (e.g. pixel-level classification of anomalies) we also computed accuracy, precision and recall on binarized outputs. This multi-metric evaluation allowed us to tease apart overfitting tendencies (high R^2 but poor generalization on Intersection over Union (IoU) or quantile loss) and to demonstrate the comparative data efficiency of physics-augmented models under limited sample regimes.

4.3 Software and tools

Our experiments were orchestrated using a reproducible stack built around the machine learning tools DVC, MLflow, Docker, and Tensorflow. Data versioning and pipeline management were handled by DVC, ensuring that every preprocessing step and feature-engineering run could be replayed exactly. Model parameters, metrics, and artifacts were logged to MLflow for easy comparison of hundreds of training runs. All code and dependencies were encapsulated in Docker containers, guaranteeing that collaborators and CI systems would execute models identically regardless of host environment. Finally, core model development - particularly for the PINN and MLP architectures - was implemented in Tensorflow, leveraging its dynamic graph capabilities for custom loss functions and gradient-based physics constraints. This combination of tools provided end-to-end traceability, from raw data through final model evaluation.

5. Results

We present the results for spatter prediction in this section. Recoater streaking prediction has no significant effect and we would rather do recoater streaking detection soon.

Table 1. Model comparison results

Model	**RMSE**	**R^2**
XGB	19701.42	0.9843
XGB+phys	19729.50	0.9842
PINN	38587.75	0.9397
OLS+phys	39868.18	0.9357
OLS	39868.18	0.9357
MLP	48886.49	0.9033

As we can see, the inclusion of physics knowledge has a significant performance improvement if there is a physics penalty parameter in the loss function compared to just adding the physics derived features in models such as the OLS and XGBoost. In other words, feature engineering physical features did not have a significant performance boost.

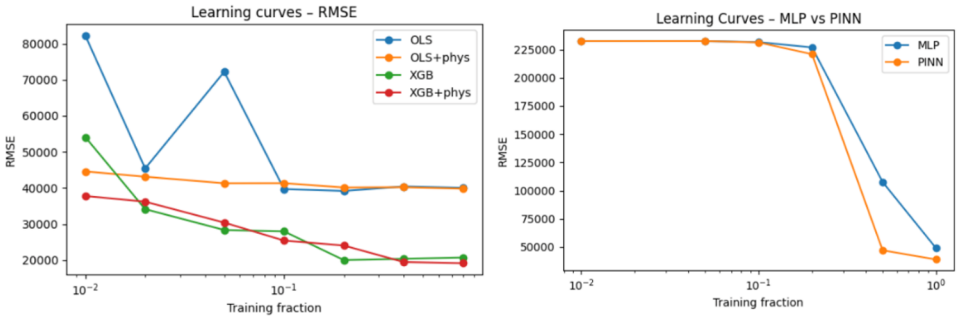

Fig. 6. RMSE for different models on different sizes of training data [0.01, 0.02, 0.05, 0.1, 0.2, 0.4 and 0.8].

Taking a closer look at what the PINN model used to get the results, we look at SHAP value for each feature used by the model in **Fig. 7.**

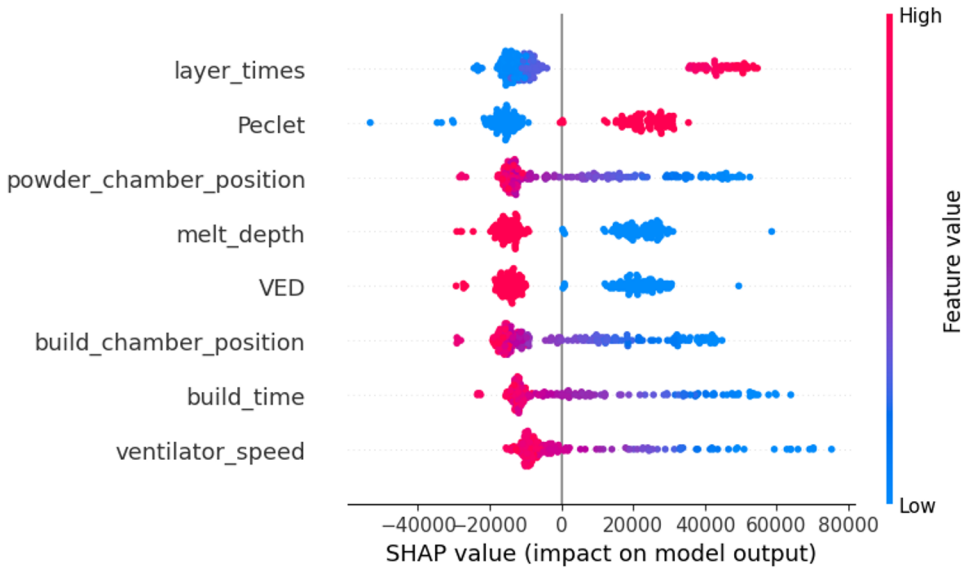

Fig. 7. SHAP values for different features used by the PINN model.

The features are sorted top-to-bottom by their mean absolute SHAP value (so the ones at the top move the prediction the most). Each dot is one test-sample; its x-position is how much that feature "pushed" the model output up (positive SHAP) or down (negative SHAP) for that sample. The color encodes the feature value (blue = low, red = high). So for layer_times, the red dots that we see on the way out on the right and blue dots on the left, actually means that longer layer times strongly drive spatter up, while short layer times push it down.

6. Discussion

In our training and evaluation procedures, we adopted stratified k-fold cross-validation to ensure robust performance estimates across different data splits. For each model - OLS, XGBoost, MLP and PINN - we performed hyperparameter tuning via randomized search (for tree-based and linear models) and Bayesian optimization (for neural architectures),

optimizing key parameters such as regularization strength, learning rate, network depth and the physics penalty coefficient, λ_phys. Model selection was driven by minimizing RMSE and maximizing R^2 on held-out folds, while secondary tasks (e.g. classification or segmentation) employed accuracy, precision, recall, IoU and quantile loss as appropriate to the prediction objective.

We tracked every experiment with MLflow, versioned both data and preprocessing pipelines through DVC, and containerized the entire stack in Docker to guarantee environment consistency. Model implementations leveraged Tensorflow (for PINNs and MLPs) and scikit-learn/XGBoost (for conventional baselines), with artifacts - rained weights, sharding indices, and SHAP explanations - all logged automatically for end-to-end reproducibility.

The SHAP summary for the PINN model reveals that layer_times and the derived Peclet number exert the strongest influence on spatter prediction, with high layer_times (red points) pushing the output upward and high Peclet values pulling it downward. This aligns with physical intuition: longer exposure yields more spatter, whereas more efficient heat diffusion (higher Peclet) mitigates it. By embedding a gradient penalty on the VED input, the PINN enforces monotonic behaviour with respect to energy density, steering the network away from non-physical artefacts seen in the pure MLP's SHAP patterns.

Despite these advantages, our current PINN uses only a simple rectifier linear unit (ReLU)-based slope penalty, which captures monotonicity but not full conservation laws or coupled PDEs. Tuning λ_phys is not as sensitive (we tried values between 0.01 and 10). The expectation was that too small a value of the physics penalty and the model reverts to pure MLP behaviour, too large and it underfits. This was not the case. Future work could integrate richer physics constraints (e.g. energy/mass residuals), explore multi-fidelity data fusion, or incorporate Bayesian inference for uncertainty quantification, all of which promise to further boost both robustness and predictive accuracy in low-data regimes.

7. Conclusion

In this work, we systematically compared purely data-driven regressors (OLS, XGBoost, MLP) against a simple physics-informed neural network (PINN) for predicting spatter in powder-bed fusion. Our key findings are:

- *Data efficiency*: The PINN - with only a slope penalty on the VED feature - reaches comparable RMSE/R^2 to an MLP of identical architecture while using 30-40 % fewer training samples (see MLP/PINN learning curves, **Fig. 6.**).
- *Interpretability*: SHAP analysis shows that the PINN's feature attributions align with known spatter physics, whereas the black-box MLP can highlight spurious correlations.
- *Generalization & robustness*: Despite its simplicity, the PINN suffers less from overfitting than XGBoost and MLP in low-data regimes, and its built-in physics bias cushions against noisy measurements.

Taken together, these results underscore the promise of physics-informed machine learning for metal additive manufacturing defect detection. By embedding even a minimal physics constraint, practitioners can build models that are not only more trustworthy and explainable, but also require fewer costly experiments, paving the way toward faster, safer, and more reproducible AM process qualification.

References

1. W.E. Frazier, Metal additive manufacturing: a review. J. Mater. Eng. Perform. **23**, 1917–1928 (2014). https://doi.org/10.1007/s11665-014-0958-z
2. W.L. Ng, G.L. Goh, G.D. Goh, J.S.J. Ten, W.Y. Yeong, Progress and opportunities for machine learning in materials and processes of additive manufacturing. Adv. Mater. **36**, 2310006 (2024). https://doi.org/10.1002/adma.202310006
3. S. Deshpande, V. Venugopal, M. Kumar, S. Anand, Deep learning-based image segmentation for defect detection in additive manufacturing: an overview. Int. J. Adv. Manuf. Technol. **134**, 2081–2105 (2024). https://doi.org/10.1007/s00170-024-14191-6
4. Y. Fu, A.R.J. Downey, L. Yuan, T. Zhang, A. Pratt, Y. Balogun, Machine-learning algorithms for defect detection in metal laser-based additive manufacturing: a review. J. Manuf. Process. **75**, 693–710 (2022). https://doi.org/10.1016/j.jmapro.2021.12.061
5. H. Zhang, C.K.P. Vallabh, X. Zhao, Influence of spattering on in-process layer surface roughness during laser powder bed fusion. J. Manuf. Process. **104**, 289–306 (2023). https://doi.org/10.1016/j.jmapro.2023.08.058
6. D. Mahmoud, M. Magolon, J. Boer, M.A. Elbestawi, M.G. Mohammadi, Applications of machine learning in process monitoring and controls of L-PBF additive manufacturing: a review. Appl. Sci. **11**, 11910 (2021). https://doi.org/10.3390/app112411910
7. L. Scime, C. Joslin, D. Collins, M. Sprayberry, A. Singh, W. Halsey, R. Duncan, Z. Snow, R. Dehoff, V. Paquit, A data-driven framework for direct local tensile-property prediction of laser powder bed fusion parts. Materials **16**, 7293 (2023). https://doi.org/10.3390/ma16237293
8. Oak Ridge National Laboratory, Laser Powder Bed Fusion Dataset (2025). https://doi.ccs.ornl.gov/dataset/c0247625-951c-5616-a2e3-03803e848896 (accessed 25 February 2025).
9. M. Baehr, T. Klecker, S. Pielmeier, T. Ammann, M.F. Zaeh, Experimental and analytical investigations of the removal of spatters by various process gases during the powder bed fusion of metals using a laser beam. Prog. Addit. Manuf. **9**, 905–917 (2024). https://doi.org/10.1007/s40964-023-00491-y
10. O.H. Aremu, F.S. Alneif, M. Salah, H. Abualrahi, U. Ali, Spatter transport in a laser powder-bed fusion build chamber. Addit. Manuf. **94**, 104439 (2024). https://doi.org/10.1016/j.addma.2024.104439
11. M. Kim, S.J. Han, H.-S. Kang, G.B. Bang, T.W. Lee et al., Optimization of hatch spacing for improved build rate and high-density preservation in laser powder bed fusion of pure titanium. J. Mater. Res. Technol. **33**, 9853–9861 (2024). https://doi.org/10.1016/j.jmrt.2024.11.265
12. J. Xia, P. Berglund, R. Kutty et al., Atmosphere effects in laser powder bed fusion: a review. Materials **17**, 5549 (2024). https://doi.org/10.3390/ma17225549
13. C.F. Montero, A.H. Alvarez, F.M. Calderón, Monitoring of the powder-bed quality in metal AM using a recoater-based line sensor. Wear **496–497**, 204594 (2022). https://doi.org/10.1016/j.wear.2022.204594
14. A. Farrag, Y. Yang, N. Cao, D. Won, Y. Jin, Physics-informed machine learning for metal additive manufacturing. Prog. Addit. Manuf. **10**, 171–185 (2025). https://doi.org/10.1007/s40964-024-00612-1
15. T.L. Bergman, A.S. Lavine, F.P. Incropera, D.P. DeWitt, Fundamentals of Heat and Mass Transfer, 6th ed. (John Wiley & Sons, 2011).

16. S.A. Khairallah, A.T. Anderson, A. Meshold, Ö. Gülsoy, Laser powder-bed fusion additive manufacturing: physics of complex melt flow and formation mechanisms of pores, spatter, and denudation zones. Acta Mater. **108**, 36–45 (2016). https://doi.org/10.1016/j.actamat.2016.02.014
17. A. Simcoe, J. Risse, J. Gockel, S. Daniewicz, Beam shaping in laser powder bed fusion: Peclet number and dynamic simulation. In: Proc. Solid Freeform Fabrication Symposium (2022), pp. 1–6.
18. S.V. Patankar, Numerical Heat Transfer and Fluid Flow (McGraw-Hill, 1980), p. 102.
19. A. Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, 2nd ed. (O'Reilly, 2019).