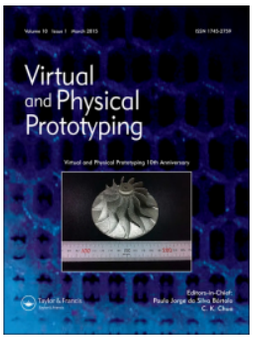


Virtual and Physical Prototyping

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/nvpp20

Enhanced geometrical control in cold spray additive manufacturing through deep neural network predictive models

Roberta Falco, Masoud Jalayer & Sara Bagherifard

To cite this article: Roberta Falco, Masoud Jalayer & Sara Bagherifard (2025) Enhanced geometrical control in cold spray additive manufacturing through deep neural network predictive models, Virtual and Physical Prototyping, 20:1, e2472388, DOI: 10.1080/17452759.2025.2472388

To link to this article: https://doi.org/10.1080/17452759.2025.2472388

Published online: 19 Mar 2025.

VIRTUAL AND PHYSICAL PROTOTYPING
2025, VOL. 20, NO. 1, e2472388 (19 pages)
https://doi.org/10.1080/17452759.2025.2472388

Taylor & Francis
Taylor & Francis Group

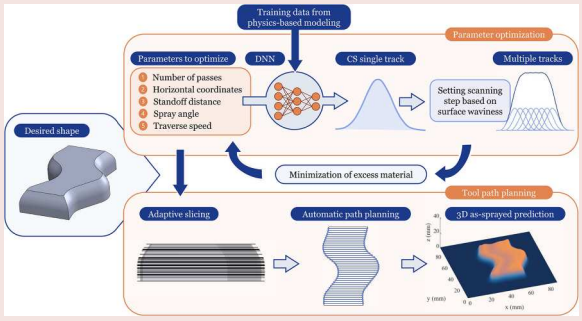
Enhanced geometrical control in cold spray additive manufacturing through deep neural network predictive models

Roberta Falco[a], Masoud Jalayer[b,c] and Sara Bagherifard[a]

[a]Department of Mechanical Engineering, Politecnico di Milano, Milan, Italy; [b]Department of Management Engineering, Politecnico di Milano, Milan, Italy; [c]Department of Materials and Mechanical Engineering, University of Turku, Turku, Finland

ABSTRACT

Cold spray additive manufacturing is a deposition technique that facilitates the fabrication of large metal components with limited thermal effects, making it suitable for a wide range of industrial applications. Despite its potential, achieving precise geometrical control remains a bottleneck, hindering cold spray's establishment as a competitive additive manufacturing technology. This study introduces a computationally efficient framework that combines an adaptive slicing algorithm and process-specific toolpath planning strategies, designed to optimise deposit accuracy and material efficiency with respect to the Standard Tessellation Language (STL) model of the part to fabricate. Central to this approach is the integration of predictive models for cold spray deposition, which utilise deep neural networks trained on data from physics-based analytical models. These models offer rapid and accurate predictions of single-track cross-sections and full 3D shapes. The adaptive slicing algorithm dynamically adjusts layer thickness based on local curvature variations, ensuring improved geometrical fidelity while minimising material waste. Additionally, the toolpath planning methodology ensures continuous deposition, effectively addressing challenges such as surface waviness and edge losses. Validated against experimental data, the framework demonstrates significant improvements in efficiency and accuracy over conventional approaches, paving the way for broader adoption of cold spray additive manufacturing in complex industrial applications.



ARTICLE HISTORY
Received 26 October 2024
Accepted 21 February 2025

KEYWORDS
Solid-state additive manufacturing; non-beam-based additive manufacturing; deep learning; toolpath optimisation; data-driven

1. Introduction

Additive manufacturing (AM) is a modern manufacturing process that fabricates objects layer by layer based on digital models. Since its inception, it has transformed industries by enabling the creation of complex geometries, reducing material waste, and allowing for high precision and customisation. Among the most widely used AM technologies are beam-based methods such as directed energy deposition (DED) and powder bed fusion (PBF), which rely on melting or sintering materials to create detailed parts [1–3]. However, Cold Spray (CS), a solid-state deposition technology, has recently emerged as a versatile alternative due to its ability to achieve high deposition rates while minimising thermal effects [4,5].

The working principle of CS was discovered in the 1980s [6–9]. Unlike beam-based methods, CS deposition is achieved by propelling micron-size (10−100 µm) metallic particles, carried by a heated inert gas, through a supersonic nozzle. When the particles reach a critical velocity, their kinetic energy allows them to form metallurgical bonds and mechanical interlocking with the

CONTACT Sara Bagherifard sara.bagherifard@polimi.it Via La Masa, 1, 20156 Milan (MI), Italy

substrate upon impact, without in-flight melting. This unique process results in dense metallic layers, free from the negative thermal effects typically associated with high-temperature methods [10].

Beyond minimising thermal effects, CS also offers scalability, enabling the fabrication of large deposits with a broad range of feedstock materials. Initially developed for protective coatings and repairs, CS has evolved into Cold Spray Additive Manufacturing (CSAM), allowing the production of complex components while leveraging its high deposition efficiency and material versatility [11,12].

Despite the advantages, deposit shape control represents a significant challenge for the further development of CSAM. In particular, compared to well-established AM technologies like DED or PBF [13–15], CSAM presents a much larger feature size. While deposit thickness can range from as thin as 0.05–10 µm [16], commercially available CSAM systems generate spray spots between 5 and 10 mm in width [17]. These spots typically exhibit a Gaussian unitary shape that is highlysensitive variations in robot kinetic parameters including standoff distance (SoD), scanning speed, or spray angle [18–20]. This inherent shape limitation restricts the precision required for intricate detailing or sharp edges, which are often essentialin AM parts [16].

While these limitations pose challenges, they also highlight opportunities for improvement through the development of optimised spray strategies and predictive models [5]. Understanding how variations in spraying parameters influence the resulting geometry often requires extensive experimental trials, especially in complex scenarios. In contrast, accurately predicting the final shape of a fully fabricated part through simulation can streamline the process planning by minimising the need for iterative approaches. Predictive modelling relies on a comprehensive understanding of how process parameters, feedstock properties, and nozzle kinematics interact. Addressing the challenges in modelling deposit shape represents a significant step toward automating the CSAM process.

A few modelling approaches, both analytical and data-driven, have been developed to tackle these issues. Among those targeted to CSAM, Li et al. [21] developed a general-purpose model, based on the superimposition of spray spots onto a generic substrate and the representation of the final geometry through a mesh reconstruction procedure. In this instance, the spray spots were modelled as Gaussian, fitting layer thickness and deposition efficiency (DE) to experimental data as a function of local particle impact angles. The framework is able to handle complex trajectories and shadow effects. On the other hand, the mesh reconstruction procedure leads to redundant faces and vertices, with an overall irregular result. In a previous paper from our group, we relied on a similar approach, introducing the effect of SoD on the spot shape and implementing a more physics-based representation of mass conservation at non-perpendicular impact angles. Furthermore, this model did not rely on mesh reconstruction, but rather on its initial refinement, granting a smoother result [22].

Both aforementioned deposition models rely on spot-by-spot superimposition, following an iterative time-step approach to calculate substrate normal vectors in the jet area as the gun's position changes. Consequently, their computational time is proportional to the total deposition duration, i.e. to the part size and complexity. Simulating the full 3-dimensional (3D) shape of a complex part using these models can take upwards of a day, which is not ideal during path and parameter optimisation processes. This can consequently compromise the applicability of automated path optimisation for CSAM.

Data-driven models have been introduced as a faster alternative to analytical models for deposit shape modelling, leveraging the speed of machine learning algorithms. Ikeuchi et al. [23] used a feedforward artificial neural network, or multilayer perceptron (MLP), to predict the cross-sectional profile of a single track of CS as a function of SoD, scanning speed, and spray angles. The application was based upon very limited experimental data, consisting of 36 single-track profiles, therefore lacking flexibility outside the parameter dominium of the available profile acquisitions. To address this issue, the authors developed a data-efficient framework [24], augmenting their dataset with information extracted through Gaussian fits of the experimental profiles. In a subsequent study [25], the same research group enhanced predictive performance by employing a Gaussian Process Regression (GPR) model. This model incorporated prior knowledge from Gaussian models via an explicit mean function. The new approach offers the possibility of predicting the shape for overlapping tracks, while neglecting the effect of the spray angle. A larger experimental dataset, consisting of 330 single-track profiles, was acquired by Liu et al. [26], who similarly fed it in a MLP. In this case, the substrate profile, SoD, scanning speed, and current layer number were identified as input features to obtain the cross-sectional single-track profile.

Despite the promising results, current data-driven approaches are limited to cross-sectional predictions of single or few tracks for very few input variables; thus, they lack the flexibility needed to predict 3D shapes

Table 1. Comparison of shape prediction models in the literature.

Study	Method	Inputs	Output	Dataset	Limitations
Li et al. [21]	General-purpose model with mesh reconstruction	Substrate shape, scanning speed, spray angles	3D shape reconstruction of CS deposits	Experimental data (amplitude and DE)	Irregular results with redundant faces and vertices, computationally intensive
Vanerio et al. [22]	Physics-based model with initial mesh refinement	SoD, substrate shape, scanning speed, spray angles	3D shape reconstruction with smoother results	Experimental data	Computationally intensive
Ikeuchi et al. [23]	Data driven (MLP)	SoD, scanning speed, spray angles	Cross-sectional profile of a single track	Limited experimental data	Lacks flexibility outside the parameter domain
Ikeuchi et al. [24]	Data driven (MLP with data augmentation)	SoD, scanning speed, spray angles	Cross-sectional profile of a single track	Augmented experimental data	Limited by experimental profile acquisitions
Ikeuchi et al. [25]	Gaussian Process Regression with gaussian mean function	Scanning speed, surface type, substrate height	Cross-sectional profile of overlapped tracks	Limited experimental data	Lack of modelling of spray angles and complex scenarios
Liu et al. [26]	Data driven (MLP)	Substrate profile, SoD, scanning speed, current layer number	Cross-sectional profile of a single track	Large experimental dataset	Not suitable for 3D multi-track shape prediction

that require the interaction of several tracks. While some efforts have been made to extend predictions to multi-layer profiles [26], overlapping-track modelling [25], and deposit profile prediction from process parameters [27], these approaches remain constrained to localised predictions and do not fully capture the complex interactions necessary for full 3D shape reconstruction. Table 1 summarises the contributions and limitations of existing shape prediction models, highlighting these challenges in greater detail

To fully harness the potential of CSAM, integrating shape prediction models with optimised tool path planning is crucial to addressing edge losses, surface waviness, overspray, and parameter selection [28]. These challenges are exacerbated by CSAM's unique characteristics, including limited precision due to spray spot size and the difficulty of maintaining uniform material distribution during continuous deposition [29].

Shape prediction models, such as those discussed earlier, provide a foundation for understanding how deposition parameters influence geometry. Coupled with advanced path planning, these models can enable precise geometrical control, reduce reliance on iterative experimental trials, and enhance process efficiency. Tool path planning is particularly critical in CSAM due to its reliance on continuous deposition and the sensitivity of the process to nozzle kinematics. Valuable tool path optimisation frameworks have recently been developed for CS in different application fields. Nault et al. [30] used a constrained gradient descent method to optimise nozzle speed to obtain the desired layer thickness to repair a leading edge. This method showed promising results, despite the complex substrate geometry. Li et al. [29] developed an automatic path generation algorithm for AM of a multi-featured component, devising optimal paths to minimise overspray when linking said features. Such workflow was then generalised to the fabrication of rotational and freestanding parts. Despite its focus on AM, the approach by Li et al. is limited in scope. It does not optimise several critical kinematic parameters, such as scanning speed and scanning step, which play a pivotal role in determining the final deposit shape. Consequently, achieving the desired geometry still relies heavily on operator expertise and trial-and-error adjustments. This reliance highlights a significant gap in current methodologies, underscoring the need for more comprehensive optimisation frameworks that fully integrate predictive models with tool path planning.

Outside of CSAM, recent advancements in AM have emphasised the integration of surrogate models to address challenges in process optimisation and product design. These models often combine data-driven machine learning techniques with physics-based principles, offering enhanced interpretability and performance, particularly for high-dimensional or sparse data scenarios. For example, Bayesian neural networks have been applied to optimise mechanical properties in metamaterials and design seismic-resistant structures [31]. Machine learning has been extensively used to predict surface roughness in MEX and PBF [32,33], model bead geometry in DED [34], and estimate thermal deformation in PBF [35]. These efforts primarily focus on process-structure relationships, leveraging techniques such as random forests, AdaBoost, and MLP to capture dependencies between process parameters and material responses. However, while these models have been successful in specific applications, their integration with optimisation strategies – such as adaptive slicing and automated tool path planning – for CSAM remains significantly limited. In summary, the main approaches in the literature to CS deposition modelling are either data-driven or physics-based. A significant drawback of data-driven methods is their reliance on vast amounts of data to reach reliable predictive performances. Such datasets are hard to obtain for

emerging technologies due to short literature history and processing costs hindering large experimental campaigns. Available CS shape prediction models that use MLPs have had to face this issue, working with only a few data points, resulting in less dependable predictions [23,24,26]. On the other hand, analytical models based on physical principles often rely on cumbersome calculations, such as iterative solvers, therefore requiring significant computational power to perform complex simulations. A few simple tool trajectory planning algorithms have also been developed to control CS shape, but they have yet to be integrated with prediction models to achieve a full optimisation framework.

The present study proposes a shape control framework that streamlines the transition from the Computer-Aided Design (CAD) model of a desired component to the robot tool path to fabricate it with CSAM. Within such framework, three key contributions are delivered:

- CS Shape Prediction Models: These models deliver both 2D (cross-sectional) and 3D deposit predictions by combining the precision of analytical formulations with the computational efficiency of Deep Neural Networks (DNNs). This hybrid approach mitigates the high computational costs of traditional physics-based models while reducing the reliance on extensive experimental datasets typical of purely data-driven methods.
- Adaptive Slicing Algorithm: A novel, computationally efficient adaptive slicing algorithm dynamically adjusts layer thickness to accommodate variations in the target geometry along the build direction. This algorithm operates in direct synergy with the shape prediction models.
- Automated Tool Path Planning: Tailored to the unique requirements of CSAM, this algorithm incorporates continuous deposition and accounts for nozzle kinematic effects. By integrating shape predictions, it optimises trajectory parameters, ensuring precise material deposition and minimising errors such as surface waviness.

2. Shape prediction model: materials and methods

2.1. Data acquisition and preparation

The present work involves the development of two distinct DNNs, predicting the 2D cross-sectional shape and the full 3D shape of a linear CS track.

The selection of the DNN model was based on a comprehensive comparison of various machine learning algorithms and deep learning architectures, including Transformers, Convolutional Neural Networks (CNNs), different Long Short-term Memory (LSTM) architectures, a gradient boosting, and Gaussian Mixture Models (GMMs), as detailed in Section 2.4. The DNN demonstrated superior performance in terms of accuracy while maintaining a reasonable runtime, making it the optimal choice for this study.

Table 2. Ti6Al4V powder particle size distribution.

Powder	D10 [μm]	D50 [μm]	D90 [μm]
Ti6Al4V	21	37	47

The DNNs were trained on an array of single-track shapes of CS deposits, generated through our previously developed physics-based analytical model [22], which will hereby be briefly introduced. In the analytical model, the distribution of particles within the jet is described by a function $J(r)$, easily adaptable to any jet distribution. In the case of Gaussian-like distributions, it is structured as follows:

$$J(r) = \begin{cases} \dfrac{e^{-k_2(r/r_n)^2} - e^{-k_2}}{1 - e^{-k_2}}, & r < r_n \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

The parameters r_n and k_2 respectively represent the radius of the distribution and its rate of decay at the edges. The model relies on the definition of DE as a function of particle local impact angle α, where each particle is assumed to fly parallel to the nozzle axis, as seen in [20,36]. The effect of SoD on deposit shape is modelled by stretching and scaling the particle's distribution profile function of Eq. 1, while ensuring the conservation of the mass of the sprayed material. Moreover, also DE is affected by SoD, therefore a DE_d (d_s) function is experimentally generated. Putting all these factors together, the thickness growth rate of the profile along the direction of the nozzle axis is presented in Eq. 2.

$$\frac{\partial z_{axis}}{\partial t} = A \cdot DE_\alpha(\tan\alpha) \cdot DE_d(d_s) \cdot J(r, d_s) \quad (2)$$

Coefficient A represents the growth rate of deposit height, and it is estimated by fitting experimental data to each combination of primary spray parameters and feedstock material. Similarly, the stretching and scaling coefficients are experimentally obtained. Since the particle impact angle α changes as the deposit grows, it is

Table 3. CS process parameters assumed constant for deposit shape characterisation.

Powder feed rate [g/min]	Carrier gas	Gas temperature [°C]	Gas pressure [MPa]
24.25	N_2	1100	5

Table 4. CS nozzle parameter levels selected for the full-factorial data generation process.

Scanning speed [mm/s]	Spray angle [°]	SoD [mm]	Number of passes
150, 250, 350	60, 70, 80, 90	20, 25, 30, 35, 40	1 to 40

obtained from profile partial derivatives. The analytical model [22], consisting of a series of partial differential equations (Eq. 2), is solved in MATLAB R2023b by applying Euler's method, using the STL mesh of a flat substrate as an initial condition. The code iteratively calculates the element surface normals within the nozzle jet, consequently obtaining the local particle impact angle α.

Ti6Al4 V feedstock (AP&C, Canada) was the selected material for this work, with powder particle size distribution shown in Table 2 and constant CS process parameters reported in Table 3. This material was selected due to its wide application in CSAM, its fairly regular (the typical Gaussian) deposit shape, and due to the data availability from our previous studies [22,37].

To obtain robust predictions, a notably large design of experiment (DoE) matrix was selected to run the simulations. The four selected nozzle parameters to vary were: scanning speed, nozzle spray angle, SoD, and the number of overlapping passes. The values for each parameter were distributed through a typical working range of CS systems, with a full-factorial combination of the levels reported in Table 4, for a total of 2400 linear single tracks. The size of this dataset is therefore more than 7 times larger than the largest comparable MLP available in the literature in terms of number of the profiles [26] and it describes a wider range of process variables [23,24,26]. The maximum element size for the substrate mesh was set to 0.05 mm to accurately capture the final shape of the track.

Therefore, the simulation was run 2400 times and, each time, the nozzle moved at the set SoD, spray angle, and scanning speed along a linear 30 mm track for a number of times equal to the number of passes. An example of a generated 3D track is reported in Figure 1.

For the first DNN model, the objective was to predict the cross-sectional (2D) profile of a single track. Therefore, the generated 3D track was sliced in its middle with a *yz* plane, according to the reference system depicted in Figure 1. This means that, for each track in the dataset, the deposit height (*z* direction) was extracted for each *y* coordinate, with a spacing of 0.05 mm, corresponding to the mesh resolution. The *y* coordinate (horizontal) was treated as an input feature for the DNN, alongside the nozzle parameters of Table 4, to render the prediction mesh-independent and adaptable to the required accuracy. The prediction output will therefore be a single value corresponding to the z coordinate of the profile at the y coordinate specified in the input. Figure 2(a) presents a schematic representation of the 2D DNN's inputs and outputs.

Similarly, for the second DNN model, an additional input was added, representing the transversal coordinate (*x* direction), to also characterise the deposit height variation along the scanning track (Figure 2(b)). The output is still a single value, corresponding to the profile height at the set transversal and horizontal coordinates. The additional dimension implies a notably larger dataset, comprising, for each of the 2400 lines, of the deposit height for each point in the mesh. The dataset was therefore randomly split into 500 batches to mitigate the amount of allocated memory during training. All features were normalised according to Eq. 3, ensuring they were on the same scale to stabilise

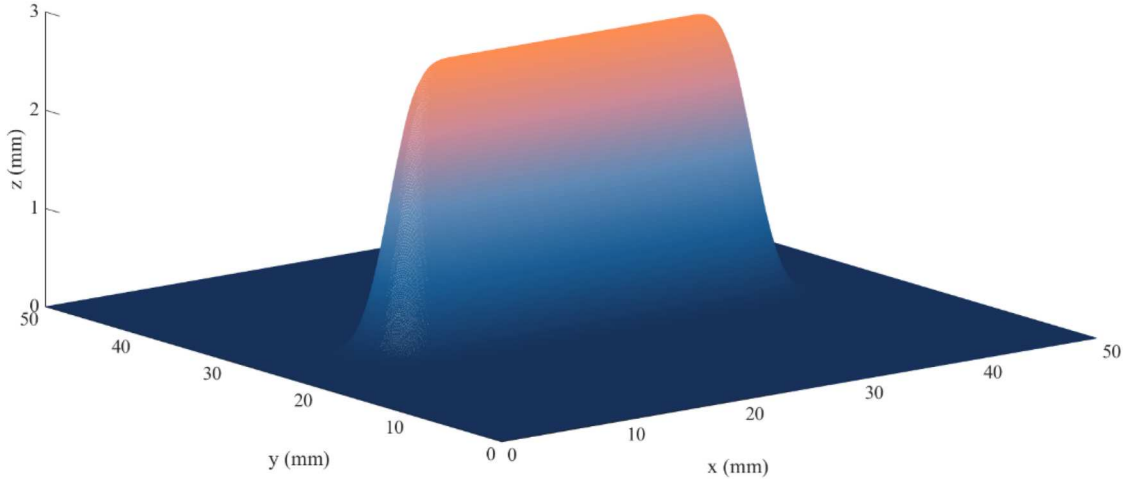


Figure 1. A representative example of the simulated 3D construction of a single track considering 250 mm/s scanning speed, 90° spray angle, 25 mm SoD and 30 passes.

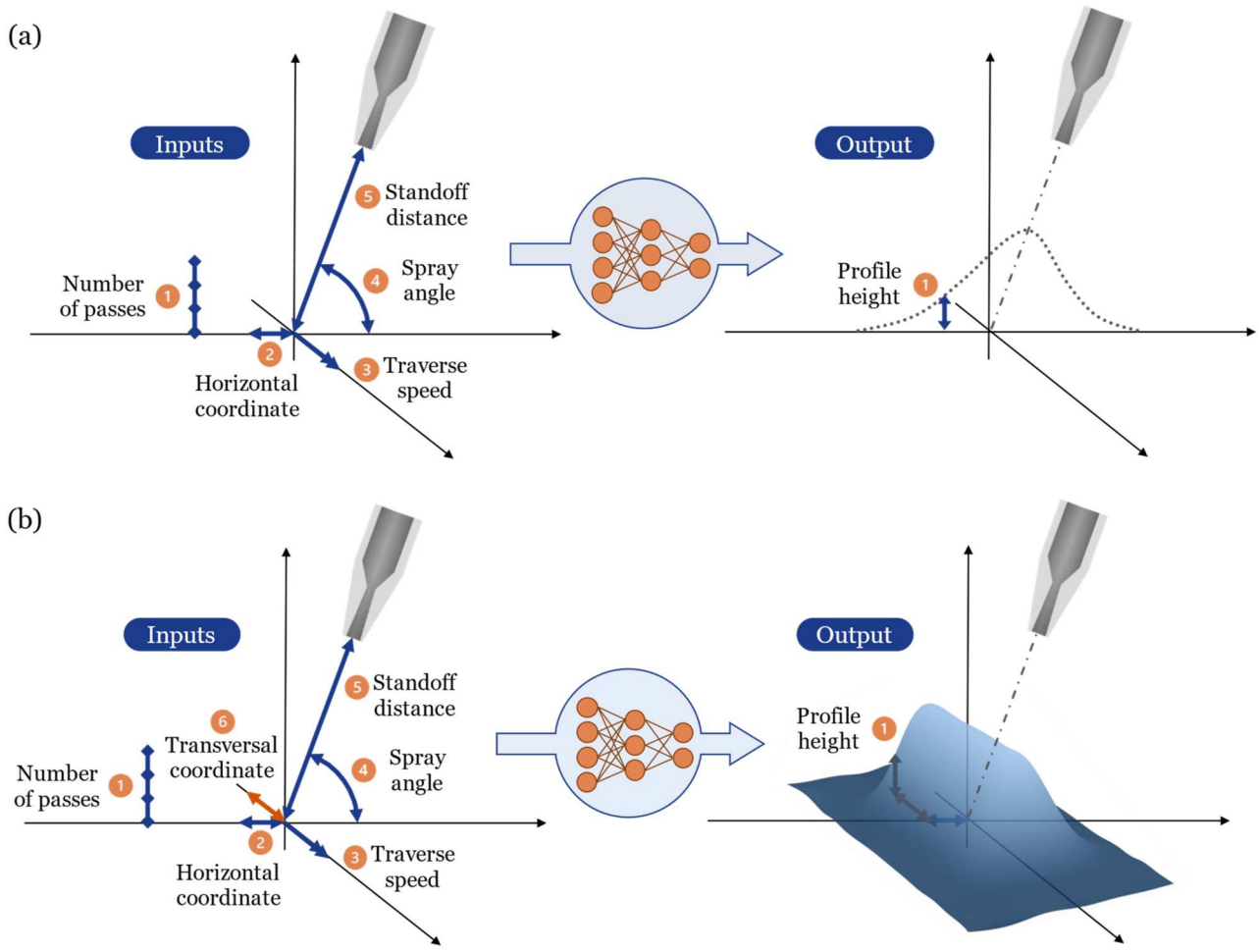


Figure 2. Schematic representation of the DNN architecture for the 2D configuration (a) and 3D configuration (b).

the DNN's learning process. Lastly, the datasets were randomly split in 70:15:15 proportions for training, validation, and testing.

$$f_{normalised} = 2 \cdot \frac{f - min(f)}{max(f) - min(f)} - 1 \tag{3}$$

2.2. Deep neural network architectures

The performance of DNNs heavily relies on the appropriate selection of architectural hyperparameters. It is particularly crucial to balance between the depth and the width of a DNN, namely regulating the number of hidden layers and the number of neurons per layer. For this purpose, a hyperparameter tuning procedure was carried out with the Optuna framework [38], sampling through 1–6 hidden layers and 4–500 neurons per layer. A Tree-structured Parzen Estimator (TPE) sampler was employed, which is a Bayesian optimisation algorithm. The TPE starts the hyperparameter optimisation with an initial belief (prior) surrounding the best configuration and updating such belief based on the performance results. The set prior for both the number of hidden layers and the number of neurons per layer was logarithmic, expecting a superior performance with increased complexity. A total of 300 trials was performed and the configuration with the lowest relative percentage L^2 error on the testing dataset was selected as optimal. This error represents the prediction error, normalised by the magnitude of the true data, making it independent from scale and magnitude of the dataset, and therefore highly comparable. Moreover, this metric penalises larger errors, which is suitable given the generally small size of CS profiles. The expression is as follows:

$$Relative\ L^2\ error = \frac{||Y_{test} - \hat{Y}_{test}||_2}{||Y||_2}, \tag{4}$$

where Y_{test} is the observed profile height in the testing dataset, $\hat{Y}_{test}$ is the predicted profile height on the testing dataset inputs, and Y is the vector of observed profile heights.

The DNN models were defined, constructed, and trained in Python using TensorFlow's [39] high-level

API, TensorFlow Keras. Rectified linear unit (ReLU), sigmoid, and hyperbolic tangent activation functions were all tested, and the latter was ultimately selected for all layers since it yielded the best accuracy. Limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) algorithm [40], a second-order quasi-Newton method, was selected for training optimisation. This algorithm is generally memory-efficient, ensuring faster convergence compared to other methods. The mean squared error (MSE) was used as a loss function for training:

$$MSE = \frac{1}{n}\int_{i=1}^{n} (Y_i - \widehat{Y}_i)^2, \tag{5}$$

where n *is* equal to the number of the samples considered, Y is the vector of the observed profile heights and $\widehat{Y}_i$ is the vector of the predicted profile heights. Training and validation losses were monitored throughout the training process to avoid overfitting on the training dataset.

2.3. Output post-processing

DNNs are function approximators, therefore, despite their powerful predictive capabilities, physical boundary conditions are not rigidly enforced. Thus, post-processing corrections were devised to make sure that the extremities of the CS deposit track reached the substrate height exactly, i.e. a value of zero profile height for the present assumption of flat substrate. A cosine-tapered (Tukey) window, described in Eq. 6, was applied to the predicted profiles. The shape of the window function is such that the prediction in the middle of the track will remain unchanged, while the extremities will smoothly converge to zero with no discontinuities.

$$\begin{cases} w(n) = \frac{1}{2}\left[1 - \cos\left(\frac{2\pi n}{\beta N}\right)\right], & 0 \le n < \frac{\beta N}{2} \\ w(n) = 1, & \frac{\beta N}{2} \le n \le \frac{N}{2} \\ w(N-n) = w(n), & 0 \le n \le \frac{N}{2} \end{cases} \tag{6}$$

The value of β in the above formula is in the interval [0,1], where $\beta = 0$ is a rectangular window and $\beta = 1$ is a Hann window. In this case, β was chosen equal to 0.4, based on the observed width of the track. An example of the result of post-processing is shown in Figure 3(a) for the 2D case. For the 3D case, the window was similarly applied to correct the extremities in the transversal direction with a lower $\beta = 0.2$, due to the longer length of the track.

Moreover, as the 3D input tracks had a set length of 30 mm, the variation of such length can be obtained through stretching and compressing of the track in its middle section, which presents a constant cross-section.

2.4. Comparative validation of the deep neural network model

The study evaluates several machine learning architectures (summarised in Table 5), each chosen based on their distinct advantages and characteristics. The Transformer model was included for its exceptional ability to capture complex sequential dependencies, making it ideal for modelling intricate relationships among parameters [41]. The Convolutional LSTM (Conv-LSTM) and its attention-enhanced variant were selected for their capacity to model spatiotemporal dependencies,

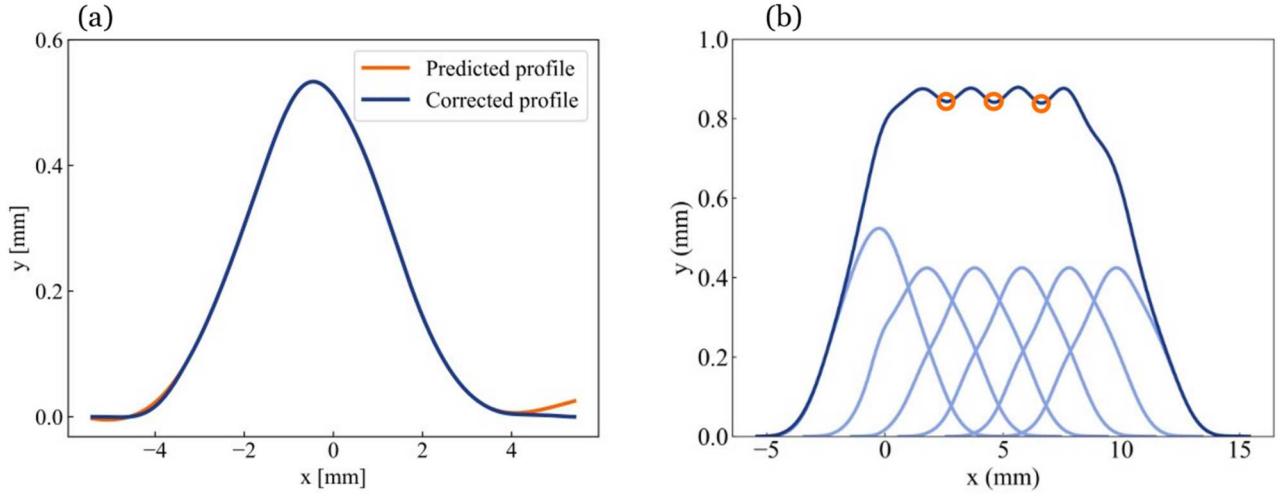


Figure 3. 2D prediction post-processing: (a) plot of the extremity correction applied to the 2D model, (b) cross-sectional view of six parallel tracks (light blue), their superimposition (dark blue) and the identified local valleys (orange).

Table 5. Overview of the various machine learning architectures implemented and compared in this study and their corresponding key parameters.

Model	Architecture Overview
Transformer	Input Dense layer for feature embedding, 3 Transformer blocks with Multi-Head Attention (4 heads), Feedforward layers, residual connections, Layer Normalisation, Flatten, and Dense layers (128 units) for regression.
Conv-LSTM-Attention	Two ConvLSTM2D layers (kernel size: 1×1) with l2 regularisation, Batch Normalisation, Attention mechanism on reshaped spatial dimensions, residual connections, and Dense layers with Dropout (0.3 and 0.2).
Conv-LSTM	Two ConvLSTM2D layers with 32 and 64 filters (kernel size: 1×1), Batch Normalisation, Dense layer (128 units), Dropout (0.2), and a final Dense(1) for regression.
Bidirectional LSTM	Two Bidirectional LSTM layers with 64 units each, followed by Dropout (0.1), Layer Normalisation, Dense layers (128 and 64 units), another Dropout (0.2), and a final Dense(1) for regression.
CNN1	Two Conv1D layers with 32 and 64 filters (kernel size: 3), followed by Dropout (0.2), Dense layer (128 units), another Dropout (0.2), and a final Dense(1) for regression.
CNN2	Three Conv1D layers with increasing filters (32, 64, 128) and kernel size 3, followed by Dropout after each convolutional block, Dense layer (128 units), Dropout (0.2), and a final Dense(1).
XGBoost	Gradient boosting with a maximum tree depth of 6, learning rate (η) of 0.1, subsample ratios of 0.8 for rows and columns, and an objective function for squared error regression.
GMMs	GMM with 10 components, using component-wise means for regression predictions.

enabling effective handling of multi-dimensional data [42]. Two Convolutional Neural Networks (CNNs) with different sets of hyperparameters were included due to their efficiency in extracting spatial features from input data, offering simpler yet effective modelling approaches [43]. XGBoost, a gradient boosting ensemble method, was chosen for its ability to efficiently model non-linear patterns through its ensemble of weak learners and its lightweight computational requirements [44]. The DNN was selected as a baseline due to its simplicity, flexibility, and demonstrated success in previous studies involving similar datasets. Additionally, Gaussian Mixture Models (GMMs) were included to compare the performance of probabilistic unsupervised models with supervised approaches. Lastly, the Bidirectional LSTM was considered for its strength in capturing sequential dependencies in both forward and backward directions, providing a comprehensive understanding of time-dependent patterns [45].

Table 6. The performance of tested models with respect to the analytical results.

Model	MSE	R^2	MAPE (%)	Runtime (s/instance)
Transformer	$1.86 \cdot 10^{-3}$	0.998	29.0	$5.78 \cdot 10^{-4}$
Conv-LSTM-Attention	$2.42 \cdot 0^{-3}$	0.997	25.1	$2.81 \cdot 10^{-4}$
Conv-LSTM	$1.24 \cdot 10^{-3}$	0.998	49.7	$1.72 \cdot 10^{-4}$
Bidirectional LSTM	$1.73 \cdot 0^{-3}$	0.998	24.2	$2.80 \cdot 10^{-4}$
CNN1	$7.47 \cdot 10^{-3}$	0.991	56.1	$1.41 \cdot 0^{-4}$
CNN2	$4.80 \cdot 10^{-3}$	0.994	52.1	$7.09 \cdot 10^{-5}$
XGBoost	$4.13 \cdot 10^{-4}$	0.999	22.2	$1.74 \cdot 10^{-7}$
GMMs	1.38	−0.742	184.8	$1.15 \cdot 10^{-6}$
DNN	$8.06 \cdot 10^{-5}$	0.999	16.8	$1.88 \cdot 10^{-5}$

The results in Table 6 indicate that the DNN outperformed all other models, achieving the lowest MSE and MAPE while maintaining a significantly faster inference runtime. While XGBoost and Conv-LSTM models displayed competitive accuracy, they did not match the DNN's optimal balance between predictive performance and computational efficiency. Transformer and ConvLSTM-Attention models, despite their accuracy, exhibited longer runtimes and higher MAPE, making them less practical for real-time applications. Notably, GMMs performed poorly, underscoring the limitations of unsupervised models in capturing complex, non-linear relationships in the dataset. The relatively poorer performance of more complex models, such as Transformers and ConvLSTM-Attention, can be attributed to overfitting due to their higher parameter counts and the limited size of the training dataset. These findings validate the DNN as the preferred model, providing an ideal trade-off between predictive accuracy and runtime efficiency, which is crucial for real-time optimisation in CSAM.

3. Optimised adaptive slicing algorithm

The predictive speed of the developed DNN models enables the use of shape prediction within larger process optimisation frameworks. The objective is to select the optimal nozzle trajectory and parameters to achieve geometrical accuracy for the desired AM component. The first step of the path design involves slicing the STL model of the part along specific planes, to obtain the cross-sections to then be filled in.

In CSAM, the slice height will depend on the overall deposit thickness and, therefore, on the selected nozzle parameters. Therefore, herein, an adaptive slicing technique was devised to determine the slice height based on the occurrence of lateral curvatures in the part. This technique is commonly utilised in various established AM technologies to produce complex geometries and minimise the staircase effect [46]. However, its application to CSAM is currently very limited.

3.1. Slice height definition algorithm

The slice height to be defined in the optimisation procedure must be related to the CS deposit thickness, which is a function of robot kinetic parameters for a specific feedstock. For optimal DE, during the main building phase, it is convenient to set the spray angle to 90° (vertical position of the nozzle with respect to the substrate) and the SoD to 30 mm. The nozzle parameters to be optimised include scanning speed and the number of passes per slice. Additionally, the distance between the scanning lines, known as the scanning step, is another critical path parameter that is optimised during this phase.

The targets selected for optimisation are the amount of overspray, for lower material waste, and the total processing time, for higher process efficiency. The optimal parameters offer a trade-off between these two objectives. The amount of overspray may be reduced by increasing the scanning speed and producing thin layers. Still, a higher number of slices may be inefficient both computationally and in terms of spraying time.

Numerically, the excess material is quantified as excess height h_{exc} in the direction of deposit growth, calculated as:

$$h_{exc} = n_{slices} \cdot h_{slice} - h_{CAD}, \tag{7}$$

where n_{slices} and h_{slice} are respectively the number of slices and the selected slice height, while h_{CAD} is the maximum height of the CAD model of the desired part. The number of slices is, in turn, calculated rounding up the model height divided by the slice height.

$$n_{slices} = \frac{h_{CAD}}{h_{slice}}, \tag{8}$$

The slice height is the parameter dependent variable that can be obtained using the 2D DNN model. The predicted profiles of several single tracks of CS must be superimposed at a set scanning step to obtain the overall deposit height, as shown in Figure 4. The process of superimposing multiple tracks must account for the interaction between the adjacent tracks. For instance, when depositing the second tracks, part of it will overlap with the first. To address this, the deposit profile tangents are calculated after depositing each track. These tangents help determine the impact angle for the subsequent one, which will be input in the 2D DNN, ensuring an accurate representation of the overall deposit profile.

As a first trial, the scanning step is set to one sixth of a single-track profile width, corresponding approximately to the Gaussian's standard deviation. The deposit surface waviness (w) is therefore calculated as:

$$w = 1 - \frac{valley}{peak}, \tag{9}$$

where the maximum height of the deposit corresponds to its peak, while valley height can be identified through local minima, as highlighted in orange in Figure 3(b). A waviness threshold can be set according to the application requirements, with the scanning step being adjusted consequently.

The processing time was assumed to be directly proportional to the total number of passes and inversely proportional to the scanning speed. The final cost

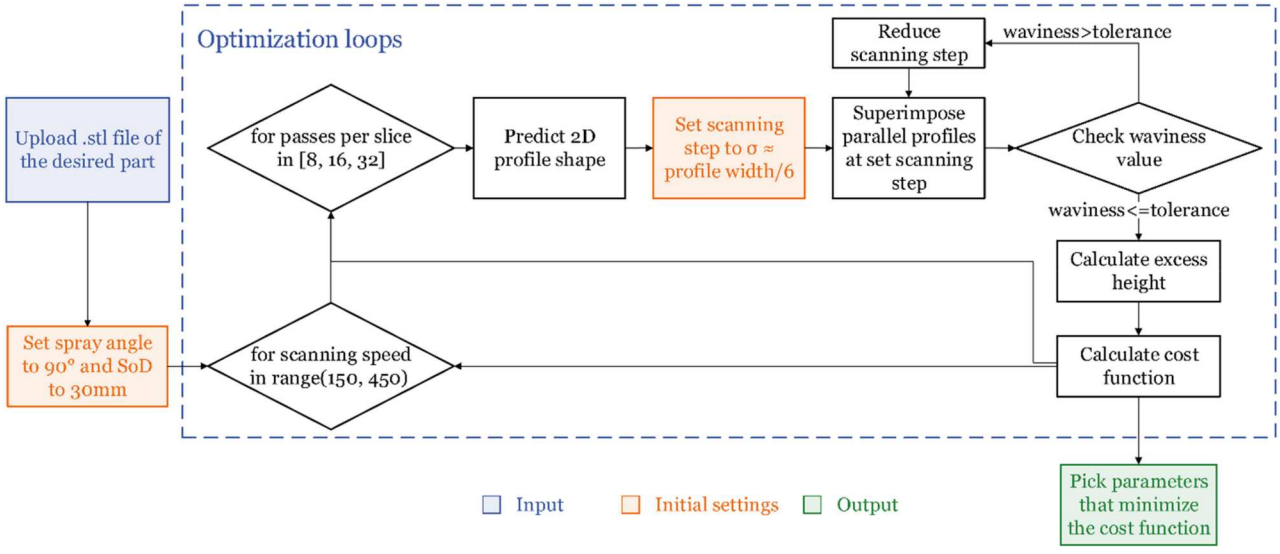


Figure 4. Flowchart of the parameter optimisation algorithm powered by the DNN predictions.

function was a linear combination of the two targets, each multiplied by a normalszation factor to ensure consistent scales. The full optimisation workflow is represented in Figure 4, culminating in the selection of the optimal kinematic parameters that minimise the defined cost function.

3.2. Adaptive slicing algorithm

The slice height optimisation algorithm described in the previous subsection (3.1) is only able to evaluate accuracy in the deposition direction. To guarantee geometrical accuracy with respect to the lateral surfaces, an adaptive optimisation was introduced, which refines the slice height around areas with strong curvatures, ensuring the minimisation of staircase effects.

In the present work, the procedure to detect lateral geometrical discontinuities relies on the identification of local cross-sectional variations. A first-approximation slicing is performed, where the optimal deposit thickness, as calculated in section 3.1, defines a constant slice height. The points of the STL model belonging to each slice plane can be identified according to traditional algorithms, as described in [29,47].

The next step for slice characterisation is contour identification. This is mostly relevant for the first slice, which corresponds to the part's base surface. Indeed, if the slice plane intersects a surface of the part, the internal vertices will also be identified as belonging to the slice. In such cases, the alpha-concave hull algorithm [48] was selected for contour identification. The alpha-concave hull is a generalisation of the convex hull that allows for concavities by introducing a parameter α, which controls the level of detail in contour representation. Given a set of points P, the alpha-concave hull $CH_\alpha(P)$ is defined such that any edge of the hull satisfies a local empty region condition based on α, effectively filtering out short, non-representative connections. By adjusting α, the algorithm balances between capturing fine geometric details and maintaining computational efficiency. The contour points, once identified, are sorted by proximity.

Once slice contours are defined, cross-sectional variations must be identified. This could be achieved through the calculation of normal vector variations for the lateral surface [49]. A more straightforward approach is to calculate the relative internal area variation between consecutive slices, as seen in [49]. However, this method is only appropriate for parts with smooth cross-sectional changes. It may not be suitable for parts with local geometry variations or for swept profiles that have a constant cross-section but variable curvature. For the latter cases, an alternative procedure was devised to capture localised geometric variations efficiently.

The deposition plane was divided into quadrants using a structured grid covering the full deposition space. Within each quadrant, the cross-sectional area A_q was computed using the Surveyor's area formula [50]:

$$A_q = \frac{1}{2}\left|\sum_{i=1}^{n}(x_i y_{i+1} - x_{i+1} y_i)\right|$$

where (x_i, y_i) are the coordinates of the i-th vertex of the contour in the quadrant, n is the number of vertices and the last vertex (x_{n+1}, y_{n+1}) is the same as the first vertex (x_1, y_1) to close the polygon.

By comparing corresponding quadrant areas between adjacent slices, this method enables the detection of even small indentations without relying on computationally expensive surface normal or curvature calculations.

A mathematical threshold τ for allowable cross-sectional variation can be defined based on the desired lateral surface smoothness:

$$\frac{|A_q^{j+1} - A_q^j|}{A_q^j} > \tau, \qquad \forall q, \tag{10}$$

where A_q^j and A_q^{j+1} are the cross-sectional areas of quadrant q in slices j and $j+1$, respectively. If any quadrant surpasses this threshold, the slice height is refined either by halving the number of passes per slice and inserting an additional slice or, in the case of single-pass slices, by increasing the scanning speed to maintain deposition uniformity.

4. Toolpath planning

Once the slice arrangement is determined, a filling path must be devised to construct the desired shape within each slice. In this study, the substrate is assumed to be flat, which is suitable for many CSAM applications. Therefore, the nozzle trajectory can be simplified by only controlling its position, while maintaining a constant inclination perpendicular to the initial substrate. A zigzag (parallel) path was selected to validate the proposed algorithm. This strategy is widely used due to its efficiency and versatility across various deposition scenarios. However, a known limitation of zigzag paths is the requirement for sharp turns at part boundaries, which can induce high jerk on the nozzle, leading to mechanical vibrations and build instabilities [21]. Despite these challenges, the zigzag pattern remains a robust choice due to its straightforward implementation and suitability for continuous material deposition.

Additionally, CSAM's continuous material flow constraint must be accounted for in path planning. Unlike other AM methods, CS does not permit deposition to be interrupted while traversing empty or concave sections within a slice. Therefore, over-spraying mitigation strategies – such as path reorientation – are necessary to ensure deposition accuracy and prevent material waste.

4.1. Toolpath generation and optimisation

Zigzag paths are generated starting from a set of parallel lines at a given distance, namely, the scanning step selected in section 3.1, and covering the full area of each slice. The key characteristics of the developed path-planning framework are as follows:

- *Path interpolation:* Interpolation methods, such as linear interpolation, were used to estimate the exact points where the scanning lines intersect with the slice contour, consisting of discrete data points.
- *Handling sharp turns:* To manage sharp turns at the edges and reduce nozzle jerk, smoother transition curves with precision zones can be implemented at the corners, promoting a more stable build process.
- *Path selection criteria:* The criteria for selecting the optimal scanning line inclination consist of minimising the total path length to reduce the number of nozzle turns while avoiding over sprays. The number of intersections with the slice contour will indicate whether a cross-section concavity is being crossed, as shown in Figure 5. If a line intersects the slice contour more than twice, the path is discarded, and a different inclination of scanning lines is tested, ensuring continuous deposition with minimal material waste. Among equivalent configurations, the path with the lower number of scanning lines was selected to minimise the number of required turns.
- *Transition between slices:* The transition between the end of one slice's path and the beginning of the next slice's path is managed by planning the path so that the final point of a slice trajectory seamlessly connects to the initial point of the subsequent slice.
- *Easy integration with offline programming software:* The path point coordinates, nozzle speeds, precision zones and nozzle rotation quaternions are automatically compiled to be easily imported into offline programming software, such as Robotstudio™.

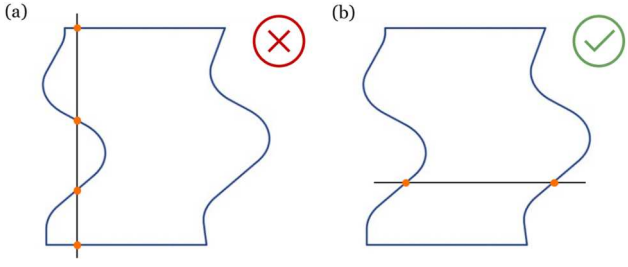


Figure 5. Representation of the intersections between slice contour and scanning lines: if more than two intersections occur (a), there will be overspray across unwanted areas; the path is optimal if only two intersections occur (b).

Table 7. CS nozzle parameter levels selected for the experimental campaign [22].

Variable parameter	No. of passes	Spray angle [°]	Scanning speed [mm/s]	SoD [mm]
No. of passes	10, 14, 20	90	250	30
Spray angle	20	70, 80, 90	250	30
Scanning speed	20	90	150, 250, 350	30
SoD	20	90	250	30, 35, 40

5. Experimental validation and demonstrators

The validity of the DNN predictions was compared against experimental acquisitions collected in a previous study of our group [22]. Each of the experiments was carried out by spraying 30 mm long tracks on a flat aluminium substrate with an Impact Innovations 5/11 CS system with OUT 1 nozzle. Samples' cross-sections were embedded in resin, ground with subsequent graded SiC papers and finally polished with diamond suspension up to 1 µm. The polished cross-sections were observed using Keyence VHX-5000 optical microscope, rendering a jpg format image. The experimental setup considered the same constant spray parameters listed in Tables 3 and 4, while the experiment matrix had a one-factor-at-a-time design, with levels listed in Table 7.

The captured images were converted into profile points through a Python script, applying a lowpass Gaussian filter to smooth out surface roughness, according to the method proposed in [51], as it is not the object of the prediction. The output was converted into grayscale and the profile points were extracted through thresholding.

For what concerns the slicing and tool path planning algorithms, their applicative potential was tested on a demonstrative CAD model. This demonstrator features concavities to test the concave-hull algorithm and verify the approach implemented for the identification of optimal scanning line orientation. Moreover, its variable cross-section validates the adaptive slicing algorithm. The devised geometry is depicted in Figure 6. The target flatness set for parameter optimisation was 98 %, while the tolerance on local cross-sectional

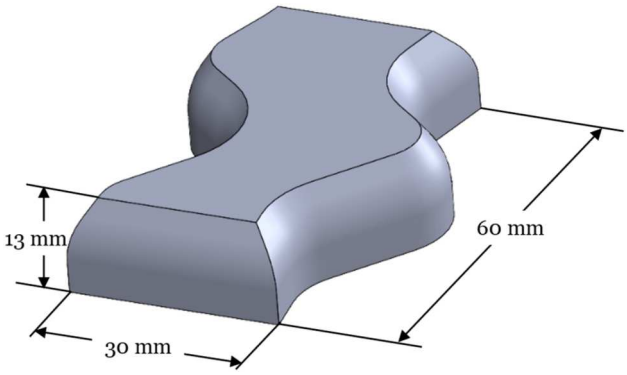


Figure 6. CAD model of the demonstrator devised for validation of the slicing and tool path planning algorithms.

variations for the slicing algorithm was set to 20 %. The present case was also used to make a realistic comparison on computational time between the analytical [22] and DNN models.

6. Results and discussion

The architecture of the shape prediction model was tested through a hyperparameter tuning procedure. The prediction results were then compared to the analytical model, and the experimental validations. The slicing and tool path planning algorithms were furthermore tested on a set of demonstrators, which were also used to evaluate the computational time reduction of the integrated framework. The following subsections describe the results obtained for validation.

6.1. Hyperparameter tuning

The DNN configuration was evaluated through the hyperparameter tuning procedure described in section 2.2. According to this analysis, the effect of the number of hidden layers on the prediction error for the testing dataset was identified and summarised in the boxplot of Figure 7. The mean error and variance for the configurations with only one hidden layer were significantly higher than those obtained for deeper networks, therefore, these configurations were excluded from the plot for better clarity.

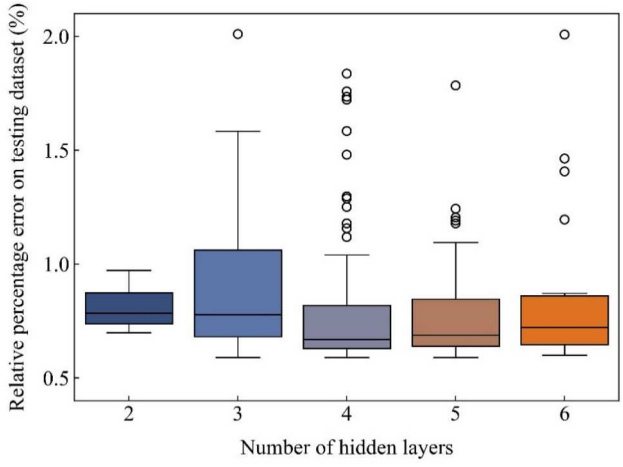


Figure 7. Boxplot showing the percent error of the trials of hyperparameter tuning, analyzing 2 to 6 hidden layers.

Table 8. Values of MSE and R^2 of the 2D DNN model with respect to the experimental results.

Variable parameter	Parameter value	MSE	R^2	MAPE (%)
N. of passes	10	$2.76 \cdot 10^{-3}$	0.976	9.78
	14	$4.97 \cdot 10^{-3}$	0.976	8.10
	20	$10.7 \cdot 10^{-3}$	0.974	9.58
Scanning speed [mm/s]	150	$41.1 \cdot 10^{-3}$	0.974	23.3
	250	$4.08 \cdot 10^{-3}$	0.991	9.58
	350	$1.74 \cdot 10^{-3}$	0.993	7.08
Spray angle [°]	90	$4.96 \cdot 10^{-3}$	0.990	8.78
	80	$5.09 \cdot 10^{-3}$	0.988	30.4
	70	$16.9 \cdot 10^{-3}$	0.954	35.3
SoD [mm]	30	$6.32 \cdot 10^{-3}$	0.986	9.92
	35	$5.61 \cdot 10^{-3}$	0.988	17.7
	40	$8.55 \cdot 10^{-3}$	0.984	23.5

The results show both the mean value of the errors and their variance decreases with increasing the number of hidden layers until reaching an optimum at four hidden layers. Deeper networks start showing lower performances, as they model a prediction complexity higher than the one required for the present problem. No specific trend was identified related to the mean number of neurons per layer or total neurons across all layers and performance. Conversely, the distribution pattern of the neurons along the layers had an identifiable effect. In particular, the top-performing configurations showed a recurring pattern of uniformly decreasing number of neurons along the depth of the network.

The configuration with the lowest error was selected as optimal. It presents four hidden layers, with 256, 128, 64 and 32 neurons. The number of training epochs was selected checking training and validation losses against overfitting, for a total training time of 16 hours for the 2D DNN and approximately 2 days for the 3D DNN on a single core of Intel® Gold Xeon® CPU 6248. The obtained relative L^2 percent error of the final configuration with respect to the testing dataset was 0.59 % for the 2D DNN and 2.62 % for the 3D one. These errors represent the ability of the DNNs to reproduce the analytical model's single-track predictions, while the following sections will analyze the experimental errors and the errors on more complex predictions.

6.2. Experimental validation

Experimental validation was performed using Mean Squared Error (MSE), the coefficient of determination

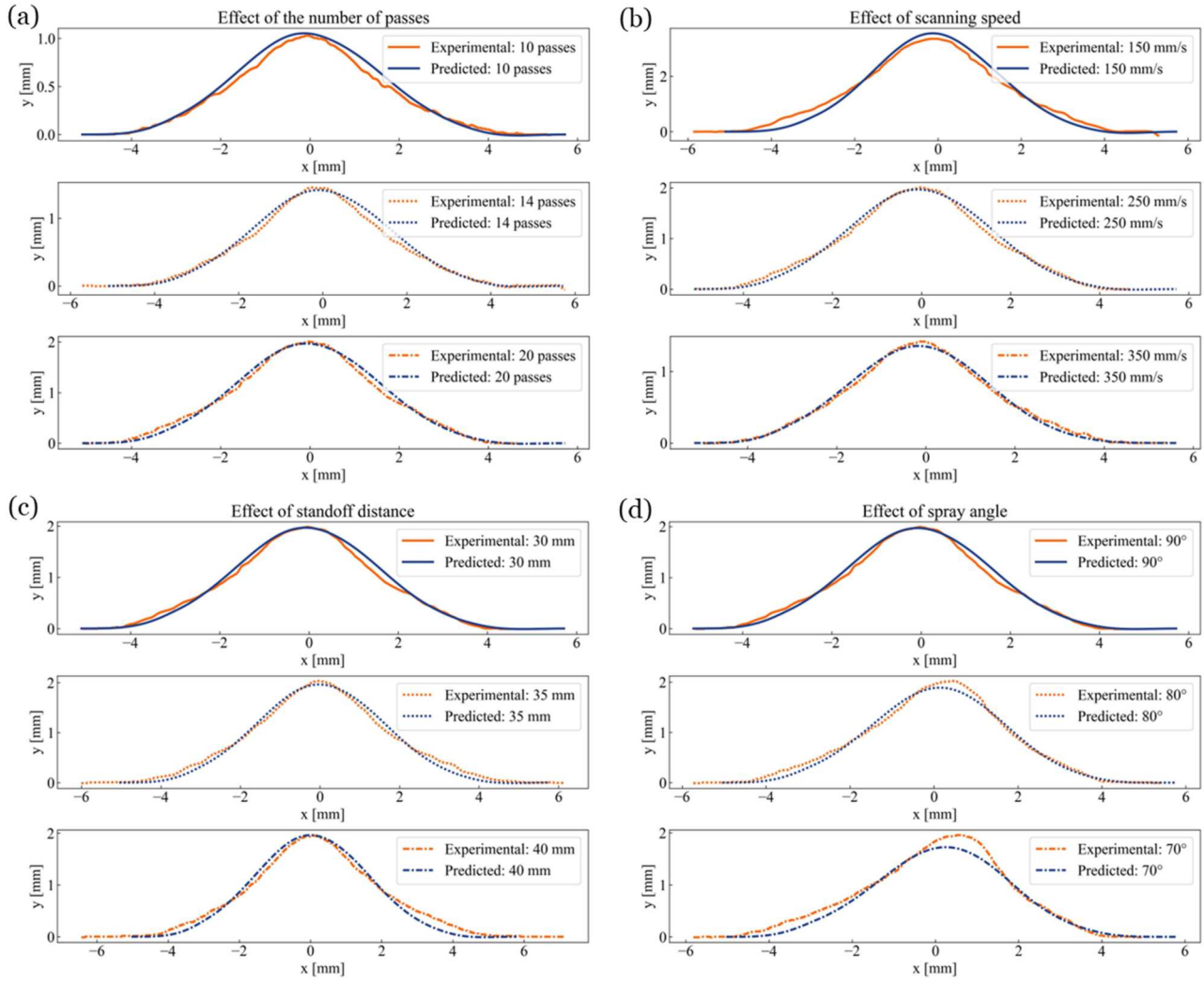


Figure 8. Comparative plot of the simulations and experimental results for different number of passes (a), scanning speed (b), spray angles (c) and SoD (d) with the 2D DNN simulation.

(R^2) and the mean absolute percentage error (MAPE) to highlight the model's susceptibility to variations in input parameters. MSE quantifies the prediction error, indicating how well the model fits the experimental data, with lower MSE corresponding to higher accuracy. In contrast, R^2 represents the proportion of variance explained by the model, where values closer to 1 correspond to lower variance. The MAPE was also included as an additional metric, for higher ease of interpretation: lower MAPE corresponds to a lower average error between experimental and predicted profiles. Concurrent evaluation of these metrics provides a more comprehensive assessment of the model's accuracy. The calculated values for all the tested cases are reported in Table 8 and comparative plots are shown in Figure 8.

The overall assessment of the model presents satisfactory results with both error metrics, averaging an MSE of $9.40 \cdot 10^{-3}$, an R^2 of 0.981 and a MAPE of 16%. These values are also competitive in comparison to other data-driven shape prediction models in the literature [23–26].

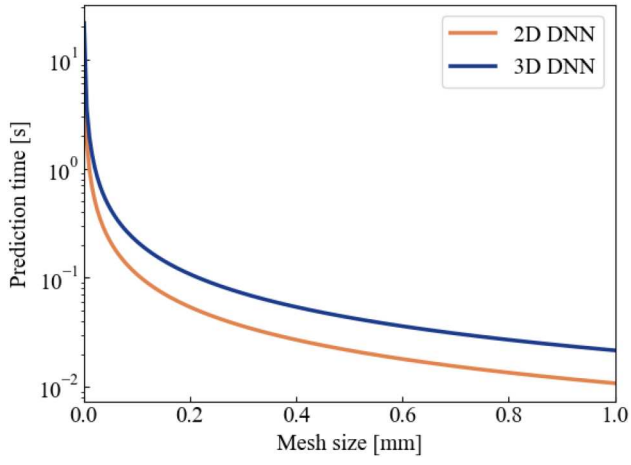


Figure 9. Plot illustrating the dependence of the 2D and 3D prediction runtime on the required grid mesh size.

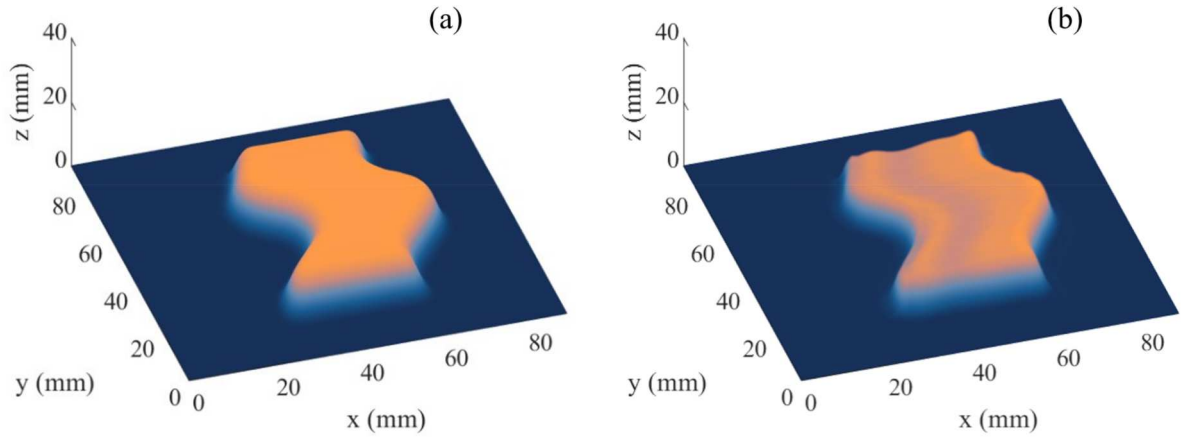


Figure 10. Comparison of final 3D shape predictions on the selected demonstrator: (a) with the analytical model, (b) with the DNN model.

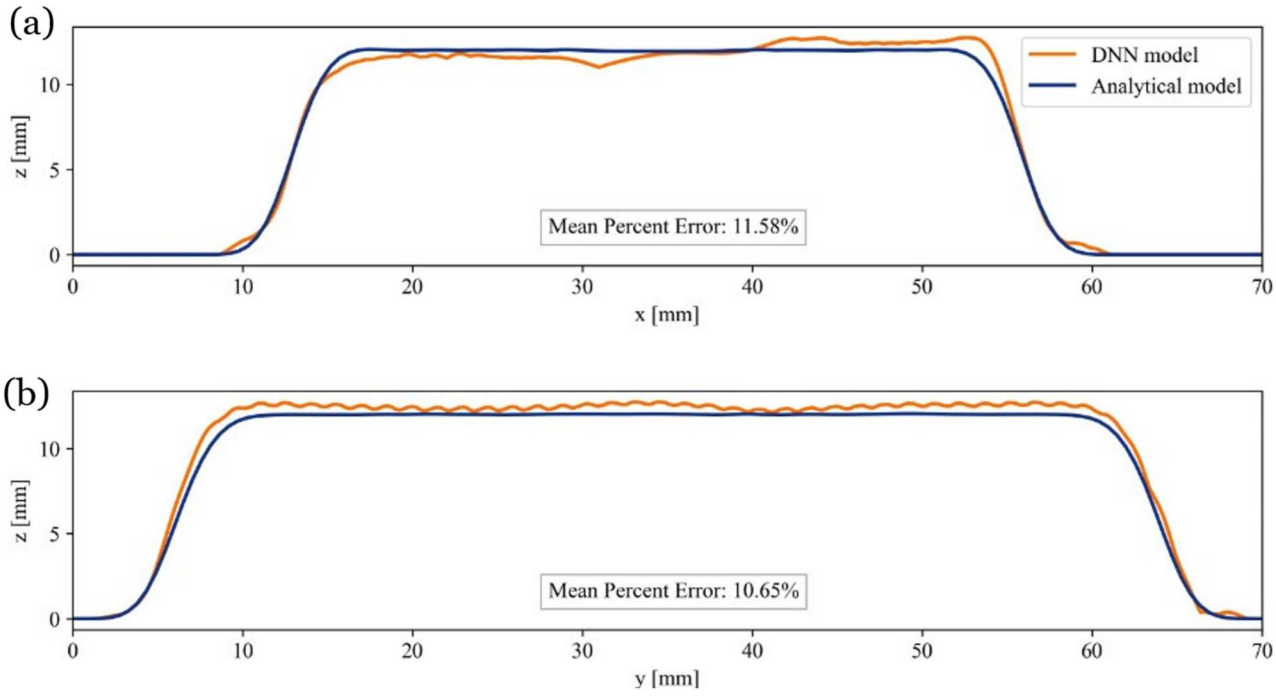


Figure 11. Comparison of the (a) *xz* and (b) *yz* side projected views of the demonstrator shape, predicted with the DNN and analytical models, with corresponding mean percent errors.

It must be highlighted how the extremities of the training domain present lower accuracy, due to the lower density of training data. Results regarding the prediction of different spray angles correctly predict the deposit volume but fail to fully capture the asymmetry of the profiles. This shortcoming mirrors the predictions of analytical models, highlighting a discrepancy between analytical and experimental profile skewness to be addressed in future works.

Table 9. Results of the parameter optimisation algorithm for the presented demonstrator (Figure 6).

Parameter	Value
Spray angle (°)	90
SoD (mm)	30
Max. passes per slice	8
Scanning speed (mm/s)	450
Flatness	99.5%
Excess height (mm)	0.45

6.3. Computational time evaluation

The evaluated runtime for the 2D and 3D DNN predictions was 1 ms/step and 2 ms/step respectively, which

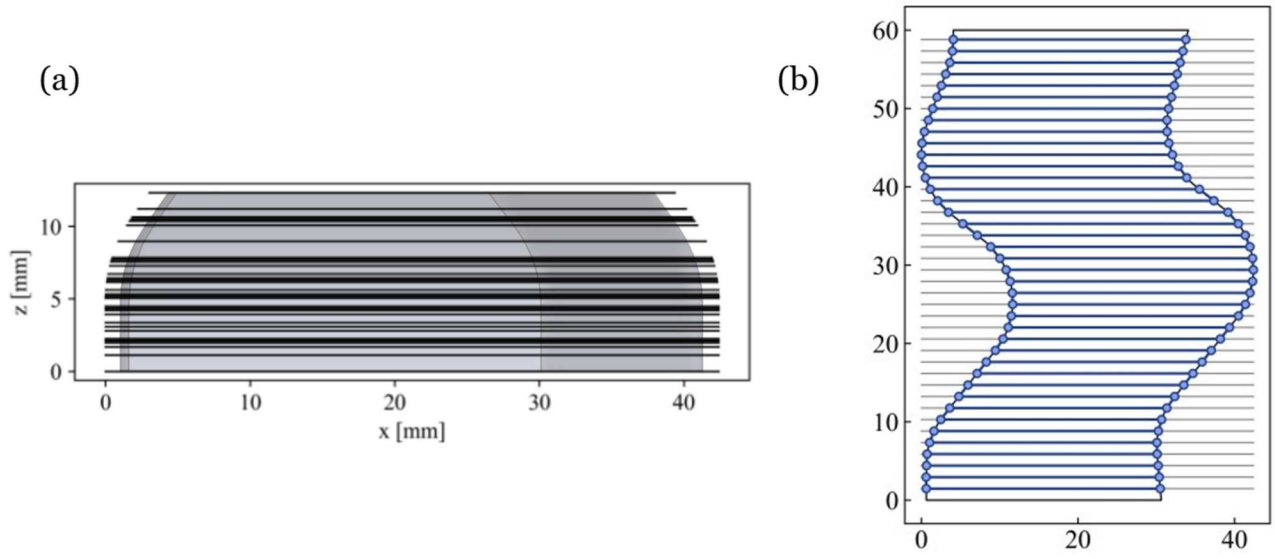


Figure 12. (a) Side view of the adaptively sliced demonstrator; (b) Tool path obtained on a single slice (blue) upon identified slice contour (black).

makes the total prediction time linearly dependent on the number of mesh points. At the same time, contrary to the analytical models, the duration is fully independent of process parameters such as the number of passes or the scanning speed. To exemplify the average prediction runtime for a single 10 mm wide and 30 mm long track, Figure 9 was produced, as a function of the required mesh size and under the assumption of a grid mesh.

Due to their different working principles, a straightforward comparison of the runtime of DNN model and the analytical model is not appropriate. The latter performs time-dependent predictions using Euler's method to solve the system of differential equations (Eq. 1, 2); therefore, both the time-step value and number are relevant parameters influencing computational time. In general, in the analytical model, the time-step magnitude is chosen to ensure the desired accuracy, while the number of time steps depends on the processing time and, therefore, on trajectory length and nozzle speed. The calculation speed may also be affected by the programming software, as the analytical model was originally developed in MATLAB, while the DNN one was based in Python. Bearing these considerations in mind, the demonstrator of Figure 7 was used to give an applicative indication of the comparative performance of the two approaches, conservatively selecting time steps and employing the same mesh size. The final shape predictions are shown in Figure 10. The results indicated that the prediction time of the analytical model was **2 h 16 min**, while that of the 3D DNN model was **32 s**, producing an overall reduction by a factor of **>250**.

For geometrical comparison, the side-views of the projected part on the *xz* and *yz* planes are represented in Figure 11 both for the analytical and DNN predictions. The mean percent error was calculated for both views, showing good correspondence between the models, with values of *11.58%* and *10.65%*, respectively. The analytical prediction shows smoother results, due to the averaging and interpolations it performs.

The overall results show that the DNN model significantly increases computational speed for predicting complex 3D shapes while preserving geometrical consistency.

6.4. Slicing and tool path planning evaluation

The slicing and tool path planning algorithms were tested on the demonstrator described in Section 5. The results of the parameter optimisation procedure are reported in Table 9, showing compliance with the set tolerances for surface waviness.

Figure 12(a) shows a representation of the adaptive slicing output, where thicknesses were refined alongside higher curvature edges, while Figure 12(b) exhibits the tool path corresponding to a single slice of the demonstrator geometry.

Additional demonstrators of the adaptive slicing algorithm are reported in Figure 13 to showcase its flexibility in handling different geometries.

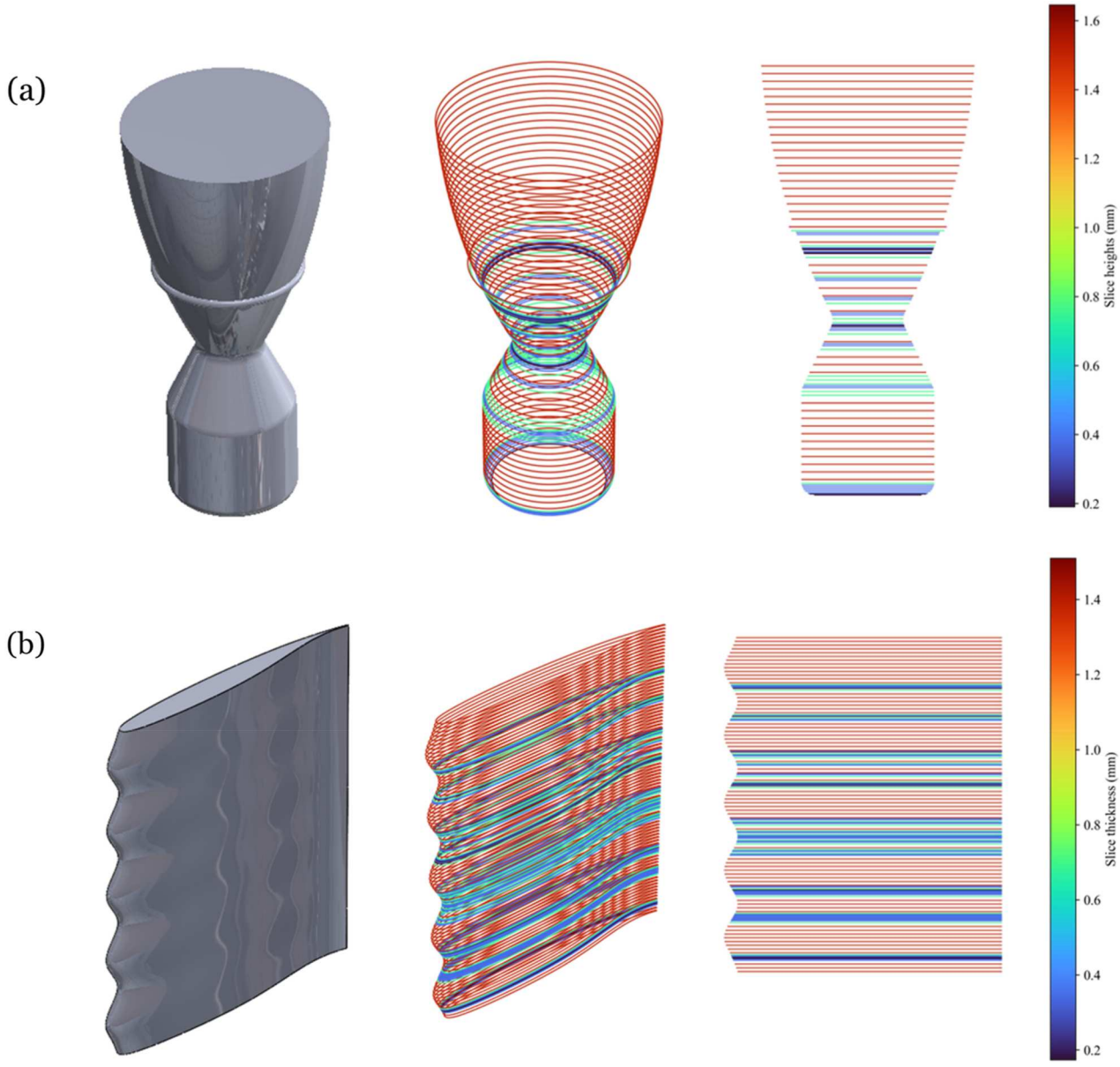


Figure 13. Additional examples of adaptively sliced models: (a) rocket nozzle, (b) leading edge.

7. Conclusions

This research successfully demonstrates the potential of combining DNNs with physics-based analytical data to enhance the predictive capabilities and efficiency of CSAM. The key findings include the development of two DNN models to predict the 2D cross-sectional and 3D shapes of cold spray deposits. These models, trained on a large dataset generated by a physics-based analytical model, exhibited robust and accurate predictions with relative percent errors of 0.59 % for the 2D model and 2.62 % for the 3D model with respect to the analytical predictions. A slightly higher error in the 3D predictions is expected, as the complexity of the 3D prediction is higher.

Experimental validation of the 2D DNN predictions showed high accuracy, with minimal MSE and high coefficients of determination (R^2). Although the models effectively captured the impact of varying process parameters, some limitations were observed for high spray angles, indicating the limitations of the analytical model.

The study highlights a significant improvement in computational efficiency, as the DNN models substantially reduced the prediction times compared to the baseline analytical model. This enhancement is crucial for real-time process optimisation and parameter adjustments, with the average prediction time for a single track of 10 mm width and 30 mm length, reduced to just 2 milliseconds per mesh point.

An adaptive slicing algorithm was developed to adjust the slice heights based on local curvature variations, reducing the staircase effect and improving geometric accuracy. Additionally, a tool path planning algorithm was implemented to ensure continuous deposition with minimal overspray, optimising the process for complex shapes. By integrating DNN predictions into an optimisation algorithm, this study achieved fine-tuning of process parameters for optimal geometric accuracy with minimal material waste. This approach proved effective in meeting target specifications for surface flatness and deposit uniformity.

The developed DNN algorithms have a high potential to be applied for producing complex, high-precision components with optimised material use and reduced production times. These advancements enable automated process adjustments, paving the way to enhance the quality and efficiency of CSAM.

Future research will focus on refining these models, enhancing their precision at the edges of the domain and introducing proper handling of curved substrates and shadow effects. The developed tool path generation procedure will therefore serve as a starting framework to suggest more flexible and complex trajectories, with varying filling patterns and spray angles. Additional improvements on computational time could be achieved through the integration with robot programming platforms. Building on these advancements, future efforts will also explore physics-informed AI models that embed domain-specific physical laws into the learning process for better generalisation and interpretability. The incorporation of continual learning frameworks will enable the models to adapt dynamically to evolving process conditions, further enhancing their scalability and utility in real-time optimisation for CSAM.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This work is supported by ERC-CO grant ArcHIDep, 101044228 Funded by the European Union. Views and opinions expressed are, however, those of the authors only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.

Data availability statement

The data that support the findings of this study are available from the corresponding author, S.B., upon reasonable request.

References

[1] Guan B, Qin L, Yang G, et al. Laser polishing of directed energy deposition metal parts: a review. Addit Manuf Front. 2024;3:200174. doi:10.1016/j.amf.2024.200174

[2] Şentürk E, Alparslan C, Bayraktar Ş, et al. A comprehensive review on sustainability in EDM process of additive manufactured materials. Measurement (Mahwah N J). 2025; 245:116626. doi:10.1016/j.measurement.2024.116626

[3] Ladani L, Sadeghilaridjani M. Review of powder bed fusion additive manufacturing for metals. Metals (Basel). 2021;11:1391. doi:10.3390/met11091391

[4] Mehta A, Vasudev H, Thakur L. Application of cold spray technology in design and manufacturing of complex geometries. Int J Interact Des Manuf. 2024;18:5407–5425. doi:10.1007/s12008-024-01826-x

[5] Wu H, Xie X, Liu M, et al. Stable layer-building strategy to enhance cold-spray-based additive manufacturing. Addit Manuf. 2020;35:101356. doi:10.1016/j.addma.2020.101356

[6] Irissou E, Legoux J-G, Ryabinin AN, et al. Review on cold spray process and technology: Part I—intellectual property. J Therm Spray Tech. 2008;17:495–516. doi:10.1007/s11666-008-9203-3

[7] Thurston SH. Method of impacting one metal upon another. US706701A; 1902.

[8] Rocheville CF. Device for treating the surface of a workpiece. US3100724A; 1963.

[9] Papyrin A, Kosarev V, Klinkov S, et al. *Cold spray technology*. Elsevier; 2006. doi:10.1016/B978-008045155-8/50001-6

[10] Villafuerte J, editor. *Modern cold spray: materials, process, and applications*. Cham: Springer International Publishing; 2015. doi:10.1007/978-3-319-16772-5

[11] Yin S, Fan N, Huang C, et al. Towards high-strength cold spray additive manufactured metals: methods, mechanisms, and properties. J Mater Sci Technol. 2024;170:47–64. doi:10.1016/j.jmst.2023.05.047

[12] Kafle A, Silwal R, Koirala B, et al. Advancements in cold spray additive manufacturing: process, materials, optimization, applications, and challenges. Materials. 2024;17:5431. doi:10.3390/ma17225431

[13] Majeed A, Ahmed A, Lv J, et al. A state-of-the-art review on energy consumption and quality characteristics in metal additive manufacturing processes. J Braz Soc Mech Sci Eng. 2020;42:249. doi:10.1007/s40430-020-02323-4

[14] Toyserkani E, Sarker D, Ibhadode OO, et al. *Metal additive manufacturing*. 1st ed. Wiley; 2021.

[15] Ahn D-G. Directed energy deposition (DED) process: state of the art. Int J Precis Eng Manuf-Green Tech. 2021;8:703–742. doi:10.1007/s40684-020-00302-7

[16] Pattison J, Celotto S, Morgan R, et al. W. Cold gas dynamic manufacturing: a non-thermal approach to freeform fabrication. Int J Mach Tools Manuf. 2007;47:627–634. doi:10.1016/j.ijmachtools.2006.05.001

[17] Ompong D, Murray R, Thorbjornsen ZS, et al. Environmental impact of cold sprayed 3D-Printed aluminium metal parts. J Cleaner Prod. 2022;380:135096. doi:10.1016/j.jclepro.2022.135096

[18] Pattison J, Celotto S, Khan A, et al. Standoff distance and bow shock phenomena in the Cold Spray process. Surf

Coat Technol. 2008;202:1443–1454. doi:10.1016/j.surfcoat.2007.06.065
[19] Binder K, Gottschalk J, Kollenda M, et al. Influence of impact angle and gas temperature on mechanical properties of titanium cold spray deposits. J Therm Spray Tech. 2011;20:234–242. doi:10.1007/s11666-010-9557-1
[20] Klinkov SV, Kosarev VF, Ryashin NS, et al. Influence of particle impact angle on formation of profile of single coating track during cold spraying. AIP Conference Proceedings. 2018;2027:020007. doi:10.1063/1.5065085
[21] Wu H, Liu S, Li W, et al. Generic implementation of path design for spray deposition: programming schemes, processing and characterization for cold spraying. Surf Coat Technol. 2023;458:129368. doi:10.1016/j.surfcoat.2023.129368
[22] Vanerio D, Kondas J, Guagliano M, et al. 3D modelling of the deposit profile in cold spray additive manufacturing. J Manuf Process. 2021;67:521–534. doi:10.1016/j.jmapro.2021.05.013
[23] Ikeuchi D, Vargas-Uscategui A, Wu X, et al. Neural network modelling of track profile in cold spray additive manufacturing. Materials (Basel). 2019;12:2827. doi:10.3390/ma12172827
[24] Ikeuchi D, Vargas-Uscategui A, Wu X, et al. Data-efficient neural network for track profile modelling in cold spray additive manufacturing. Appl Sci. 2021;11:1654. doi:10.3390/app11041654
[25] Ikeuchi D, Vargas-Uscategui A, Wu X, et al. Data-driven overlapping-track profile modeling in cold spray additive manufacturing. J Therm Spray Tech. 2024;33:530–539. doi:10.1007/s11666-024-01733-3
[26] Liu M, Wu H, Yu Z, et al. Description and prediction of multi-layer profile in cold spray using artificial neural networks. J Therm Spray Tech. 2021;30:1453–1463. doi:10.1007/s11666-021-01212-z
[27] Savangouder RV, Patra JC, Palanisamy S. A Machine learning technique for prediction of cold spray additive manufacturing input process parameters to achieve a desired spray deposit profile. IEEE Trans Ind Inf. 2024;20:12275–12283. doi:10.1109/TII.2024.3417300
[28] Lewke M, Wu H, List A, et al. Automated path and trajectory planning for automated repair of damaged components by cold spray. ASM International. 2023: 436–442. doi:10.31399/asm.cp.itsc2023p0436
[29] Li W, Yao Y, Wu H, et al. Spray trajectory planning for complex structural components in robotized cold spray additive manufacturing. J Therm Spray Tech. 2024;33:71–87. doi:10.1007/s11666-023-01696-x
[30] Nault IM, Ferguson GD, Nardi AT. Multi-axis tool path optimization and deposition modeling for cold spray additive manufacturing. Additive Manufacturing. 2021;38:101779. doi:10.1016/j.addma.2020.101779
[31] Matthews J, Klatt T, Morris C, et al. Hierarchical design of negative stiffness metamaterials using a Bayesian network classifier1. J Mech Des. 2016;138(4):041404. doi:10.1115/1.4032774
[32] Li Z, Zhang Z, Shi J, et al. Prediction of surface roughness in extrusion-based additive manufacturing with machine learning. Robot Comput Integr Manuf. 2019;57:488–495. doi:10.1016/j.rcim.2019.01.004
[33] Mahapatra SS, Sood AK. Bayesian regularization-based Levenberg–Marquardt neural model combined with BFOA for improving surface finish of FDM processed part. Int J Adv Manuf Technol. 2012;60:1223–1235. doi:10.1007/s00170-011-3675-x
[34] Xiong J, Zhang G, Hu J, et al. Bead geometry prediction for robotic GMAW-based rapid manufacturing through a neural network and a second-order regression analysis. J Intell Manuf. 2014;25:157–163. doi:10.1007/s10845-012-0682-1
[35] Chowdhury S, Anand S. *Artificial neural network based geometric compensation for thermal deformation in additive manufacturing processes*. Blacksburg, Virginia: American Society of Mechanical Engineers Digital Collection; 2016; doi:10.1115/MSEC2016-8784
[36] Klinkov SV, Kosarev VF, Shikalov VS. Control of cold spray process by changing of nozzle setting angle. AIP Conf Proc. 2019;2125:020022. doi:10.1063/1.5117382
[37] Ardeshiri Lordejani A, Romanenghi L, Pollastri A, et al. Deposit shape control for local repair and welding by cold spray. J Manuf Process. 2024;112:45–59. doi:10.1016/j.jmapro.2024.01.023
[38] Akiba T, Sano S, Yanase T, et al. Optuna: a next-generation hyperparameter optimization framework. In: *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. New York, NY: Association for Computing Machinery; 2019. p. 2623–2631. doi:10.1145/3292500.3330701
[39] Abadi M, Agarwal A, Barham P, et al. TensorFlow: large-scale machine learning on heterogeneous distributed systems; 2016. doi:10.48550/arXiv.1603.04467
[40] Liu DC, Nocedal J. On the limited memory BFGS method for large scale optimization. Math Program. 1989;45:503–528. doi:10.1007/BF01589116
[41] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. In: *Advances in neural information processing systems*, vol. 30. Long Beach, California: Curran Associates, Inc.; 2017.
[42] Shi X, Chen Z, Wang H, et al. Convolutional LSTM network: a machine learning approach for precipitation nowcasting. In: *Advances in neural information processing systems*, vol. 28. Montreal Canada: Curran Associates, Inc.; 2015. p. 802–810.
[43] Lecun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition. Proc IEEE. 1998;86:2278–2324. doi:10.1109/5.726791
[44] Chen T, Guestrin C. XGBoost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. 2016: 785–794. doi:10.1145/2939672.2939785
[45] Schuster M, Paliwal KK. Bidirectional recurrent neural networks. IEEE Trans Signal Process. 1997;45:2673–2681. doi:10.1109/78.650093
[46] Du X, Shen Y, Zhao W, et al. Wire arc additive manufacturing from the perspective of remanufacturing: a review of data processing. J Manuf Process. 2023;107:385–410. doi:10.1016/j.jmapro.2023.10.019
[47] Shishkovsky I. *New trends in 3D printing*. IntechOpen; 2016. doi:10.5772/61398

[48] Asaeedi S, Didehvar F, Mohades A. α-Concave hull, a generalization of convex hull. Theor Comput Sci. 2017;702:48–59. doi:10.1016/j.tcs.2017.08.014

[49] Jamieson R, Hacker H. Direct slicing of CAD models for rapid prototyping. Rapid Prototyp J. 1995;1:4–12. doi:10.1108/13552549510086826

[50] Braden B. The surveyor's area formula. College Math J. 1986;17:326–337. doi:10.1080/07468342.1986.11972974

[51] Venturi F, Gilfillan N, Hussain T. A convolution-based approach to cold spray additive manufacturing. Addit Manuf Lett. 2021;1:100014. doi:10.1016/j.addlet.2021.100014