



eXplainable artificial intelligence for automatic defect detection in additively manufactured parts using CT scan analysis

Harsh Bordekar^{1,3} · Nicola Cersullo¹ · Marco Brysch¹ · Jens Philipp² · Christian Hühne^{1,3}

Received: 24 April 2023 / Accepted: 4 November 2023
© The Author(s) 2023

Abstract

Additive Manufacturing (AM) and in particular has gained significant attention due to its capability to produce complex geometries using various materials, resulting in cost and mass reduction per part. However, metal AM parts often contain internal defects inherent to the manufacturing process. Non-Destructive Testing (NDT), particularly Computed Tomography (CT), is commonly employed for defect analysis. Today adopted standard inspection techniques are costly and time-consuming, therefore an automatic approach is needed. This paper presents a novel eXplainable Artificial Intelligence (XAI) methodology for defect detection and characterization. To classify pixel data from CT images as pores or inclusions, the proposed method utilizes Support Vector Machine (SVM), a supervised machine learning algorithm, trained with an Area Under the Curve (AUC) of 0.94. Density-Based Spatial Clustering with the Application of Noise (DBSCAN) is subsequently applied to cluster the identified pixels into separate defects, and finally, a convex hull is employed to characterize the identified clusters based on their size and shape. The effectiveness of the methodology is evaluated on Ti6Al4V specimens, comparing the results obtained from manual inspection and the ML-based approach with the guidance of a domain expert. This work establishes a foundation for automated defect detection, highlighting the crucial role of XAI in ensuring trust in NDT, thereby offering new possibilities for the evaluation of AM components.

Keywords Additive Manufacturing (AM) · Laser-Powder Bed Fusion (L-PBF) · Non Destructive Testing (NDT) · Computed Tomography (CT) · Machine Learning (ML) · EXplainable Artificial Intelligence (XAI) · Internal defects

Introduction

Additive Manufacturing (AM) techniques have attracted a lot of attention in the last years due to the possibility to create and produce complex geometries through an optimized design approach. The possible mass and cost-saving have made this technology highly competitive compared to conventional manufacturing processes. In particular, Laser-Powder Bed Fusion (L-PBF) with its rapid progress started to open a potential market with new applications and parts

in different fields such as aerospace and automotive (Gibson et al., 2021). Although the increased maturity of this technology, the production process needs to be properly mastered to avoid process-induced characteristics. Indeed, despite extensive research and technological advancement, internal defects may always occur (Yap et al., 2015). These might be attributed to different causes: non-uniform powder, slight changes in the process parameters with an effect on the laser beam, scanning and building strategy, and deformations during the manufacturing process are just a few examples (Kasperovich et al., 2016; Yadollahi & Shamsaei, 2017). These factors may lead to a wide range of potential defects. Among them, the main may be classified according to their nature in inclusions, pores, and lack of fusions (LoF). Inclusions are caused by foreign material while pores and LoF are due to the process-related gas bubbles entrapped during the solidification process of the melt pool and the low melting power beam, respectively. The different defect types may affect the overall part quality due to their impact on the

✉ Harsh Bordekar
h.bordekar@tu-braunschweig.de

¹ Institute of Mechanics and Adaptronics, Technische Universitaet Braunschweig, Braunschweig, Germany

² Institute of Joining and Welding, Technische Universitaet Braunschweig, Braunschweig, Germany

³ Institute of Composite Structures and Adaptive Systems, German Aerospace Center (DLR), Braunschweig, Germany

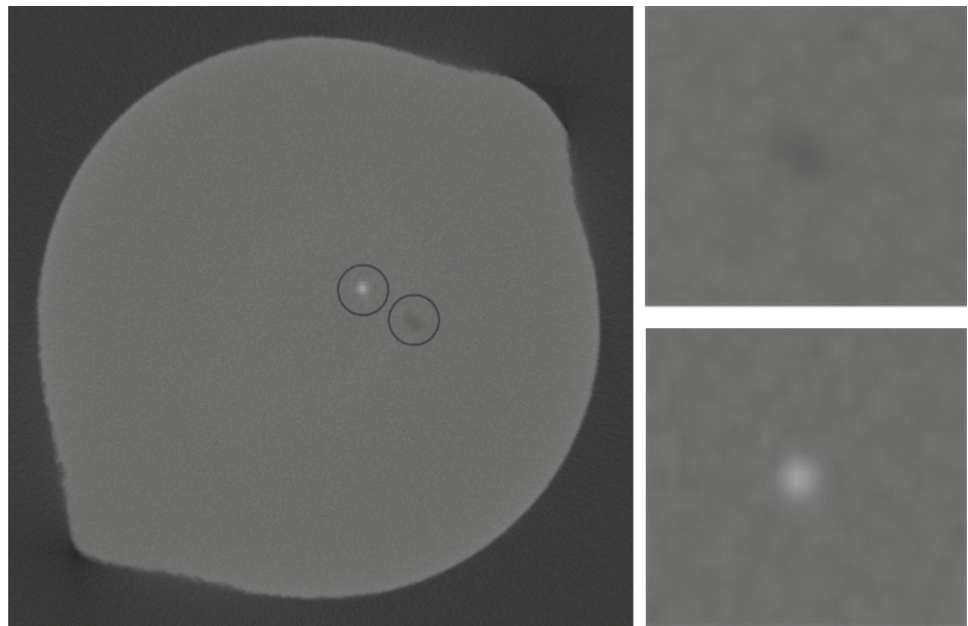
mechanical properties (Greitemeier et al., 2017; Cersullo et al., 2022). Today industries mainly define part acceptance criteria according to their size, shape, or even distribution pattern. In this framework, Non Destructive Testing (NDT) is an essential step in the overall process chain to identify and characterize the different defects. Among the various NDT techniques (Koester et al., 2019), Computed Tomography (CT) is commonly used for structural integrity investigations (De Chiffre et al., 2014) and part certification (du Plessis et al., 2018). The ability to inspect complex geometries and to provide a 3D volumetric analysis of internal defects makes CT more suitable for AM analysis compared to 2D X-ray or ultrasonic techniques. It is worth mentioning that the use of CT has been widely adopted in literature to improve the understanding of the effect of specific defects on the mechanical properties (du Plessis et al., 2016; Gong et al., 2019) and to identify inspection and qualification criteria for AM components (Seifi et al., 2017, 2016).

In a CT scan, X-ray measurements of an object at different rotating angles are taken to record the projection data. Such projections are later used to construct a cross-sectional image at multiple height levels using image reconstruction algorithms (Seeram & Sil, 2013). The fully reconstructed volumetric data can be stored as voxels, where the brightness level of each voxel corresponds to the local penetration depth of X-rays (or the material attenuation). A higher penetration ability represents a darker spot in the corresponding cross-sectional image or volumetric data set. Final data can be viewed as a stack of 2D images in any plane. When this sort of conversion from a 3D model to 2D image stacks happens, voxels are converted into pixels that maintain the same characteristics in terms of value intensity and position. In particular, the different brightness levels of the features in the layer-wise gray-scale images can be used as a key indicator to detect the different defects. As shown in Fig. 1, an inclusion appears as a bright spot and similarly, pores are visible as dark spots in a sufficiently high contrast compared to the surrounding bulk material. It is worth mentioning that LoF and pores are similar in terms of feature characteristics due to their nature and for this reason from now on the term pore will be used indistinctly to address both of them.

The CT scan analysis process still represents today a time-consuming activity from both an inspection process and an evaluation point of view. In particular, the latter usually requires a manual defects detection process through the various scanned layers. This activity is usually performed by technical experts in the domain who use dedicated software to tune several visualization parameters for optimal feature interpretation and recognition. Indeed, this brings an inevitable user dependency data interpretation that may cause a non-full match between data sets evaluated by different operators. In this optic, an automatic defect detection tool might not only eliminate human interactions but also

reduce the inspection time obtaining a fully automatic control process with a consequent improvement in the overall process chain and a reduction in the cost per part. Commercial NDT analysis software provides an alternative to manual inspection through different image processing and segmentation tools. In most of the adopted methodologies, semantic segmentation is done based on the brightness of the voxels to select an appropriate threshold that is used to binarize the different images into the main foreground (usually represented by pores) and background (Gong et al., 2019). In this way, voxels belonging to the different pores can be identified. The threshold itself is customarily obtained from well-known methods (Otsu, 1979) but it may vary locally for different layers. The definition of an optimal threshold is made by the user within the evaluation software, which makes the result of the analysis dependent on individual decisions and thus reproducibility is limited. Moreover, it has to be pointed out that CT scan artifacts such as beam hardening, scatter, Poisson noise, and ringing effect may slightly change pixel intensities (Boas & Fleischmann, 2012), affecting the efficiency of the thresholding. To overcome some of these limitations and to obtain a reliable analysis that is as far as possible independent of individual decisions, the use of Machine Learning (ML) algorithms for objects detection and automatic classification is constantly increasing (Spierings et al., 2011; Erhan et al., 2014; Sadoon, 2021). In particular, to identify internal defects numerous researchers followed different approaches. Caggiano et al. (2019) proposed an ML-based online SLM process monitoring routine to detect defects in an AM part. Schlotterbeck et al. (2020) adopted a deep Artificial Neural Network (ANN) trained on human-annotated defects to analyze 2D images of specimens at locations identified from a Support Vector Machine (SVM) model. The ANN used by the authors was trained on 30.000 samples and it was labeled for different types of defects by experts. Mutiargo et al. (2019) used instead an ANN known as U-NET to detect porosity in AM-built samples; the pre-segmentation of the training images was performed using the open-source software Fiji (Schindelin et al., 2012). Those images were then controlled and re-labeled by a human operator for the final decision. The results from the ANN were post-processed to determine the total porosity in the specimen, and compared with results from Archimedes tests. Fuchs et al. (2019) used the U-NET architecture to detect defects in aluminum casting parts; a total of 675 simulated CT scans with defects like pores and shrinkage were used for training purposes. Furthermore, three real CT scan data sets hand-labeled by eight experts were used to check as the ground truth label. Gobert et al. (2020) presented a methodology for automatic porosity segmentation of X-ray images based on convolutional neural networks in combination with OTSU thresholding. Shipway et al. (2021) trained a CNN architecture known as ResNet to perform automated defect detection in fluores-

Fig. 1 A CT scan layer with a pore (dark spot) and an inclusion (bright spot)



cent penetration inspection. A visible spectrum camera was used in Bacioiu et al. (2019) to capture 30008 images of multiple welding runs. Such data was later used for training an ML model for automatic defect classification. Du Du et al. (2019) used a deep learning approach to improve the X-ray image defect detection of casting specimens. Similarly, Torbati-Sarraf et al. (2021) used four different states of art CNN models for semantic segmentation of a 3D Transmission x-ray Microscopy (TXM) nano-tomography image data. Recently, Wong et al. (2021) performed a 3D volumetric segmentation using a 3D U-NET CNN architecture. In their work, three variants of 3D U-NETs were trained and tested on an AM specimen. The 3D U-NET achieved a mean Intersection Of Union (IOU) score of 88.4%.

As observed by all the authors, the role of an expert is fundamental for building up a good training data set. Moreover the choice of Convolutional Neural Networks (CNNs) comes with certain challenges. Firstly, CNNs often demand substantial amounts of data for effective training (Singh et al., 2020). Secondly, these models are complex in nature, making them challenging to explain or interpret (Linardatos et al., 2020). In contrast, Support Vector Machines (SVMs) and Random Forests (RFs) are more data-efficient, requiring comparatively smaller datasets to perform well (Vargas-Lopez et al., 2021). Moreover, SVMs and RFs are known for their transparency and interpretability. They provide clear decision boundaries and feature importances, making it easier for humans to comprehend and trust their predictions.

Furthermore, the methods discussed are considerably effective in the reckoning of the overall porosity level using layer-wise 2D analysis. However, the 3D shape and size characterization with the actual location identification of

every single defect requires additional effort. Moreover, all of them suffer from low explainability and interpretability as they tend towards deep learning techniques. In the context of machine learning systems, interpretability refers to the capacity to explain or present information in a way that is understandable to humans (Samek & Müller, 2019). It should provide insights into the decision-making process, revealing the relationship between the input variables and the corresponding outputs (Lundberg et al., 2020). On the other hand, interpretability goes beyond solely obtaining accurate predictions and instead focuses on comprehending the underlying factors and mechanisms employed by the model. Under the scope of NDT, interpretability plays a paramount role to develop trust in the model and the overall methodology. Indeed, by adopting such methodologies, understanding the logic and the reasoning behind a particular decision becomes a challenging activity. Several recent studies have actually emphasized the need for an explainable and interpretable AI (sub)systems model (Linardatos et al., 2020; Burkart & Huber, 2021; Bussmann et al., 2020). In this framework, the European Union Aviation Safety Agency (EASA) published a guideline for level-1 ML applications (Torens et al., 2022). It is important to additionally underline that all state of art deep learning techniques mentioned focus their attention mainly on pores, not taking into account the detection of inclusions. Moreover, the bottleneck of training data requirement that comes with such methods represents an additional constraint for their application.

To overcome the shortcomings of the state-of-the-art methods, an eXplainable Artificial Intelligence (XAI) approach for automatic detection and characterization was employed in the present work. Moreover, the presented method also

Table 1 CT scan setup parameters

X-ray beam parameter	
U_B [kV]	110
I_R [μ A]	110
P [W]	12.1
Detector settings	
Timing [ms]	600
Average	3
Skip	2
Acquisition	
Images	1441
Duration [min]	216:15 (multiscan)
Voxel [μ m]	12.5

provides additional location and morphological information regarding individual defects.

Materials and methods

Data set

In the present work, three cylindrical Ti6Al4V specimens were manufactured by L-PBF in a single build. Coupons were subjected to a stress relief at 720°C after the printing process, followed by an annealing procedure at 920°C for 2 hours. Samples diameter and height were 11 mm and 90 mm, respectively. CT scan measurements were carried out on the entire specimens through a phoenix “v-tome-x s 240” μ CT system by GE Sensing & Inspection Technologies GmbH (Wunstorf, Germany) with a 240 kV and 320 W microfocus tube enabling a theoretical detail detectability $< 1 \mu\text{m}$. A tungsten target is used in this type of direct X-ray tube for the generation of X-radiation with a usable beam cone of approx. 25°. The 3D tomography system was operating via “xs control” and “phoenix Datos-x acquisition 2.0” software (both GE Sensing & Inspection Technologies GmbH). Reconstruction of the CT scan was performed using phoenix “Datos-x reconstruction” software [39]. An overall resolution of 12.5 μm is achieved for all the analyzed samples. Further information regarding the CT scan parameters adopted are shown in Table 1.

Two of the printed specimens were analyzed by a technical expert who manually identify and labeled both defects and inclusions. The methodology adopted by the expert is not part of the analysis and it is therefore not reported in the context of the present work. Nevertheless, for the proposed activity the results provided were considered acceptable in terms of defects detection. The labeled data were used for training (80%) and testing (20%) purposes. On the third

printed coupon, a complete 3D defects characterization was performed. No evaluation of this latter sample was done a priori. The final results from the selected model were then validated by the same expert.

Machine learning framework for defect detection

The methodology proposed in this work may be divided into two main steps (a schematic overview is provided in Fig. 2):

- Pixe-wise defect detection;
- Defect clustering and analysis.

CT scans were performed on the additively manufactured specimens using appropriate scanning parameters to capture internal details and defects. The CT scan data comprised a series of layered images. A trained machine learning model was employed to detect and classify pixels belonging to pores and inclusions within the CT scan images. The model was developed using layered images as input sequentially, and it classified pixels based on their characteristics and features. The pixels identified as pores and inclusions were extracted from the CT scan images and used to generate a 3D point cloud representation. This point cloud consisted of points corresponding to the detected defects within the specimen.

A sophisticated spatial clustering technique, specifically DBSCAN, was applied to the 3D point cloud to identify clusters of pixels that were in close proximity to each other. The clustering process effectively grouped pixels belonging to the same defect, aiding in the localization and characterization of defects within the specimen. The identified defect clusters were further analyzed to extract valuable information about each defect. Parameters such as projected area, projected length, and location of the defects were computed and recorded. These parameters provided insights into the size, shape, and distribution of the detected defects within the additively manufactured specimen.

Overall, the methodology involved a sequential process of CT scanning, machine learning-based pixel detection, 3D point cloud generation, DBSCAN clustering, and subsequent defect analysis to extract key defect characteristics. This approach enabled the automated detection and characterization of internal defects in the additively manufactured specimens, facilitating quality control and further analysis of the AM components.

Defects detection using supervised ML models

Traditional ML models require a feature extraction pipeline to aid the learning process (Douglass, 2020). The training features utilized in the model consisted of the pixel intensities

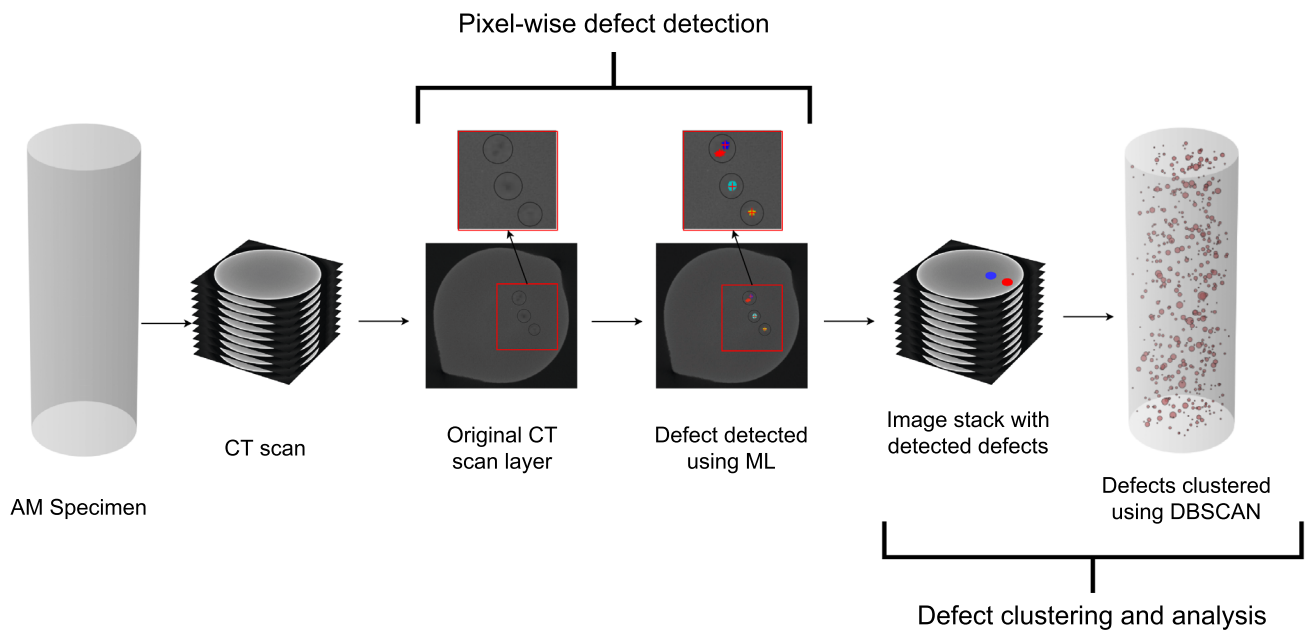


Fig. 2 A flow diagram describing the proposed methodology

from the original gray-scale image, along with the intensities obtained after applying a Gaussian filter with a kernel size of 3. This combined approach of considering both the actual pixel intensities and the Gaussian filtered intensities aids the model in recognizing and classifying pixels based not only on their gray-scale intensity but also on the abrupt changes in the surrounding gray-scale intensities. A data repository comprised of hand-labeled images was assembled with the help of an expert. All images containing key features were cropped at a specific location from the labeled XY layer to create the dataset. A few examples from the dataset adopted are shown in Fig. 3. In particular, Fig. 3(a) shows the original patches and Fig. 3(b) the ground truth labels. In the latter, the black pixels represent the bulk of the material while the white and the blue pixels represent the pore and the inclusion, respectively.

Re-sampling of imbalance data set

In the realm of data mining and machine learning applications, it is uncommon to encounter datasets that are perfectly balanced in their distribution. In a multi-class classification, class imbalance occurs when one or more classes, representing the majority of the data instances, affect the capability of the ML model to perform accurately. In an imbalanced dataset, the majority class is called negative class and the minority class is called positive class. ML model trained on imbalance dataset are prone to be biased towards negative class. Moreover, statistical evaluation techniques like Con-

fusion Matrix and F-1 Score do not provide a valuable result for a skewed test dataset.

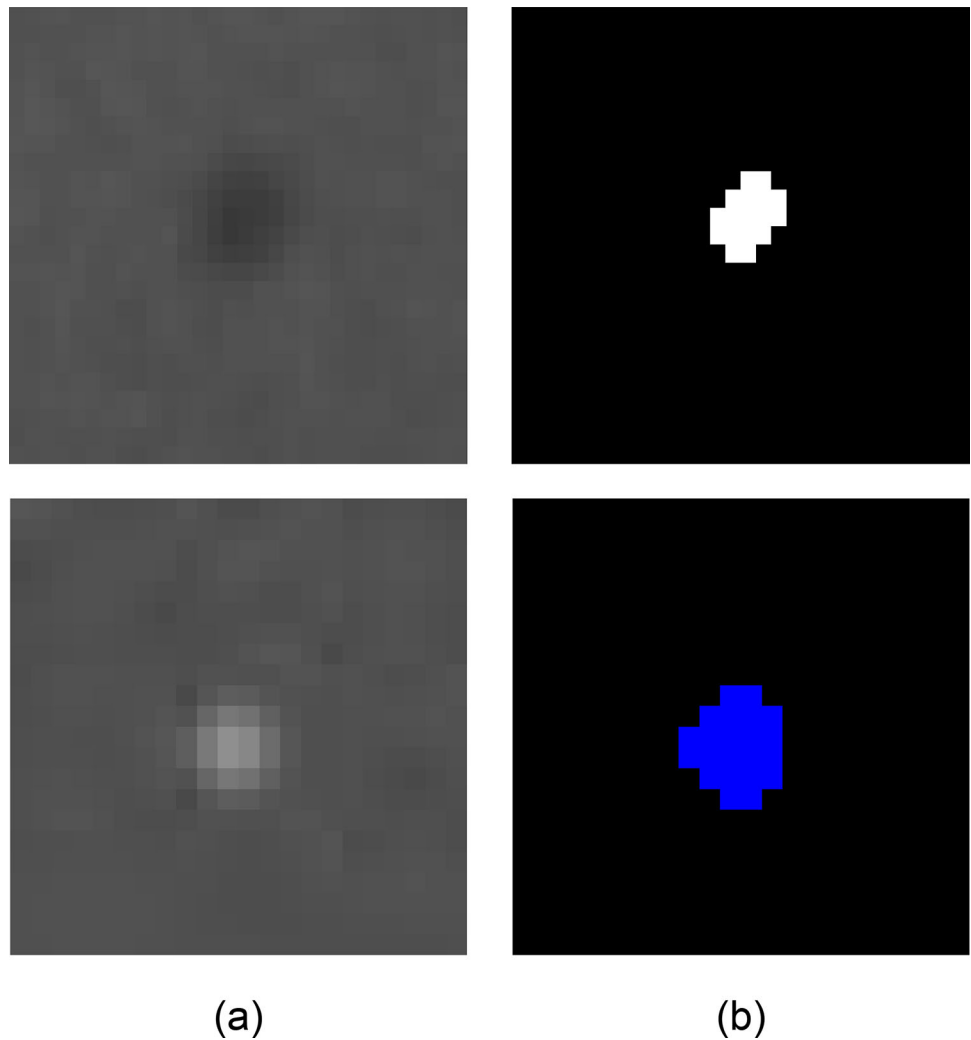
In the presented work the the skewness of the class distribution can be measured with the help of imbalance ratio (IR), which can be measured for each class using the Eqn. 1.

$$IR_{\text{specific class}} = \frac{\text{majority class}}{\text{Specific minority class}} \quad (1)$$

The IR for pore and inclusion were 2.567 and 14.943 respectively. Therefore it was necessary to tackle this imbalance in the dataset, before employing it for training and testing.

Depending on the data type, several techniques were implemented to tackle the class imbalance problem (Zhu et al., 2019; Du et al., 2019; Shipway et al., 2021). Among them, the data level technique is the most intuitive way to treat a skewed dataset due to the fact that changes are made on the actual imbalance data to convert it into a balanced dataset. This is usually achieved by either oversampling the positive class or down-sampling the negative class. However, a simple over sampling with replacement improves the performance of the ML model insignificantly (Bhagat & Patil, 2015). To solve this issue, (Chawla et al., 2002) introduced an oversampling technique known as Synthetic Minority Over-sampling Technique (SMOTE). In SMOTE, the positive class is over-sampled by introducing new random synthetic data instances along the line-segment. This allows to join together the k^{th} nearest neighbour and the positive data instance in consideration. The number of k^{th} neighbours are selected on the basis of the amount of oversampling needed. SMOTE generates a

Fig. 3 **a** Original cropped patches containing defects (pore and inclusion) and **b** ground truth for labeling (white for the pore and blue for the inclusion)



new synthetic data instances by using the following equation (Bhagat & Patil, 2015):

$$New(X_i) = X_i + (\hat{X}_i - X_i) * \delta \quad (2)$$

where X_i is a data instance from the positive class, \hat{X}_i is k^{th} nearest neighbour of X_i , and δ is a random number between $[0, 1]$.

In the present work, the labelled data provided by the expert were mainly unbalanced with the majority of the pixels analyzed falling in the back-ground and material classes. Such skewed data may results in poor classification capacity of the ML model. Moreover, an imbalanced testing dataset may lead to untrustworthy results. For these reason, minority data classes (porosity and inclusion) were over-sampled using SMOTE and a balanced dataset was adopted for training and testing. The value of k was chosen to be equal to 5, as smaller value leads to the generation of very similar synthetic data and higher value may lead to cross-over the boundary of

the class. The actual data distribution of the different classes and the results from the SMOTE are shown in Fig.4.

Machine learning models

In the context of non-destructive testing (NDT), where the cost of obtaining labeled data through CT scans and the expertise required for data annotation can be substantial, selecting suitable machine learning (ML) algorithms becomes crucial. This study focuses on defect detection in CT scans and aims to strike a balance between model performance, data requirements, and interpretability. While more sophisticated models like deep neural networks may offer high accuracy, they often demand large amounts of labeled data, which may not be feasible in NDT applications. Additionally, their lack of explainability raises concerns in critical industries where interpretability is essential. To overcome these challenges, a pragmatic approach is adopted by utilizing three supervised ML models to classify the pixel data into the four different classes: Decision Tree (DT), Support

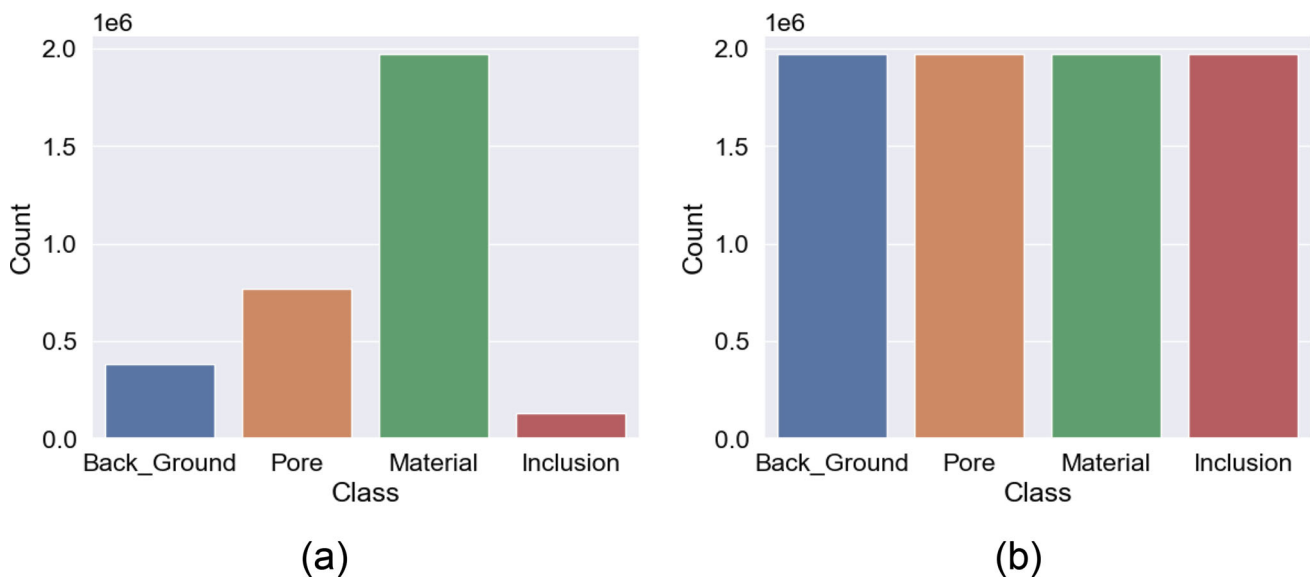


Fig. 4 A depiction of Skewed data-set **a** and the Balanced data-set **b** using SMOTE

Vector Machines (SVM), and Random Forest (RF). A popular state of the art python library named sci-kit (Nelli, 2015) was used to implement these models. They are known for their simplicity, requiring less data for training, and offering better interpretability. By leveraging these advantages, this study aims to develop effective defect detection models in CT scans while considering the constraints of limited data availability, high CT scan costs, and the need for explainability in NDT applications.

DT is a non-probabilistic classifier used to distinguish between classes of a given data (Douglass, 2020). The data obtained from the feature extraction pipeline of the cropped images with their labels are fed into the root node of the DT model as shown in Fig. 5. A random search optimization technique was used to obtain the optimal hyper-parameters to get the maximum gain at each level (entropy criterion is here adopted).

SVM, here used for pixels classification, is inherently a binary classifier (Douglass, 2020). In the present work, one vs all approach is used to train the SVM model to predict the class of the pixels. Hyper-parameters of the SVM model like 'C' and 'gamma' was tuned using k-folds methods with radial bias kernel.

RF algorithm works on the concept of ensemble learning. Multiple decision trees are employed to predict the class of the random batches of the training data set. The average of all the votes from different DTs is used as a final class. Hyper-parameters such as the number of trees, depth of trees and minimum data instances to split the node are tuned using a grid search method (Syarif et al., 2016) with predefined design space.

Once the models are trained, for each input cross-sectional image (see Fig. 6(a)) a label vector is obtained as output. This is concatenated and used to create an RGB image. The predicted results from DT, SVM and RF are depicted in Fig. 6(b), (c) and (d), respectively. The RGB image is constructed to provide visual aid for the user to identify the defect from the bulk. Information regarding different defects such as pores and inclusions are stored into different channels of an RGB image leading to a better binarization. This provides further information such as defect size and location.

Results and discussion

Testing and evaluation of ML models

To evaluate the different ML models, the trained algorithms are used to predict the class of the selected testing dataset. Predictions are then compared with the ground truth labels provided by the expert. Well known evaluation metrics such as precision, recall and F1-score are used to assess the performance of the different ML models. Moreover, metrics such as Receiver Operator Characteristics (ROC) and Area Under the Curve (AUC) were employed to calculate the performance of the ML models for a specific class. Finally, Intersection of Union (IoU), also known as Jaccard index, was applied to check the segmentation capability of the different algorithms.

Confusion matrix and derived units

Predicted results from ML models are compared with ground truth labels provided by the expert. Instances from this

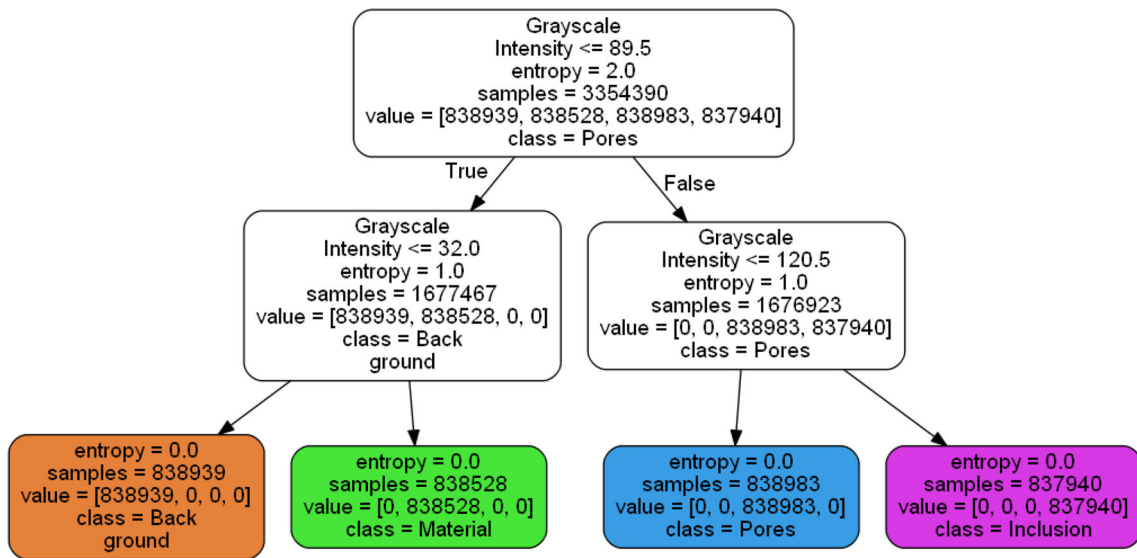
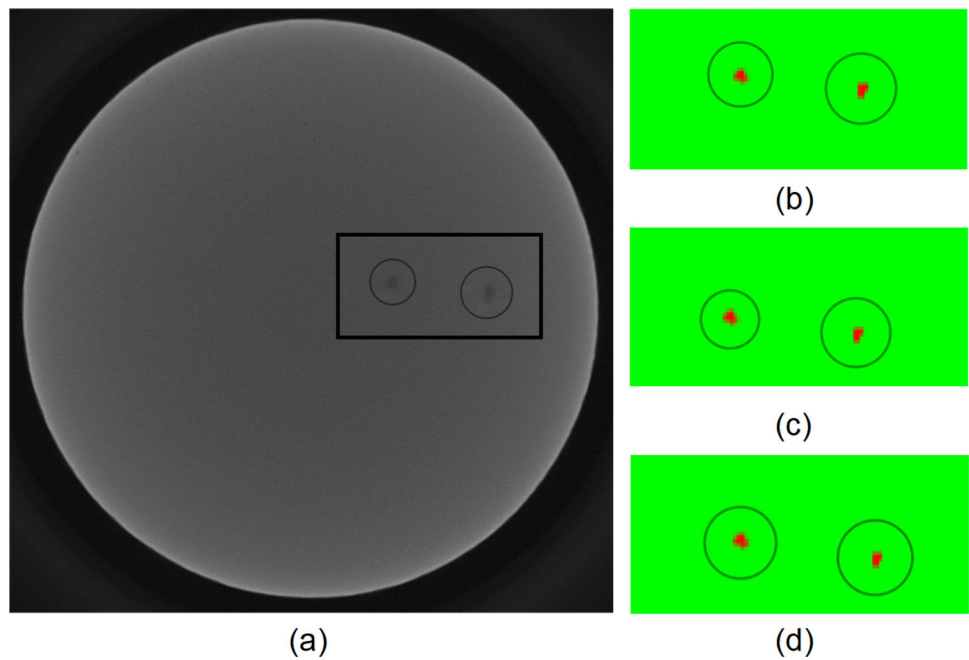


Fig. 5 Illustration of the trained Decision Tree model

Fig. 6 Results from the trained ML models: a Original cross-sectional CT image and the classification result from DT b, SVM c and RF d



comparison are categorized according to four classes: true positive (TP), true negative (TN), false positive (FP) and false negative (FN). These classes are later used to construct a Confusion Matrix (CM) such that all the rows represent the actual class and the columns describe the predicted class. An example of a CM for a binary class classification is shown in Fig. 7.

The FP and FN errors represent the type - 1 and type - 2 errors in statistics, respectively. A CM for multi-class classification can also be used to evaluate the performance of the models as in Fig. 8. In a typical CM, a strong diagonal is favourable suggesting the better performance of the trained

		Predicted	
		Class 1	Class 2
True	Class 1	True positive	False positive
	Class 2	False negative	True negative

Fig. 7 CM for a binary class classification result

ML model on the test dataset. In Fig. 8, elements of CM are normalized between 1 and 0, where 1 represents 100% of the

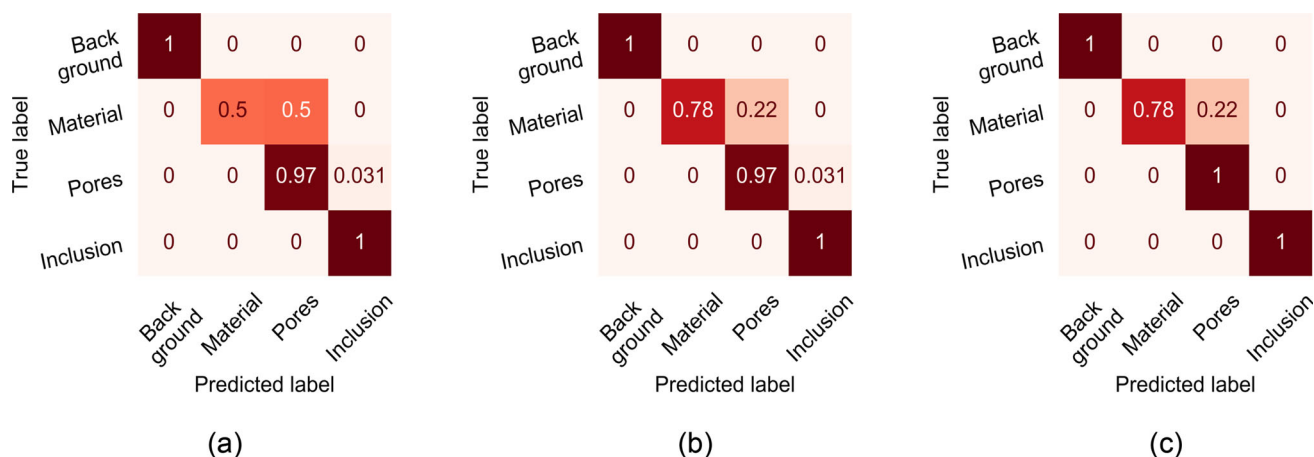


Fig. 8 CM of all the trained ML models: **a** DT, **b** RF and **c** SVM

testing instances of that particular class. The off-diagonal values represents the miss-classified data instances.

It has to be pointed out that the CM cannot be used as an absolute metric to compare the performance of the trained models. Indeed, different metrics obtained from the elements of the confusion matrix, such as precision, recall and F1-Score (Fawcett, 2006), were adopted:

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

$$F1 - Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{5}$$

TP, FP and FN here represent the instances in the evaluation of the testing data set.

The above metrics focus on different aspects of the model. The Precision metric emphasises more the true detection of each class, neglecting the effect of false detection (the maximum score value of 94.4% was achieved by the SVM model). Relying only on precision may lead to a model which is biased toward positive detection. On the other hand, Recall measures the result by emphasising on negative instances; SVM performed marginally better than all the other models by obtaining a score of 94.9%. Nevertheless, to obtain a balanced evaluation of the trained model both positive and negative detection are equally important. Therefore, F1-score was also adopted. This represents a harmonic mean of Precision and Recall providing more weight to lower value out of the two metrics. The maximum F1-score was obtained by the SVM model with a value of 94%. The macro average of all the metrics for all the three ML models are shown in Table 2.

Receiver operator characteristics curve

The derived metrics from the CM provide an average value of the performance of the different models in all the predicted classes. However, it is also essential to have an indicator of the performances in each of them, especially for pores and inclusions. Therefore, to have a performance indicator in each individual classes, ROC curve was employed. The ROC allows to obtain a visual representation of the ML model’s trade-off between correctly predicted positive samples and falsely predicted negative samples. Traditionally, a ROC curve assessment is a strategy to assess the performance of a binary ML classification model but it can be used for multi-class classification problems using the one vs all approach. For comparing the performances of all the models using ROC, Area Under the Curve (AUC) was used. Fig. 9 depicts the ROC and AUC of the three models on the test dataset for inclusion and pore classes.

As shown in Fig. 9(a), SVM and RF performs better than the DT model in the Pore class with a similar AUC of 0.89. Fig. 9(b) shows the exceptional performance of all the models in the Inclusion class due to a balanced test dataset. It can be concluded that the Inclusion detection is a relatively trivial task for a well trained model, as the contrast between inclusion and the rest of the classes is substantial.

Jaccard index

In the evaluation of standard ML models, predictions are customarily graded using statistical evaluation techniques such as CM, F1-score and ROC. However, for the evaluation of the image segmentation task, Intersection of Unions (IoU), aka Jaccard index, provides better insight. IoU is obtained from the ratio of the number of pixels in common between ground truth and predicted region by the total numbers of pixels present across both regions (Shi et al., 2014). In the

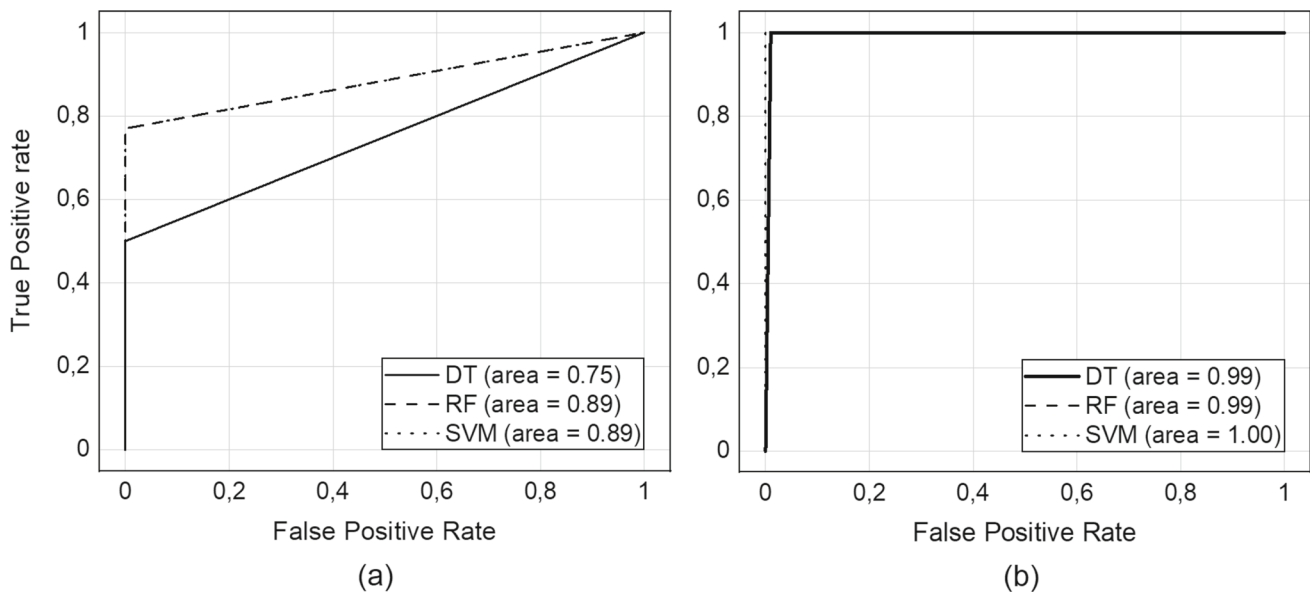


Fig. 9 ROC curve and AUC of the different ML models for **a** Pore and **b** Inclusion class

Table 2 Models performances from the testing evaluation data adopted

	Precision [%]	Recall [%]	F1-score [%]	AUC	IoU
SVM	94.4	94.9	94.4	0.94	0.97
DT	86.6	86.7	85.8	0.87	0.71
RF	93.7	93.7	93.7	0.94	0.82

present work, the ground truth images (see Fig. 10(b)) for the original patch (see Fig. 10(a)) provided by the expert are overlapped with the predicted results (see Fig. 10(c)) from the trained ML model. The Jaccard index score ranges from 0 to 1, where 1 represents a perfect match. For each model, the mean Jaccard index score of selected patches comprising both inclusions and pores is shown in Table 2.

Results suggest that the SVM model performs slightly better than DT and RF in all the evaluated metrics. Therefore, SVM was adopted to characterize the CT scan data of the third 3D-printed specimen.

Model explainability

Despite the exponential growth in the utilization of ML methodologies across diverse domains, they still possess a black box nature. Utilizing different post-hoc evaluation methods such as statistical analysis with confusion matrix, ROC and AUC metrics, feature importance analysis together with partial dependency plots can contribute towards improving a sense of confidence in the user. Indeed, ML models play a role in making crucial decisions, often in sensitive contexts. In order to effectively integrate them into critical

applications, establishing a trust factor with users becomes paramount. Such factors can be differentiated into two different but related aspects: prediction trusting and model trusting. Ribeiro et al. (2016) proposed a method to assert both of the trust aspects for a user.

Explainable Artificial Intelligence (XAI) aimed to improve the transparency, interpretability, and understandability of machine learning models for building trust in AI systems and ensuring that AI-driven decisions can be explained and justified. There are several methods one can use to tackle the explainability of the ML model depending on the type of explanation, type of data, and type of ML algorithm used.

SHAP (SHapley Additive exPlanations) (Lundberg & Lee, 2017) is a powerful and versatile framework for explaining the predictions of machine learning models. It provides a way to understand the contributions of individual features to model predictions, helping to interpret and debug complex models. However the computational cost of the SHAP increases exponentially with respect to the amount of data used to training and testing purposes.

Permutation Feature Importance is a technique used to assess the importance of individual features in a machine learning model. It provides insights into how much a model's performance would affect if the values of a specific feature were randomly shuffled while keeping other features same. In Teufl et al. (2021), authors have used Permutation feature importance method for gait kinematics analysis.

Permutation Feature Importance is quick and simple however, it provides limited insights. On the other, SHAP offers comprehensive explanations but can be computationally expensive. Therefore to strike a balance between the complexity and depth of the in-sight in the explanation, Local

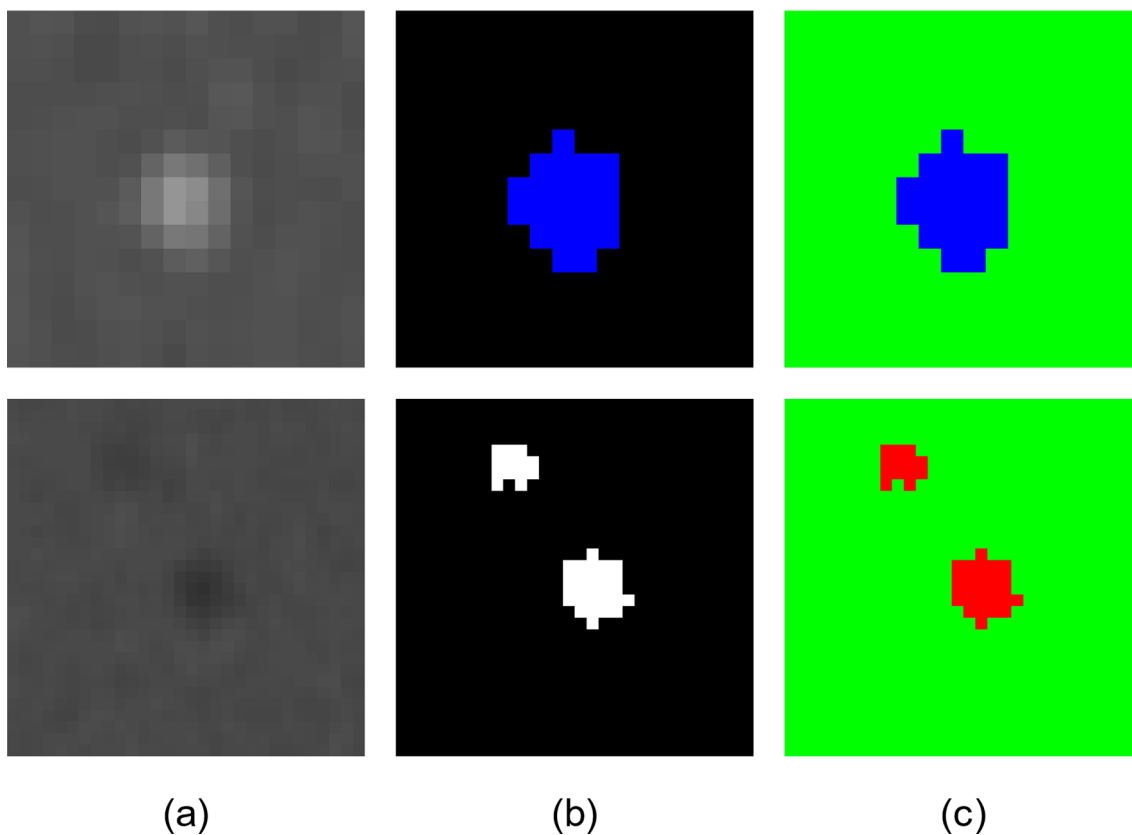
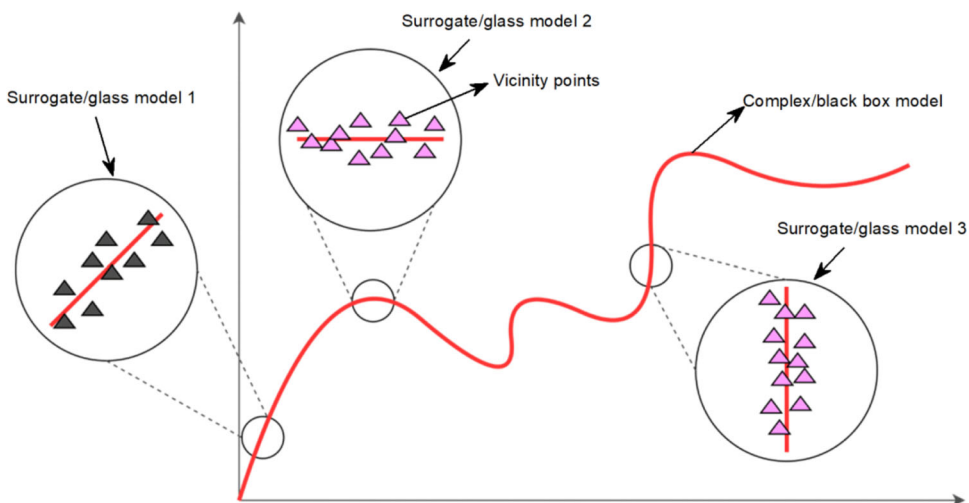


Fig. 10 A few examples of original cropped data **a**, ground truth **b** and predicted results **c** used to calculate IoU

Fig. 11 A graphical representation for a prediction using LIME’s glass surrogate model. A complex model [red line] is explained locally with LIME glass models using synthetic vicinity data points



Interpretable Model-agnostic Explanation (LIME) method was used

LIME is a technique used to provide explanations for individual predictions made by machine learning models. It aims to address the “black box” nature of many complex models by offering insights into ‘how the model arrives at a specific prediction?’.

When it comes to “Trusting a prediction,” LIME provides a local explanation for an individual instance by identify-

ing the features in the input data that are most influential in determining the prediction outcome. LIME achieves this by generating a simplified and interpretable (glass) model, such as linear regression, to approximate the behavior of the complex model locally in the vicinity of the instance of interest (see Fig. 11). By examining the coefficients or rules of this simplified model, we gain insights into which features played a significant role in the prediction. This explanation

helps users understand why a particular prediction was made and builds trust in the decision.

Additionally, LIME can also be utilized for “Trusting a model” by selecting multiple predictions and generating explanations for them. By analyzing multiple instances, LIME provides an overview of the model’s behavior and general decision-making patterns. This allows the user to identify any possible biases which results in a better-performing and robust model helping the user to make informed decisions.

The basic idea of LIME is to identify a glass model over the interpretable representation, which is locally faithful to the black box model under consideration. LIME provides an explanation for a glass box that may not be able to approximate the black box model globally but in the vicinity of an individual data instance from the test dataset. The explanation produced by LIME is obtained by minimizing the function $\xi(x)$.

$$\xi(x) = \operatorname{argmin}_{g \in G} \mathcal{L}(f, g, \pi_x) + \Omega(g) \quad (6)$$

Where G is a set of glass models replicating the black box ML model f and g is an individual glass model from G locally faithful to f in the vicinity π_x of the data instance x . $\Omega(g)$ represents the penalty for the complexity of the model g . Minimizing this function results in the generation of locally interpretable glass box models that approximate the behavior of black box models for a given vicinity point.

An individual LIME evaluation for a data instance from the testing dataset is shown in Fig. 12. The figure provides qualitative reasoning for a particular decision using a set of conditions to classify a pixel into respective classes. It can be inferred that the LIME glass model has successfully learned a few thresholds for every data class. As in Fig. 12(a), pixels with a grayscale intensity less than or equal to 68, are more likely to fall into the pore class. It also provides a sense of confidence by providing weight for each feature that contributes towards the prediction. Similarly, Fig. 12 shows the related feature contribution with their weights for all the classes. Such evaluation helps to ensure the “Trusting a prediction” factor.

By utilizing LIME to generate local explanations for multiple instances, users can assess the consistency and reliability of these explanations across multiple data points. When the explanations consistently align with their domain knowledge and expectations, it serves as evidence supporting the model’s reliability in making accurate predictions. Through an examination of numerous instances and their corresponding explanations, users can establish “Trusting a model” factor. Ultimately, the transparency and interpretability provided by LIME’s local explanations help users to make well-informed judgments regarding the model’s overall performance and instill confidence in its predictions.

LIME employs an additive approach to build indirect global explanations by accumulating trust across various samples. However, it is important to note that LIME has limitations. One such limitation is its scalability, as higher-dimensional data can lead to increased computational costs, making it challenging to generate explanations efficiently. Furthermore, LIME is susceptible to sample dependence, where in different samples may yield varying explanations, adding uncertainty to the interpretability process.

Clustering process and defects characterization

The ML model classifies all the pixels according to the defect classes (pores and inclusions). However, it is necessary to cluster the different pixels into their respective defects. To do that, un-supervised ML techniques were adopted.

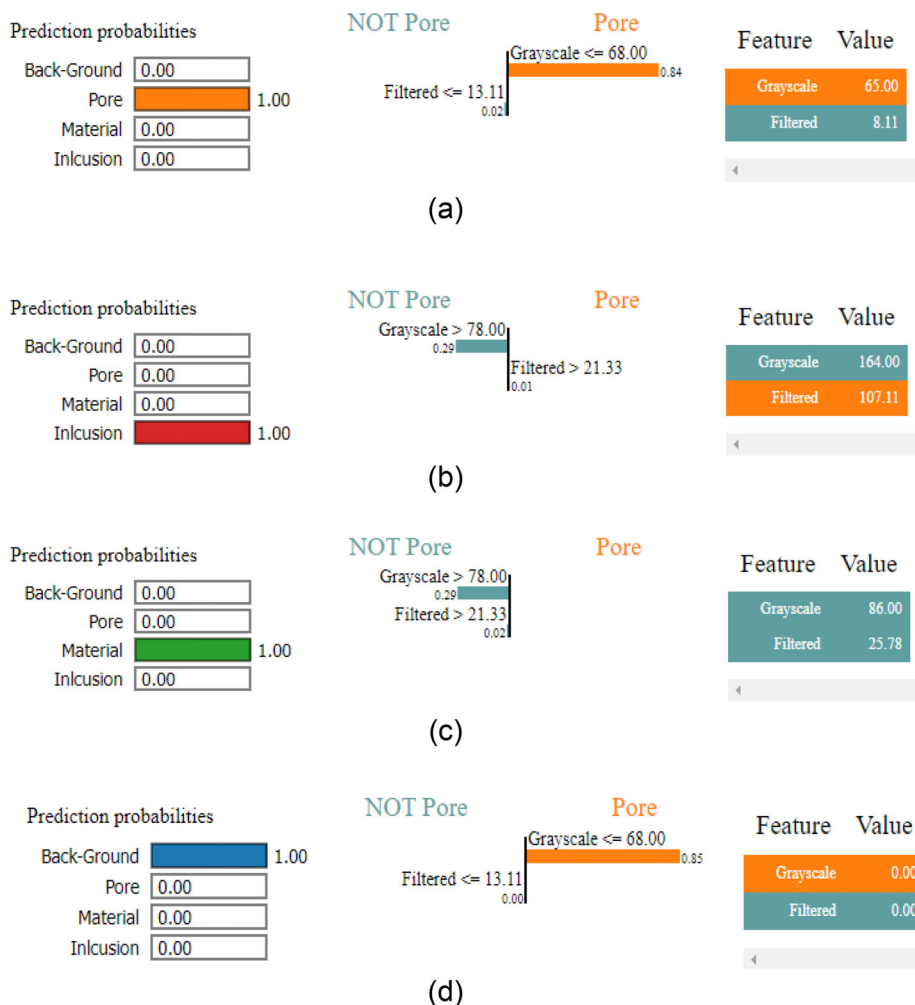
2D defect clustering

The layer-wise results obtained from the SVM model are subjected to cluster analysis using K-Means to identify distinct clusters for individual defects. This analysis utilizes the Euclidean distance metric to systematically separate different defects. While K-Means typically requires the predetermined number of clusters (K), this information can be determined using clustering quality measures such as the silhouette score. By leveraging these measures, the appropriate number of clusters can be determined to facilitate an effective clustering process.

As an example, results for a specific layer consisting of multiple defects are analyzed and shown in Fig. 13. The pixels classified as pores in a specific layer, are clustered into distinct defects using K-means and silhouette score. Fig. 13(a) depicts the original gray scale image with four distinct pores. All the pixels are classified using SVM and the locations and number of all the pixels classified as pores are later analysed by K-means. The silhouette score analysis suggesting four distinct clusters aid the K-mean algorithm to determine the optimal value of ‘ K ’ to cluster the pixels into distinct defects (see Fig. 13(c)). Each distinct cluster may later be used to obtain information regarding individual defects in the specific layer such as location (see Fig. 13(b) with cross-hair as the center), size and shape characteristics (Montero & Bribiesca, 2009).

The aforementioned layer-wise evaluation can be extended to a 3D cluster analysis. However, in higher dimensional data, the accuracy of the K-Means can suffer poorly as the sphericity of the clusters decreases. Moreover, the presence of any outlier may affect the cluster quality, and this may result in merging multiple defects and the outliers into a single defect. The clustering metric “ k ” can be obtained using the

Fig. 12 A graphical representation for a prediction using LIME’s glass surrogate model



silhouette score method for a range of potential defects. Nevertheless, the information regarding such a range of defects in a scanned volume must be known a priori. A faulty range of clustering metrics can yield poor clustering results. To obtain a complete 3D description of the defects and to overcome the mentioned limitations in the K-Means application, a sophisticated unsupervised ML method is used.

3D defect clustering

For 3D defects evaluation, layer-wise pixel data are stacked on top of each other in the build direction to construct a 3D point cloud with all the identified pixels in the complete scan volume. To analyze such point cloud, a more sophisticated unsupervised ML model known as Density-Based Spatial Clustering with Application of Noise (DBSCAN) (Ester et al., 1996; Liu et al., 2007) is used with a clustering score of 0.73. Representative-based clustering methods like K-Means are suitable for finding spherical and ellipsoidal-shaped clusters. However, when it comes to arbitrary-shaped clusters, density-based techniques are more efficient. Fur-

ther, K-means clustering can lead to misleading results when it comes to identifying defects, as it tends to incorrectly cluster noise pixels with actual defects. Consequently, this can introduce inaccuracies in the derived morphological information associated with the defects. Therefore DBSCAN is later used to evaluate the 3D defects.

DBSCAN is a density-based un-supervised ML technique which uses the local density of the points to determine the clusters rather than using only the distance between them. DBSCAN include the use of two hyperparameters: Eps and minPts. These parameters are used to divide the dataset into three categories: (a) core point, (b) border point, and (c) outlier. A point is said to be a core if there are at least a MinPts number of data points within its Eps radius. A border point is defined such that it is reachable from a core point, but do not have MinPts number of point within its Eps radius. Finally, a point is said to be an outlier if it is neither reachable from a core point nor it has MinPts number of points within its Eps radius.

The optimal Eps for a dataset can be obtain from the 3rd nearest neighbor plot for all the data points (Liu et al., 2007).

Fig. 13 Results from unsupervised ML model: **a** original image from the CT scan, **b** defect pixels clustered using Kmeans (the different pixels group with their cluster center are highlighted) and **c** silhouette score for the number of clusters determination

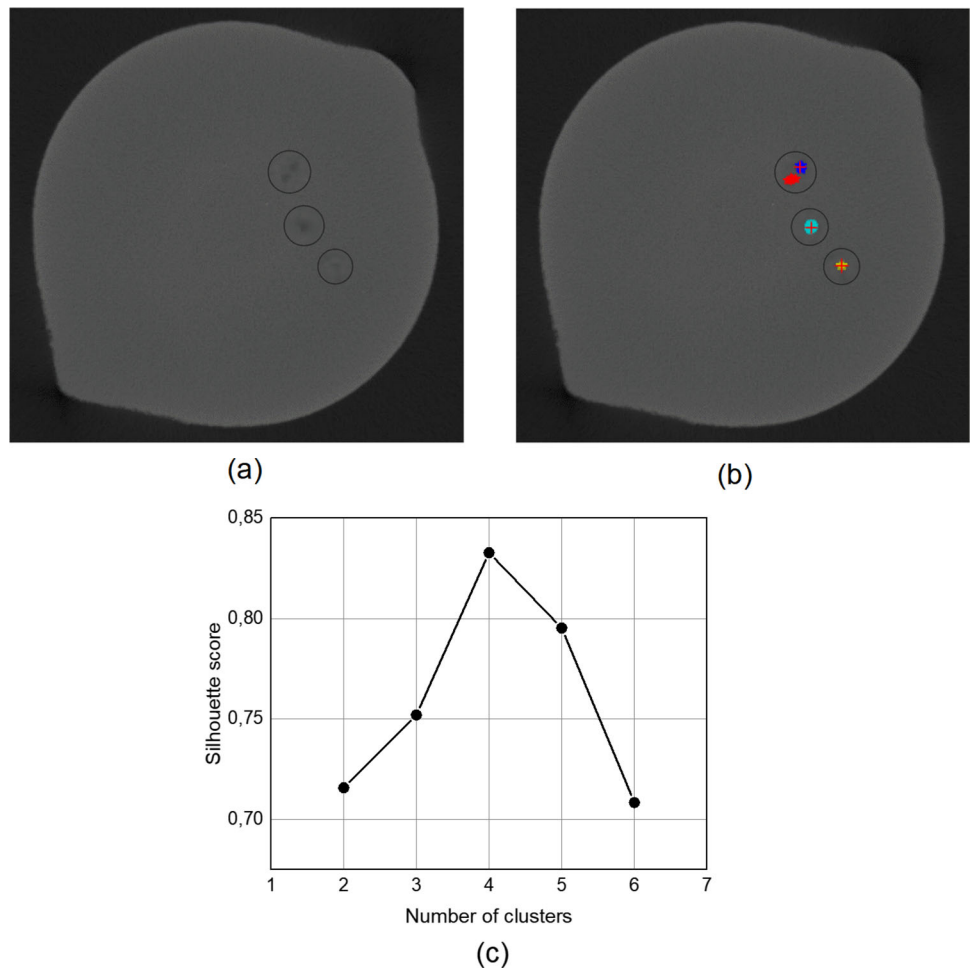


Figure 14 suggests that the 3D point cloud data for the pixels in the Pore class have one density which can be obtained from the elbow point of the curve. The minPts parameter value was chosen higher than three times the size of the voxel adopted in the CT scan to eliminate possible noises.

Thanks to DBSCAN, 3D point clouds can be clustered into distinct defects. Moreover, groups of data points or pixels, which cannot be clustered due to selected hyper-parameters are classified as noise. For instance, several CT artifacts like streaks and beam hardening may result in a sudden change in intensities for a small region of pixels and therefore they are also classified as outliers.

Defect characterization

Defects clustered using DBSCAN may be analyzed individually to characterize their location, size and shape. All the identified defects in the scanned volume are shown in Fig. 15(a). To analyze each defect individually and to build up a surface mesh, convex hull was adopted (see Fig. 15(b)).

The application of the convex hull facilitates the evaluation of directional dependent size such as the projected area

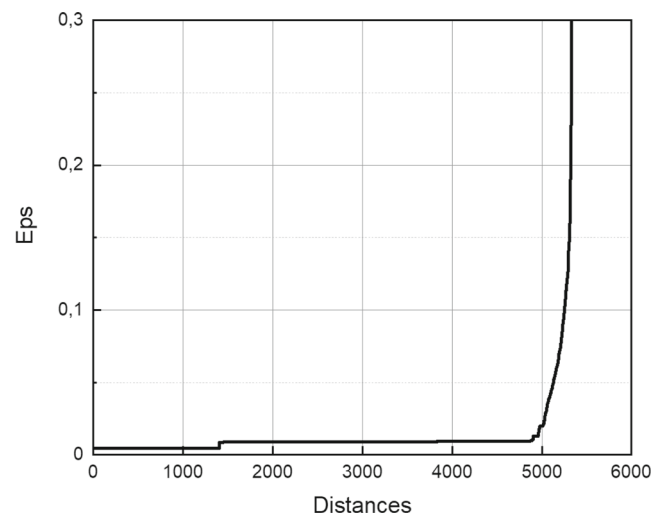
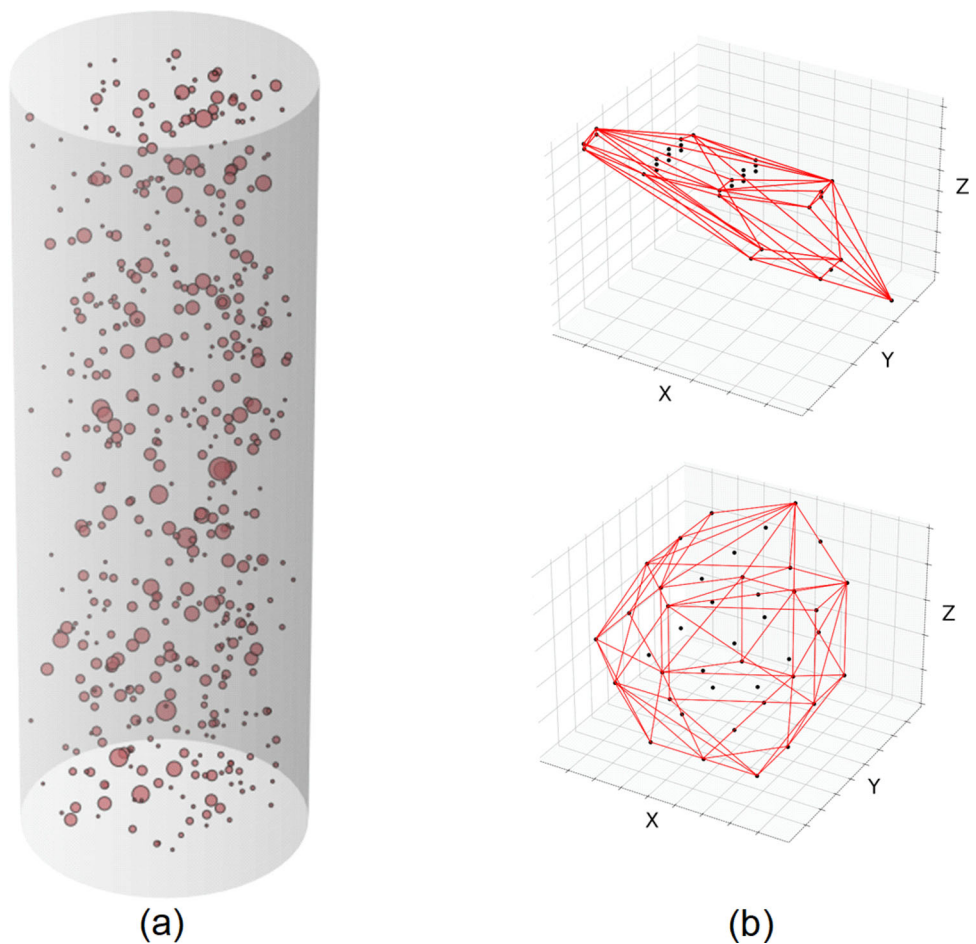


Fig. 14 Elbow point for optimal Eps parameter using 3rd K-nearest neighbor for the specimen not used in the training and testing process

of the defect onto all three primary planes (see Fig. 16(a)). Similarly, projected length, which is the projection of a defect onto primary axes, can be calculated for every single defect

Fig. 15 Result from 3D clustering using DBSCAN: **a** defects detected in the complete scan volume, **b** individual defect analyzed by Convex hull



(see Fig. 16(b)). Other morphological characteristics of the defects, such as surface area and volume can also be obtained; this provides a crucial aspect for the 3D defect shape characterization such as sphericity and compactness of the defect.

Each of the clusters defined by DBSCAN can be traced back to the corresponding layers in the original CT scan image data. In Fig. 17, a few original cropped CT scan images are shown containing a small section of the same defects in multiple layers together with the results obtained by the automatic ML detection tool. For completeness, the defect defined using DBSCAN is also shown.

Conclusion and outlook

The paper presents a novel methodology for automatically detecting and analyzing internal defects in CT scan data of additive manufacturing (AM) components. Key points and findings from the paper can be summarized as follows:

- Existing methodologies for defect detection in CT scan data are discussed, highlighting their limitations;

- Three traditional machine learning (ML) models were trained and developed using an explainable AI (XAI) framework, utilizing popular Python libraries such as scikit learn and LIME;
- The trained XAI models were evaluated using various ML model evaluation techniques, including confusion matrix, precision, recall, F-1 Score, and ROC curve analysis with AUC;
- The segmentation quality of the best-performing model was evaluated using the Jaccard index, a traditional segmentation evaluation technique;
- A sophisticated spatial clustering technique (DBSCAN) was employed to cluster 3D point clouds of defects in the scanned volume;
- A methodology was proposed to analyze the results obtained from the unsupervised ML technique, providing information such as defect location, size, projected areas, and lengths onto primary planes and axes.

It is worth noting that the tool's sensitivity in detecting minimum defect sizes depends on the resolution of the CT scan used. Future research will be dedicated to analyzing different materials and CT conditions in trying to

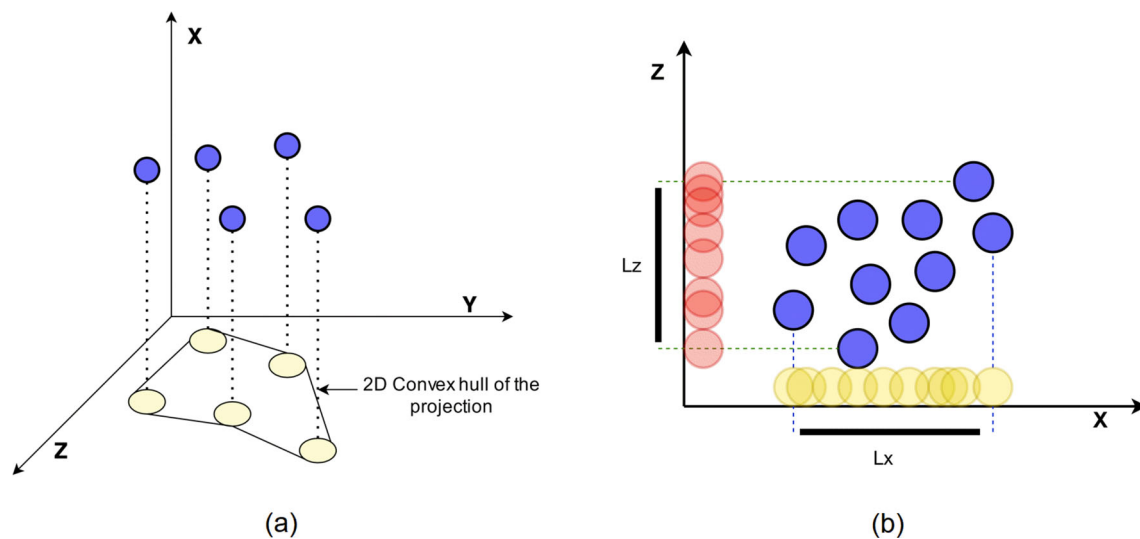


Fig. 16 Schematic representation of the projected area evaluation by convex hull **a** and the projected lengths for a single defect into primary axes **b**

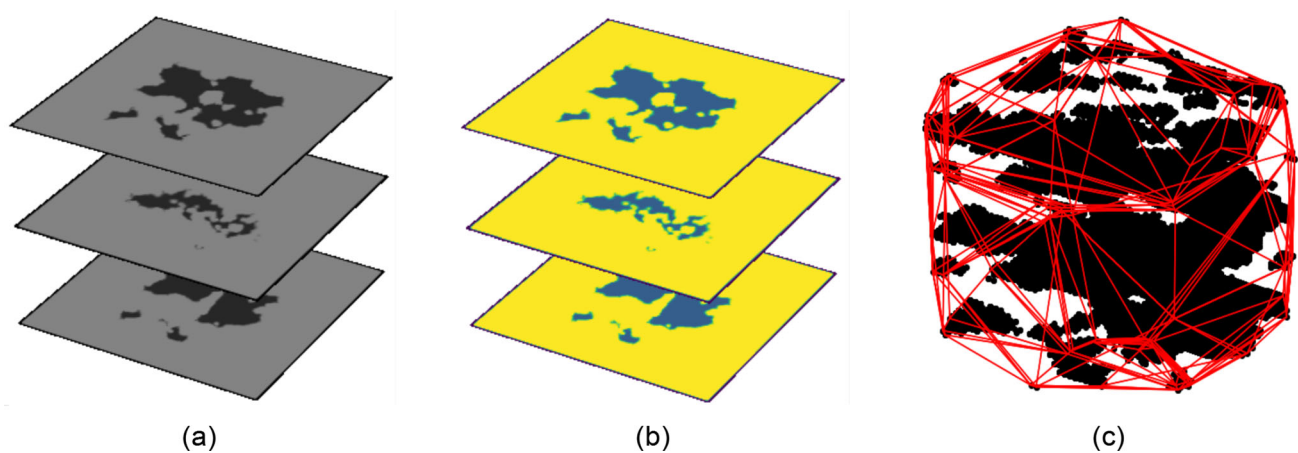


Fig. 17 Cropped patches of multiple CT scan layers showing several sections of the same defect **a** and the prediction obtained by the ML algorithm **b**. The complete defect in a 3D space is shown for completeness **c**

enlarge the database adopted. Indeed, the predictions of the trained model may be influenced when applied to materials with different compositions than those learned during the training. The X-ray absorption characteristics of various materials contribute to the gray-scale intensities of pixels in the recorded data. Moreover, variations in CT machine parameters, such as beam parameters, can lead to differences in the absorption rates, thereby resulting in varying gray-scale intensities of pixels.

The proposed methodology establishes a groundwork for automatic defect detection in AM components, highlighting the use of XAIs and providing guidelines for ML certification in CT analysis. The defect detection tool could be a valid support to enable the development of online process monitoring

tools by comparing the obtained results with sensor data recorded during the printing. Overall, the proposed methodology shows promising results for enhancing AM component quality control.

Author Contributions H. Bordekar: Conceptualization, Methodology, Software, Writing - Original draft, Writing - editing, Validation. N. Cersullo: Conceptualization, Methodology, Writing - Original draft, Writing - editing, Validation. M. Brysch: Methodology, Writing - editing, Validation. J. Philipp: Validation, Software, Writing - Original draft, Investigation. C. Huehne: Supervision, Writing - reviewing, Validation, Funding acquisition.

Funding Open Access funding enabled and organized by Projekt DEAL. The authors would like to acknowledge the funding by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC 2163/1 - Sustainable and Energy Efficient Aviation - Project-ID 390881007.

Data availability The data used in the presented work can be made available upon request.

Declarations

Conflict of interest The authors declare that they have no competing financial interests or personal relationships that could influence the presented work

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Bacoiu, D., Melton, G., Papaalias, M., & Shaw, R. (2019). Automated defect classification of SS304 TIG welding process using visible spectrum camera and machine learning. *NDT & E International*, *107*, 102139. <https://doi.org/10.1016/j.ndteint.2019.102139>
- Beretta, S., & Romano, S. (2017). A comparison of fatigue strength sensitivity to defects for materials manufactured by AM or traditional processes. *International Journal of Fatigue*, *94*, 178–191. <https://doi.org/10.1016/j.ijfatigue.2016.06.020>
- Bhagat, R. C., & Patil, S. S. (2015). Enhanced SMOTE algorithm for classification of imbalanced big-data using Random Forest. 2015 IEEE International Advance Computing Conference (IACC). <https://doi.org/10.1109/iadcc.2015.7154739>
- Boas, F. E., & Fleischmann, D. (2012). CT artifacts: causes and reduction techniques. *Imaging in Medicine*, *4*(2), 229–240. <https://doi.org/10.2217/iim.12.13>
- Burkart, N., & Huber, M. F. (2021). A Survey on the Explainability of Supervised Machine Learning. *Journal of Artificial Intelligence Research*, *70*, 245–317. <https://doi.org/10.1613/jair.1.12228>
- Bussmann, N., Giudici, P., Marinelli, D., & Papenbrock, J. (2020). Explainable Machine Learning in Credit Risk Management. *Computational Economics*, *57*(1), 203–216. <https://doi.org/10.1007/s10614-020-10042-0>
- Caggiano, A., Zhang, J., Alfieri, V., Caiazzo, F., Gao, R., & Teti, R. (2019). Machine learning-based image processing for on-line defect recognition in additive manufacturing. *CIRP Annals*, *68*(1), 451–454. <https://doi.org/10.1016/j.cirp.2019.03.021>
- Cersullo, N., Mardaras, J., Emile, P., Nickel, K., Holzinger, V., & Hühne, C. (2022). Effect of Internal Defects on the Fatigue Behavior of Additive Manufactured Metal Components: A Comparison between Ti6Al4V and Inconel 718. *Materials*, *15*(19), 6882. <https://doi.org/10.3390/ma15196882>
- Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, *16*, 321–357. <https://doi.org/10.1613/jair.953>
- De Chiffre, L., Carmignato, S., Kruth, J.-P., Schmitt, R., & Weckenmann, A. (2014). Industrial applications of computed tomography. *CIRP Annals*, *63*(2), 655–677. <https://doi.org/10.1016/j.cirp.2014.05.011>
- Dougllass, M. J. J. (2020). Book Review: Hands-on Machine Learning with Scikit-Learn, Keras, and Tensorflow. In Aurélien Géron (ed). *Physical and Engineering Sciences in Medicine*, 2nd edition. 43(3), 1135–1136. <https://doi.org/10.1007/s13246-020-00913-z>
- du Plessis, A., le Roux, S. G., Booyesen, G., & Els, J. (2016). Directionality of cavities and porosity formation in powder-bed laser additive manufacturing of metal components investigated using X-ray tomography. *3D Printing and Additive Manufacturing* *3*(1): 48–55. <https://doi.org/10.1089/3dp.2015.0034>
- du Plessis, A., Sperling, P., Beerlink, A., Tshabalala, L., Hoosain, S., Mathe, N., & le Roux, S. G. (2018). Standard method for microCT-based additive manufacturing quality control 1: Porosity analysis. *MethodsX*, *5*, 1102–1110. <https://doi.org/10.1016/j.mex.2018.09.005>
- Du, W., Shen, H., Fu, J., Zhang, G., & He, Q. (2019). Approaches for improvement of the X-ray image defect detection of automobile casting aluminum parts based on deep learning. *NDT & E International*, *107*, 102144. <https://doi.org/10.1016/j.ndteint.2019.102144>
- Erhan, D., Szegedy, C., Toshev, A., & Anguelov, D. (2014). Scalable Object Detection Using Deep Neural Networks. 2014 IEEE Conference on Computer Vision and Pattern Recognition. <https://doi.org/10.1109/cvpr.2014.276>
- Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, *96*(34), 226–231.
- Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, *27*(8), 861–874. <https://doi.org/10.1016/j.patrec.2005.10.010>
- Fuchs, P., Kröger, T., Dierig, T., Garbe, C. S. (2019). Generating Meaningful Synthetic Ground Truth for Pore Detection in Cast Aluminum Parts. *E-Journal of Nondestructive Testing*. <https://doi.org/10.58286/23730>
- Gibson, I., Rosen, D. W., Stucker, B., Khorasani, M., Rosen, D., Stucker, B., & Khorasani, M. (2021). *Additive manufacturing technologies* (Vol. 17). Cham, Switzerland: Springer. <https://doi.org/10.1007/978-3-030-56127-7>
- Gobert, C., Kudzal, A., Sietins, J., Mock, C., Sun, J., & McWilliams, B. (2020). Porosity segmentation in X-ray computed tomography scans of metal additively manufactured specimens with machine learning. *Additive Manufacturing*, *36*, 101460. <https://doi.org/10.1016/j.addma.2020.101460>
- Gong, H., Nadimpalli, V. K., Rafi, K., Starr, T., & Stucker, B. (2019). Micro-CT Evaluation of Defects in Ti-6Al-4V Parts Fabricated by Metal Additive Manufacturing. *Technologies*, *7*(2), 44. <https://doi.org/10.3390/technologies7020044>
- Greitemeier, D., Palm, F., Syassen, F., & Melz, T. (2017). Fatigue performance of additive manufactured TiAl6V4 using electron and laser beam melting. *International Journal of Fatigue*, *94*(2017), 211–217. <https://doi.org/10.1016/j.ijfatigue.2016.05.001> <https://www.bakerhughesds.com/industrial-x-ray-ct-scanners/phoenix-vtomex-s-micro-ct>, note = Accessed: 2023-11-02
- Kasperovich, G., Haubrich, J., Gussone, J., & Requena, G. (2016). Correlation between porosity and processing parameters in TiAl6V4 produced by selective laser melting. *Materials & Design*. <https://doi.org/10.3390/met8100830>
- Koester, L. W., Bond, L. J., Taheri, H., & Collins, P. C. (2019). Non-destructive evaluation of additively manufactured metallic parts. *Additive Manufacturing for the Aerospace Industry*. <https://doi.org/10.1016/b978-0-12-814062-8.00020-0>
- Linardatos, P., Papastefanopoulos, V., & Kotsiantis, S. (2020). Explainable AI: A Review of Machine Learning Interpretability Methods. *Entropy*, *23*(1), 18. <https://doi.org/10.3390/e23010018>

- Linardatos, P., Papastefanopoulos, V., & Kotsiantis, S. (2020). Explainable AI: A Review of Machine Learning Interpretability Methods. *Entropy*, 23(1), 18. <https://doi.org/10.3390/e23010018>
- Liu, P., Zhou, D., & Wu, N. (2007). VDBSCAN: Varied Density Based Spatial Clustering of Applications with Noise. 2007 International Conference on Service Systems and Service Management. <https://doi.org/10.1109/icsssm.2007.4280175>
- Liu, P., Zhou, D., & Wu, N. (2007). VDBSCAN: Varied Density Based Spatial Clustering of Applications with Noise. 2007 International Conference on Service Systems and Service Management. <https://doi.org/10.1109/icsssm.2007.4280175>
- Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30.
- Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., Katz, R., Himmelfarb, J., Bansal, N., & Lee, S.-I. (2020). From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, 2(1), 56–67. <https://doi.org/10.1038/s42256-019-0138-9>
- Montero, R. S., & Bribiesca, E. (2009). State of the art of compactness and circularity measures. In *International mathematical forum* (Vol. 4, No. 27, pp. 1305–1335).
- Mutiargo, B., Pavlovic, M., Malcolm, A. A., Goh, B., Krishnan, M., Shota, T., and Putro, M. I. S. (2019). Evaluation of X-ray Computed Tomography (CT) images of additively manufactured components using deep learning. In *Proceedings of the 3rd Singapore International Non-Destructive Testing Conference and Exhibition (SINCE2019)*, Singapore (p. 9).
- Nelli, F. (2015). Machine Learning with scikit-learn. *Python Data Analytics*. <https://doi.org/10.1007/978-1-4842-0958-5-8>
- Otsu, N. (1979). A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1), 62–66. <https://doi.org/10.1109/tsmc.1979.4310076>
- Ribeiro, M., Singh, S., & Guestrin, C. (2016). Why Should I Trust You?: Explaining the Predictions of Any Classifier. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*. <https://doi.org/10.18653/v1/n16-3020>
- Sadon, T. M. (2021). *Classification of medical images based on deep learning network (CNN) for both brain tumors and covid-19*. Diss: Ministry of Higher Education.
- Samek, W., & Müller, K.-R. (2019). Towards Explainable Artificial Intelligence. *Lecture Notes in Computer Science*. <https://doi.org/10.1007/978-3-030-28954-6-1>
- Schindelin, J., Arganda-Carreras, I., Frise, E., Kaynig, V., Longair, M., Pietzsch, T., Preibisch, S., Rueden, C., Saalfeld, S., Schmid, B., Tinevez, J.-Y., White, D. J., Hartenstein, V., Eliceiri, K., Tomancak, P., & Cardona, A. (2012). Fiji: an open-source platform for biological-image analysis. *Nature Methods*, 9(7), 676–682. <https://doi.org/10.1038/nmeth.2019>
- Schlotterbeck, M., Schulte, L., Alkhaldi, W., Krenkel, M., Toeppe, E., Tschechne, S., & Wojek, C. (2020). Automated defect detection for fast evaluation of real inline CT scans. *Nondestructive Testing and Evaluation*, 35(3), 266–275. <https://doi.org/10.1080/10589759.2020.1785446>
- Seeram, E., & Sil, J. (2013). Computed Tomography: Physical Principles, Instrumentation, and Quality Control. *Practical SPECT/CT in Nuclear Medicine*. <https://doi.org/10.1007/978-1-4471-4703-9-5>
- Seifi, M., Gorelik, M., Waller, J., Hrabe, N., Shamsaei, N., Daniewicz, S., & Lewandowski, J. J. (2017). Progress Towards Metal Additive Manufacturing Standardization to Support Qualification and Certification. *JOM*, 69(3), 439–455. <https://doi.org/10.1007/s11837-017-2265-2>
- Seifi, M., Salem, A., Beuth, J., Harrysson, O., & Lewandowski, J. J. (2016). Overview of Materials Qualification Needs for Metal Additive Manufacturing. *JOM*, 68(3), 747–764. <https://doi.org/10.1007/s11837-015-1810-0>
- Shi, R., Ngan, K. N., & Li, S. (2014). Jaccard index compensation for object segmentation evaluation. 2014 IEEE International Conference on Image Processing (ICIP). <https://doi.org/10.1109/icip.2014.7025904>
- Shipway, N. J., Huthwaite, P., Lowe, M. J. S., & Barden, T. J. (2021). Using ResNets to perform automated defect detection for Fluorescent Penetrant Inspection. *NDT & E International*, 119, 102400. <https://doi.org/10.1016/j.ndteint.2020.102400>
- Singh, A., Sengupta, S., & Lakshminarayanan, V. (2020). Explainable Deep Learning Models in Medical Image Analysis. *Journal of Imaging*, 6(6), 52. <https://doi.org/10.3390/jimaging6060052>
- Spierings, A. B., Schneider, M., & Eggenberger, R. (2011). Comparison of density measurement techniques for additive manufactured metallic parts. *Rapid Prototyping Journal*, 17(5), 380–386. <https://doi.org/10.1108/13552541111156504>
- Syarif, I., Prugel-Bennett, A., Wills, G. (2016). SVM Parameter Optimization using Grid Search and Genetic Algorithm to Improve Classification Performance. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, 14(4), 1502. <https://doi.org/10.12928/telkomnika.v14i4.3956>
- Teuffl, W., Taetz, B., Miezal, M., Dindorf, C., Fröhlich, M., Trinler, U., Hogan, A., & Bleser, G. (2021). Automated detection and explainability of pathological gait patterns using a one-class support vector machine trained on inertial measurement unit based gait data. *Clinical Biomechanics*, 89, 105452. <https://doi.org/10.1016/j.clinbiomech.2021.105452>
- Torbati-Sarraf, H., Niverty, S., Singh, R., Barboza, D., De Andrade, V., Turaga, P., & Chawla, N. (2021). Machine-Learning-based Algorithms for Automated Image Segmentation Techniques of Transmission X-ray Microscopy (TXM). *JOM*, 73(7), 2173–2184. <https://doi.org/10.1007/s11837-021-04706-x>
- Torens, C., Durak, U., & Dauer, J. C. (2022). Guidelines and Regulatory Framework for Machine Learning in Aviation. *AIAA SCITECH 2022 Forum*. <https://doi.org/10.2514/6.2022-1132>
- Vargas-Lopez, O., Perez-Ramirez, C. A., Valtierra-Rodriguez, M., Yanez-Borjas, J. J., & Amezcua-Sanchez, J. P. (2021). An Explainable Machine Learning Approach Based on Statistical Indexes and SVM for Stress Detection in Automobile Drivers Using Electromyographic Signals. *Sensors*, 21(9), 3155. <https://doi.org/10.3390/s21093155>
- Wong, V. W. H., Ferguson, M., Law, K. H., Lee, Y.-T. T., & Witherell, P. (2021). Segmentation of Additive Manufacturing Defects Using U-Net. *Volume 2: 41st Computers and Information in Engineering Conference (CIE)*. <https://doi.org/10.1115/detc2021-68885>
- Yadollahi, A., & Shamsaei, N. (2017). Additive manufacturing of fatigue resistant materials: Challenges and opportunities. *International Journal of Fatigue*. <https://doi.org/10.17485/ijst/2015/v8i14/68808>
- Yap, C. Y., Chua, C. K., Dong, Z. L., Liu, Z. H., Zhang, D. Q., Loh, L. E., & Sing, S. L. (2015). Review of selective laser melting: Materials and applications. *Applied physics reviews*, 2(4), 041101. <https://doi.org/10.1063/1.4935926>
- Zhu, P., Cheng, Y., Banerjee, P., Tamburrino, A., & Deng, Y. (2019). A novel machine learning model for eddy current testing with uncertainty. *NDT & E International*, 101, 104–112. <https://doi.org/10.1016/j.ndteint.2018.09.010>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.