https://doi.org/10.1038/s44334-025-00021-4

An unsupervised approach towards promptable porosity segmentation In laser powder bed fusion by segment anything

Check for updates

Israt Zarin Era¹, Imtiaz Ahmed¹ \square , Zhichao Liu¹ \square & Srinjoy Das^{1,2} \square

X-ray Computed Tomography (XCT) plays a vital role in characterizing internal porosity in laser Powder Bed Fusion (PBF-L) parts, where defects critically impact mechanical properties. Traditional methods for porosity segmentation rely on supervised machine learning, which requires large labeled datasets and high computational resources, limiting scalability. Foundation Models, pre-trained on diverse datasets, offer a flexible alternative by enabling high accuracy through user-provided prompts with minimal finetuning. This study develops a novel unsupervised porosity segmentation framework based on the Segment Anything Model (SAM), a Vision Transformer-based Foundation Model. Utilizing a multi-point prompt generation scheme with unsupervised clustering, it achieves a Dice Similarity Coefficient (DSC) of over 80%, enabling efficient defect segmentation without labeled data. Additionally, the framework's performance is validated across varied prompt sets by bootstrapping for uncertainty quantification. By addressing scalability and automation challenges, this work highlights the transformative potential of Foundation Models in enhancing XCT-based porosity characterization for PBF-L parts.

Laser Powder Bed Fusion (PBF-L)¹ is a Laser Additive Manufacturing (LAM) technique increasingly being used for producing complex parts across industries like aerospace, automotive, and biomedical. Its strength lies in its ability to fabricate components with intricate geometries² and achieves fine resolution and complex geometries, with precision depending on process parameters and post-processing³. In PBF-L, a thin layer of metal powder is spread across a level bed, where it is selectively melted using a highly focused laser beam. Despite its wide adoption in various industries, PBF-L is prone to process-induced defects, such as porosity, cracking, and geometric distortions. Porosity is a common defect in PBF-L that occurs mainly due to a lack of fusion, entrapped gas, and keyholes^{4,5}. It negatively affects mechanical properties like tensile strength, stiffness, and hardness, thereby compromising the quality of the final product⁶. Non-destructive testing methods, like high-resolution X-ray Computed Tomography (XCT), are widely used to isolate and locate pores in fabricated parts⁴⁻⁷. However, identifying and measuring internal porosity from XCT data manually can be arduous and challenging. With the advent of Machine Vision⁸ and Industry 4.0⁹, different image processing^{10,11} and machine learning (ML)-based techniques^{12,13} have been widely adopted for such tasks, significantly improving the quality of defect segmentation in different fields of manufacturing, including PBF-L.

Semantic segmentation¹⁴ is a computer vision task that assigns a label to every pixel and helps to understand a pixel-wise mapping of the contents

in an image. In the fast-evolving field of AI, image segmentation in manufacturing has seen significant advancements. ML tools like Waikato Environment for Knowledge Analysis (WEKA)¹⁵ and image analysis platforms like Fiji/ImageJ¹⁶ have been employed for segmenting XCT datasets. Fiji/ImageJ, with plugins like Trainable Weka Segmentation¹⁷, offers advanced tools for interactive image segmentation and machine learningbased pixel classification. For instance¹⁸, utilized the Trainable WEKA Segmentation (TWS) plugin in Fiji/ImageJ to perform pore segmentation in XCT and Serial Sectioning (SS) images. Ground truth labels were manually created to train a supervised ML classifier based on the Random Forest algorithm. Similarly¹⁹, applied the TWS plugin in Fiji/ImageJ to manually generate ground truth labels for XCT images and conducted supervised training using a Fast Random Forest classifier. This approach was employed to establish a correlation between the identified pores and strain localization in the built material. Both studies highlighted that interactive image segmentation methods, such as WEKA, require supervised training to ensure consistency in segmentation results for XCT or similar optical imaging. This process necessitates the generation of ground truth labels, manually for cases where the labels are absent. Consequently, there is a pressing need for a reliable, consistent, computationally efficient, and automated approach, free from operator bias, to accurately characterize defects in metal additive manufacturing (AM)¹⁸. Another notable example is the use of U-Net²⁰ for defect segmentation in LAM. Originally developed for medical image

¹Industrial and Management Systems Engineering, West Virginia University, Morgantown, 26505 WV, USA. ²School of Mathematical and Data Sciences, West Virginia University, Morgantown, 26505 WV, USA. Virginia University, Morgantown, 26505 WV, USA.



Fig. 1 | Comparison of the supervised technique redrawn from²⁰ and the proposed model in terms of label dependency and training requirements.

segmentation, U-Net has become a standard for detecting defects across various industrial applications such as surface anomalies²¹, internal porosity²² and 3D volumetric defects²³. Alternative methods have also been explored, such as the Hessian matrix-based analysis in²⁴ for pore identification and autonomous defect detection techniques using Support Vector Machines (SVMs) and neural networks²⁵. Similarly²⁶, combined Principal Component Analysis (PCA) and K-means clustering to analyze XCT images. Traditional ML methods, including SVMs and Random Forests, often use thresholding techniques like Bernsen²⁷ and Otsu²⁸, though these are increasingly replaced by deep learning models. For example²⁹, used Convolutional Neural Networks (CNNs) trained on micro-CT X-ray images to estimate the properties of porous media. Similarly³⁰, combined Otsu thresholding with CNNs to segment porosity in metallic additive manufacturing (AM) specimens. Likewise³¹, utilized Otsu-based ground truth for U-Net training, and²³ applied a U-Net with a ResNet backbone and Bernsen thresholding for volumetric porosity segmentation. Although deep learning-based methods have advanced segmentation accuracy, they often require large labeled datasets and extensive computational resources. Transfer learning³² has been explored as a way to reduce the need for large training datasets by adapting pre-trained models, yet it still depends on partial training with labeled data²², which may not always be feasible in realworld manufacturing settings. Moreover, most LAM datasets are either unlabeled or have limited sample sizes, making them incompatible with conventional supervised training methods such as CNNs and their variants. Furthermore, unlike standardized medical imaging datasets^{33,34}, porosity in L-PBF is highly variable and often inadequately labeled, making it more challenging to analyze and segment. These challenges highlight the need for more adaptable and versatile segmentation techniques.

In this paper, we focus on the segmentation of pores in cylindrical parts produced using various setups of the PBF-L process, as detailed in⁵. To address the challenges of porosity segmentation, we develop an unsupervised framework based on the Segment Anything Model (SAM)³⁵, a Foundation Model created by Meta for image segmentation tasks³⁶. It has a pre-trained image encoder, a prompt encoder, and a light-weight mask decoder. SAM has been recognized for its transformative impact on image segmentation by utilizing user-specified prompts, such as points or bounding boxes, to guide the segmentation process on new datasets without requiring prior training^{37,38}. It has demonstrated strong performance in medical imaging³⁹⁻⁴¹, camouflaged object detection⁴², object tracking⁴³, pseudo label generation^{44,45}, and image captioning⁴⁶. However, SAM faces challenges in real-world tasks without fine-tuning⁴¹, particularly in additive manufacturing^{44,47}. Moreover, SAM's reliance on manually generated prompts limits its applicability in scenarios where annotations are unavailable, particularly in industrial settings where real-time human intervention is impractical. In our analysis, we find that applying SAM directly to XCT images does not yield optimal results for the porosity segmentation

task in our case. Thus, we propose a novel visual prompt generation scheme to address these limitations, applicable to datasets similar to those used in this case study. This method allows SAM to effectively segment porosity in unlabeled XCT images of PBF-L specimens, eliminating the need for manual inputs. We summarize our contributions in this paper as follows:

- We minimize the dependency on labels and supervised training of conventional ML models by incorporating a foundation model for image segmentation, SAM in our proposed framework.
- We develop a novel porosity segmentation framework for PBF-L parts by introducing an unsupervised clustering-based method to generate contextual prompts, guiding SAM to effectively segment porosity in unlabeled XCT datasets.
- We utilize bootstrapping to quantify the uncertainty of the model's performance across various prompt sets, providing a robust evaluation of the proposed framework's reliability and accuracy.

Our research reveals that integrating contextual prompt generation into the SAM framework provides weak supervision⁴⁸, significantly improving performance and final segmentation quality. Our approach avoids the need for supervised training or fine-tuning, which requires large, labeled datasets as demonstrated in Fig. 1. As demonstrated later, our unsupervised prompt generation achieves high-accuracy predictions and simplifies the challenges of manual prompt annotation and label creation for our unlabeled dataset.

Results

Our proposed framework in Fig. 2 consists of three primary stages: data preprocessing, prompt generation, and performing inference using SAM. These steps are discussed in detail followed by the segmentation results and performance evaluation of the framework in the following subsections.

Data Pre-processing

Our framework is designed to consider the layer-by-layer printing approach of the PBF-L process and we treat the data collected as the function of the layers in order as the XCT images are collected along the build direction according to⁵. The morphology and characteristics of pores remain relatively consistent within the same sample as described in⁵. While porosity is inherent to the material and influenced by process parameters, we use consecutive slices of XCT images along the build direction to identify similarities among the images. This allows us to determine the central tendency of pore locations within a group of images, enabling efficient prompt generation for SAM. This approach aligns with the layer-by-layer nature of LAM, making it suitable for real-time sequential data processing if needed. Taking the above considerations into account, we opt to use partitional clustering techniques⁴⁹ which can provide a mean or center point accurately representing the central tendency of the clusters at different layers. Thus, the goal at this stage of the framework is to identify a



Fig. 2 | Our proposed framework of unsupervised prompt generation with Segment Anything Model.

Table 1	Mean Dice Sin	nilarity Score o	obtained from th	e different promp	ot generation tec	hniques on all the sar	nples

Samples	Dice Similarity Score				
	K-means	K-medoids	Reference binary masks	No Prompts	
Sample 3	0.62 ± 0.034	0.62 ± 0.036	0.64 ± 0.029	0.0019 ± 0.0006	
Sample 4	0.44 ± 0.12	0.44 ± 0.13	0.49 ± 0.11	0.0299 ± 0.0081	
Sample 5	0.88 ± 0.019	0.88 ± 0.015	0.88 ± 0.016	0.0022 ± 0.0021	
Sample 6	0.70 ± 0.092	0.75 ± 0.042	0.79 ± 0.098	0.0002 ± 0.0002	

The standard deviations are calculated to show the spread of the Dice Similarity Scores, indicating the variability of the results obtained from different prompt generation techniques across all samples. The first two columns show the DSC scores achieved using centroids derived from different clustering techniques respectively. The next column represents the DSC scores obtained from prompts generated from the respective reference binary masks of the individual image. The final column displays the DSC scores obtained without using any prompts.

representative image that serves as the basis for generating prompts to effectively guide the segmentation task.

First, we select a set of images from each of the PBF-L samples discussed in the Dataset section and perform unsupervised clustering. The goal is to determine the centroid images representing individual clusters of XCT images, which will be utilized later to generate prompts for SAM. In this work, we use two Partitional Clustering⁴⁹ techniques namely K-means clustering⁵⁰ and K-medoids⁵¹ clustering to determine the centroid images of the clusters. This is due to their unique methods of forming clusters and the ability of K-medoids to use different distance metrics for clustering⁵². K-means is a widely used unsupervised clustering method due to its simplicity yet effectiveness in diverse data instances, but it is sensitive to outliers. On the other hand, K-medoids clustering, which employs an actual data point as the centroid of a given cluster, is less susceptible to noise or outliers. Additionally, the method relies on the selection of an appropriate distance metric which directly influences the clustering outcome⁵¹. These assumptions align well with our application, as the use of actual data points (medoids) ensures that the representative images are realistic and interpretable. However, it may overlook subtle yet significant cluster information from the rest of the data. Given their unique characteristics, we employ both K-means and K-medoids clustering in our experiments for prompt generation. The purpose of using both K-medoids and K-means is to evaluate the effectiveness of our unsupervised prompt generation methodology by comparing centroids and medoids as representative images. We find later in Table 1 that the final segmentation results are in the same range using the representative images from both the clustering techniques. It is worth noting that K-medoids clustering has higher computational complexity compared to K-means, primarily due to the repeated evaluation of dissimilarities between all points and

nedoids) ensures that le. However, it may in the rest of the data. eans and K-medoids 'he purpose of using iveness of our unsu-

Prompt Generation

the ex-situ characterization of internal porosity, it is typical for data to be unlabeled, which creates difficulties for the prompt-tuning of pre-trained models such as SAM. Therefore, we focus on using unsupervised techniques for creating visual prompts to assist the defect segmentation of the unlabeled XCT images using the SAM inference model. In our study, we investigate the utilization of point prompts in SAM and use the location coordinates of the pores that make up the foreground. The steps are described below.

potential medoids, resulting in an overall complexity of $O(n^2 \times k)$ per iteration,

whereas K-means, with its reliance on mean calculations, has a lower com-

plexity of $O(n \times k)$ per iteration⁵³. Despite this added computational cost, it is

not a significant concern in our analysis because the dataset size for each

sample remains manageable. To ensure fairness in comparison, we maintain equal-sized selected sets for all samples. While applying K-medoids clustering

on the XCT images, we utilize both Euclidean distance⁵⁴ and Dynamic Time

Warping (DTW) distance⁵⁵ as distinct metrics for effective comparison of the

resulting medoids. Notably, the medoid images obtained from K-medoids

Prompts can be defined as the creation of a task-specific information tem-

plate. In natural language processing (NLP), prompting transforms the

downstream dataset into a language modeling problem⁵⁶. This approach

allows a pre-trained language model to adapt to a new task without modifying

clustering are identical when using either distance metric.

As described before in the data preprocessing step, we first select a set of images from a PBF-L printed sample, run unsupervised clustering i.e., Kmeans, and K-medoids, to group them in respective clusters, and obtain the centroid or the medoid images as the representative image for that cluster. For ease of reference, we call both the centroids and medoids as 'Centroids' in the rest of the paper given that they serve the same purpose in our methodology. We then employ Binary Thresholding⁵⁷ to partition each centroid image into the foreground representing defects, and background encompassing the remaining data. Following this we randomly select a small subset of 2D location coordinates from the foreground pixels of these stored centroid images. These selected coordinates are labeled as class '1' to comply with SAM's prompt encoder requirement for foreground labeling. Finally, we apply them as prompt inputs to SAM during the inference. Generating prompts using centroids of clustered images as a representative source avoids the overhead of creating prompts from each individual image on which segmentation is performed. The pseudocode for our Prompt Generation algorithm is demonstrated as Algorithm 1.

Algorithm 1. Prompt Generation

Input: Centroid Image *C_k*; **Output:** 2D point coordinates *p*, Labels *l*;

Steps:

Select the respective centroid images C_k from cluster k; Apply the Binary Thresholding to C_k ; Collect p = (x, y);

Create labels $l = \{1\}^n$ for (x, y), where n = |x| = |y|; end

Inference Using SAM

In this stage, the clustered images and their respective prompts are provided to the SAM inference model to generate binary masks as the output. Since a single input prompt can correspond to multiple valid masks, SAM provides three valid final masks simultaneously instead of one to avoid ambiguity i.e., Subpart, Part, and Whole object, covering different sections of the respective image as shown in Fig. 2. Sub-part masks typically contain comparatively less information about the pores than part masks. On the other hand, masks segmenting the entire circular surface of the samples as the whole object are not useful in our case. For porosity segmentation, we observe that masks identified as parts are the most suitable for our application and achieve the highest DSC score (refer to the '*Performance Analysis using Centroid-Based Prompts*' section for how DSC scores are calculated) among the three masks.

SAM returns an estimate of the Intersection over Union (IoU)⁵⁸ score along with every predicted mask based on the granularity of the segmentation. These IoU scores are predicted by SAM based on its pertaining (not the current dataset), demonstrating confidence in segmenting different parts of the same object. These IoU scores are not computed by comparing the masks with the ground truth of the current dataset being used. Since we opt for an end-to-end unsupervised protocol in our framework, we select the most appropriate masks based on their ranks by the estimated IoU scores from SAM as ground truth may be not available in many cases. The three masks representing various parts of the object are derived from different object boundaries set by SAM itself. It is important to clarify that SAM predicts IoU scores as estimations for the different masks it generates for the same prompt and object. The masks are consistently indexed in the order of their ranks determined by SAM's predicted IoU. Typically, sub-part masks exhibit the lowest IoU scores (< 0.5) due to the complexity of segmenting small, intricate regions. This results in higher boundary errors but captures finer details of smaller objects. Part masks exhibit moderate IoU scores (<0.8) balancing granularity and segmentation complexity while whole object masks tend to have the highest IoU scores (0.88 - 0.9) among the three masks because SAM is more confident in segmenting larger and coarse regions. However, in our case, these masks represent the overall structure of the object and do not capture the internal pores of interest. We find that for all the samples of our dataset, the predicted IoU for any mask returning the whole object is usually higher than 0.88-0.90 and remains stable across all the samples. Thus we use it as a limit of the estimated IoU score to select the corresponding mask as shown in Algorithm 2. The strategy is to always select the mask that identifies the parts, unless its IoU score surpasses the limit, indicating that it also returns the entire object. In such cases, the sub-part mask is selected as the final predicted result. For instance, in our study for Sample 4, the sub-part mask is most effective at capturing the finer granularity of very small pores (< 0.06 mm) as shown in Fig. 3. In contrast, for samples with larger pore sizes (e.g., Samples 3, 5, and 6 in Fig. 3, where pore sizes are > 0.5 mm), the part masks are more accurate and relevant. Whole object masks, on the other hand, represent the broader structure and are less suitable for segmenting pores. Please note that one can obtain the multimask outputs and select the desired mask by visual inspection in case an endto-end unsupervised methodology is not required. The pseudocode for our porosity segmentation framework using SAM with centroid-based prompt generation through unsupervised clustering is referred to as Algorithm 2.

Algorithm 2. Unsupervised Porosity Segmentation Framework

Input: XCT images I;

Output: Binary Masks M_i , Predicted IoU score S_i ;

Steps:

```
Run unsupervised clustering to images I to group them into k clusters;
Select the respective centroid images C_k for cluster k;
Generate point prompts p using Algorithm 1;
Input the images I_k from cluster k and Point Prompts p to SAM;
Run SAM inference and obtain M_i, S_i where 0 \le i \le 2;
for each M_i, S_i do
Select M_1, S_1 (part);
if S_1 > 0.88 then
| Select M_0, S_0 (subpart);
end
end
```

Prompt Bootstrapping

As outlined in Algorithm 1, prompts for a given image are generated by randomly selecting a subset of points, specifically the 2D location coordinates of the foreground corresponding to the centroid image. To assess the variability of this process and its impact on the final segmentation results, we employ bootstrapping⁵⁹ of the prompts. Bootstrapping is a resampling technique used to estimate the sampling distribution of a statistic or to generate prediction intervals^{60,61}. However, in this study, we do not aim to estimate prediction intervals but rather to verify the stability of the segmentation process using a subset of point prompts corresponding to foreground or Class '1' from a centroid image. To achieve this, we apply *m* out of *n* bootstrapping⁶², where the point prompts are sampled with replacement. Here, *n* represents the total number of points for Class '1', and *m* < *n* is the number of points resampled.

For each bootstrap iteration, we perform m out of n bootstrapping on the 2D location coordinates of the foreground of the respective centroid image for the particular cluster of images. The total number of foreground points (n) varies across samples, ranging from 1,400 to 300,000. We set m =10, 000 for all samples, except for Sample 4, where m = 1, 000 due to the significantly smaller sample size. In each bootstrap iteration, a new set of prompts is randomly selected with replacement, meaning that previously selected points remain eligible for selection in subsequent iterations. For each of the m resampled sets of prompts, we run our framework using the SAM inference model on the clustered images and compute the Dice Similarity Coefficient (DSC) scores (see the next section) from the predicted masks. This process is repeated for B = 100 bootstrap iterations, generating a distribution of segmentation scores (DSC) for each image.

Afterward, we calculate 95% confidence intervals for the segmentation results from Samples 3, 4, 5, and 6, providing an estimate of the variability in our results (see Table 3).



Fig. 3 | Prediction results by our framework. Rows 1, 2, 3, and 4 represent examples of XCT images from all the samples. (**a**.1 - **a**.4) Input XCT Image from Sample 3, 4, 5 and 6 respectively, (**b**.1 - **b**.4) Predicted mask by our framework using K-means-based prompts for Sample 3, 4, 5 and 6 respectively, (**c**.1 - **c**.4) Predicted mask by our

framework using K-medoids-based prompts Sample 3, 4, 5 and 6 respectively, (**d**.1 - **d**.4) Predicted mask by our framework using corresponding reference binary Masks-based prompts for Sample 3, 4, 5 and 6 respectively, (**e**.1 - **e**.4) Corresponding reference binary mask Sample 3, 4, 5 and 6 respectively.

Algorithm 3. Bootstrapping Prompts

Input: Clustered XCT images I_k , Centroid image C_k ; **Output:** DSC scores, D_k^i ;

Steps:

First, choose the size of prompts, m, where $m \ll n$ where n = |(X, Y)|; for each cluster k do Collect (X, Y) from C_k [Algorithm 1]; for sample i = 1 to B do Randomly sample p_i , $(|p_i| = m)$; Run SAM inference model on I_k and p_i ; Calculate DSC scores for predicted masks, D_k^i ; Store DSC scores, D_k^i ; end Compute means and confidence interval of DSC scores, D_k^i ;

end

Performance Analysis using Centroid-Based Prompts

Since no ground truth is available for this dataset, we generate 'Reference binary masks' from the original images only to evaluate the performance of the proposed framework. To accomplish this, we adopt a few-step thresholding technique similar to the one employed by²² on the same dataset to generate labels. Thresholding is a common method used in image processing to create binary masks, particularly when ground truth labels are unavailable, as it allows for segmentation based on pixel intensity or other features. We select a set of 50 consecutive images representing approximately six consecutive layers, as each layer is captured by about eight images. We select the number of images considering computational constraints while preserving the original resolution of each image to ensure no loss of information. These images are then divided into three clusters using unsupervised methods such as K-means and K-medoids. This clustering approach effectively groups images from consecutive layers together, capturing the nuances and similarities in pore distribution across the different

	K-means prompts		K-medoids prompts		Reference binary masks prompts	
	Precision	Recall	Precision	Recall	Precision	Recall
Sample 3	0.86 ± 0.0007	0.48 ± 0.0016	0.86 ± 0.008	0.48 ± 0.0013	0.86 ± 0.0003	0.48 ± 0.001
Sample 4	0.29 ± 0.011	0.87 ± 0.01	0.23 ± 0.013	0.81 ± 0.007	0.34 ± 0.016	0.95 ± 0.017
Sample 5	0.85 ± 0.001	0.94 ± 0.0001	0.84 ± 0.0009	0.93 ± 0.0001	0.84 ± 0.0009	0.93 ± 0.0001
Sample 6	0.79 ± 0.001	0.63 ± 0.01	0.82 ± 0.0003	0.68 ± 0.01	0.84 ± 0.0016	0.75 ± 0.01

Table 2 | Comparison of Precision and Recall scores among K-means, K-medoids, and Reference binary masks

layers. Each cluster contains images based on the similarities in pore distribution and their centroid images are retrieved to represent the central tendency of the pores in respective clusters for contextual prompt generation. This approach can be repeated over the images across the volume of the PBF-L samples. An NVIDIA A10 GPU (Graphics Processing Unit) with 23 GB VRAM (Video Random Access Memory) is used as the computing system. PyTorch version 2.5.1, integrated with CUDA 12.1, is utilized in this study alongside Scikit-learn and essential Python libraries such as NumPy and OpenCV. Each reference binary mask is treated as a surrogate ground truth for its corresponding image. These masks are then used to assess the accuracy of the segmentation results obtained from our centroid-based prompt generation technique. To measure this accuracy, we calculate the DSC⁶³, which compares the spatial overlap between the reference binary masks and the predicted masks generated by our framework. The DSC quantifies the similarity between two binary masks, as shown in equation (1). A DSC score ranges from 0 to 1, where a score of 1 indicates perfect overlap and a score of 0 indicates no overlap between the masks.

$$DSC(M,R) = \frac{2 \times |M \cap R|}{|M| + |R|} \tag{1}$$

Here, M = Predicted mask by our framework, R= Reference binary mask obtained by Thresholding, |M| and |R| are the sizes (cardinalities) of sets M and R, and $|M \cap R|$ is the number of elements common to both sets M and R respectively. We demonstrate the comparison among the DSC scores obtained from the prediction by SAM with different prompt generation techniques in Table 1. Here, in addition to our centroid-based prompt generation approach, we also evaluate prompt generation using the reference binary masks for each individual image and explore the scenario where no prompts are provided to the SAM. It is to be noted in the case of column 3, that the prompts used in our approach are a small subset of the foreground information of the pores derived from each reference binary mask for each image, not the mask itself. The purpose of column 3 is to evaluate whether using prompts from each reference binary mask specific to an individual image produces significantly different segmentation results compared to prompts derived from representative centroid images. Our results show that the scores are within a very similar range, which supports the effectiveness of our centroid-based approach. We would like to clarify that column 3 represents a hypothetical scenario, as ground truth data for similar datasets like ours are usually unavailable in practical applications.

We also calculate the Precision and Recall scores⁶⁴ to evaluate the pixel classification highlighting trade-offs between false positives, false negatives, and overall balance. Precision evaluates the proportion of correctly identified positive predictions, while recall measures the model's ability to detect all actual positive instances.

$$Precision = \frac{|\mathrm{TP}|}{|\mathrm{TP}| + |\mathrm{FP}|}$$
(2)

$$Recall = \frac{|\mathrm{TP}|}{|\mathrm{TP}| + |\mathrm{FN}|} \tag{3}$$

Here in equation (2) and (3), TP (True Positives) = The number of correctly predicted positive instances (e.g., correctly segmented defect pixels), FP

(False Positives) = The number of instances incorrectly predicted as positive (e.g., non-defect pixels mistakenly classified as defect pixels). FN (False Negatives) = The number of actual positive instances that were incorrectly predicted as negative (e.g., defect pixels missed by the model).

Table 2 compares the Precision and Recall scores achieved using different prompt generation techniques with SAM predictions.

Without any prompts, SAM returns ambiguous defect masks, leading to low DSC values, indicating poor segmentation performance in Table 1. Our experiments show that a prompt size of at least 10,000 pixels for defect regions leads to the highest DSC scores across most samples. However, the prompt size needs to be adjusted based on the number of pixels corresponding to defects. For instance, Sample 4 contains significantly fewer defect pixels, therefore, the prompt size is reduced to around 1,000 points. This adjustment helps to align the prompt size with the available defect data, ensuring the framework can better focus on relevant areas.

Therefore, we utilize a prompt size of 10,000 points of location coordinates of total pixels belonging to the defect regions for Samples 3, 5, and 6, and 1000 points for Sample 4 respectively. Our framework demonstrates effective performance in segmenting pores within Sample 5, characterized by moderate pore dimensions and uniform pore distribution demonstrated in Fig. 3 and Table 4. It also achieves the best balance, with both high precision ($\simeq 0.86$) and very high recall ($\simeq 0.93$), reflecting robust segmentation and a low false negative rate. The framework provides slightly reduced yet acceptable results in the case of Sample 6, featuring irregularly shaped and large-to-medium-sized pores, and Sample 3, exhibiting a moderately intricate distribution of smaller-sized pores i.e. equivalent spherical diameter of 0.01 mm to 0.5 mm as shown in both Fig. 3 and Table 4. In Sample 6, the model has moderate precision ($\simeq 0.800$) and recall ($\simeq 0.68$) suggesting it has missed a few true defects. In Sample 3, the results demonstrate high precision ($\simeq 0.8628$) but low recall ($\simeq 0.48$), indicating it reliably predicted defects but missed many true positives. This could be attributed to the wide variation in pore sizes within this sample, ranging from as small as 0.01 mm to a moderate size of 0.5 mm. However, its performance is less satisfactory in the case of Sample 4 with low precision ($\simeq 0.2972$) and high recall ($\simeq 0.8794$), resulting in a low mean DSC score of 0.44 due to increased false positives. The high number of false positives could be attributed to the small proportion of pores (2.07%) from Table 4 and their minimal sizes, with equivalent spherical diameters ranging from 0.008 mm to 0.06 mm (Fig. 3). Additionally, a subset of pixel values within non-defective regions closely resemble those associated with porous regions. This arises from the XCT imaging of densely solid, defect-free regions, further complicating segmentation challenges.

We observe that the prediction accuracies achieved using both K-means and K-medoids clustering methods, as well as prompts generated from individual reference binary masks for every image are quite similar across all samples. Although reference binary mask-based prompts yield slightly higher accurate results due to their precise location information for each image, this approach is not practical in real-world applications. In typical settings, LAM data is unlikely to come with pre-annotated labels, and generating separate reference binary masks for each image makes this method unsuitable for real-time segmentation tasks. In contrast, our approach, which leverages centroid-based prompts generated from unsupervised clustering, offers a more practical and efficient solution. It generalizes well across multiple images and significantly reduces the time Table 3 | Mean length and standard deviation of the 95% confidence intervals calculated from the bootstrapped distributions of DSC scores, using quantiles, across all samples

Samples	Mean length
Sample 3	0.0103 ± 0.0025
Sample 4	0.0396 ± 0.0236
Sample 5	0.0017 ± 0.0009
Sample 6	0.0153 ± 0.0034

required for prompt generation, while still delivering comparable accuracy, as shown in Table 1. The best predictions obtained for each of the samples using our framework are demonstrated in Fig. 3. This efficiency makes our framework more suitable for large-scale and real-time applications.

Performance Analysis using Bootstrapping

To estimate the variability of our segmentation results using a random set of coordinates from the centroids as prompts, we utilize Algorithm 3, which involves the application of bootstrapping the prompts for all four samples. To do this, we use centroids from K-means clustering. The results from K-medoids clustering are expected to be on similar lines.

We run our framework on the set of images selected across all four samples as demonstrated in Algorithm 3 i.e., we run our framework on each of the images from each of the clusters from every sample 100 times to perform prompt bootstrapping and compute the DSC scores. Using the distributions estimated through bootstrapping the prompts, we generate 95% confidence intervals based on the quantiles⁶⁵ of the estimated distribution of the DSC scores obtained from each image from every iteration. We then calculate the mean lengths of these intervals across all images from the four sets of samples. The results are shown in Table 3. Using the quantiles (2.5th and 97.5th percentiles) of the bootstrapped distributions, the length of each 95% confidence interval is determined by calculating the difference between the upper and lower quantiles. The mean length represents the average of these interval lengths across all bootstrap iterations for each image, while the standard deviation quantifies the variability in the lengths of the confidence intervals. This approach effectively captures both the central tendency (mean length) and the spread (standard deviation) of the confidence interval lengths, offering insights into the consistency and variability of the segmentation process. These mean lengths provide a quantitative measure of the segmentation variability and help in evaluating the robustness of the process across samples of the data.

A smaller mean length with a small standard deviation suggests lower variability in the DSC scores, indicating that the segmentation process is more stable and reliable across different randomly sampled prompts. Conversely, a larger mean length with a large standard deviation, such as in Sample 4, indicates higher variability, reflecting a less consistent segmentation performance for that sample. Sample 5 has the smallest mean length and standard deviation compared to the other samples, indicating that our framework provides consistently reliable segmentation results for this sample across multiple independent runs. Sample 3 and Sample 6 have slightly higher mean lengths, suggesting some variability in the segmentation results and indicating that the framework faces minor challenges with these samples. In summary, we observe that except for Sample 4, the intervals are relatively tight thereby indicating a low variability and the model's outcome is consistent in individual images across all the samples by using centroid-based randomly sampled prompts. This step aims to verify that the proposed framework consistently delivers reliable outcomes across samples during independent runs.

The 95% confidence intervals of the DSC scores for each of the images for all the samples are shown in Figs. 4 and 5 respectively. The purpose is to visualize the consistency of the performance of the proposed framework for multiple independent runs on the same image from each of the samples. Sample 5 demonstrates the narrowest intervals in Fig. 5, highlighting the



Fig. 4 | The 95% confidence intervals, based on the quantile of the distribution of DSC scores for the predicted masks generated by our framework, calculated for both Sample 3 and Sample 4 through prompt bootstrapping.

most consistent and reliable segmentation results across independent runs. Sample 3 in Fig. 4 and Sample 6 in Fig. 5 show slightly wider intervals, indicating moderate variability but generally stable segmentation performance for the same image. In contrast, Sample 4 in Fig. 4 exhibits the widest intervals, reflecting significant variability and challenges in achieving consistent segmentation results.

Discussion

The accuracy of semantic segmentation is influenced by the number, size, and distribution of instances within an image. This becomes particularly challenging when multiple similar objects are present, making it difficult to differentiate between them. According to⁶⁶, segmentation tends to be easier with datasets containing fewer instances, whereas more diverse datasets with broader distributions introduce added complexity. Datasets with heavy-tailed distributions pose even greater challenges due to their increased diversity. In our case study, we visualize the distribution of pore counts per image using connected components⁶⁷ in Fig. 6.

Our proposed framework shows promising results for samples with low to moderate instances i.e. pore counts and relatively uniform distributions, as demonstrated by Sample 5, which achieves a high DSC score of (0.88 ± 0.09) in Table 1. However, the framework faces challenges with



Fig. 5 | The 95% confidence intervals, based on the quantile of the distribution of DSC scores for the predicted masks generated by our framework, calculated for both Sample 5 and Sample 6 through prompt bootstrapping.

datasets exhibiting heavy-tailed distributions, such as Sample 6, and those with higher instance counts, like Sample 3 (Table 1). For Sample 3, in addition to the high pore count and complex spatial distribution, we also observe the presence of small trapped powder particles within some pores. These particles are marked as non-defect backgrounds in the reference binary masks during thresholding. This leads to misclassifications, or segmentation errors, as their pixel intensities closely resemble those of the nondefect solid surface (Fig. 3). In the case of Sample 6, our framework exhibits limited success by segmenting the non-defect regions instead of the pores. This outcome may be attributed to the unique pore morphology observed in this sample, characterized by large size, irregular shapes, and interconnectedness. It is important to highlight that, for this particular sample, we have generated the reference binary masks based on non-defective regions rather than exclusively focusing on the defect areas. This methodology has been adopted to ensure a comprehensive evaluation of the accuracy of our framework.

One limitation of our framework is observed in the context of Sample 4, where the segmentation outcomes exhibit significant fluctuations. This is evidenced by the higher standard deviation, indicating greater variability, as shown in Table 1. Although the pore count is low for this particular sample, the variability can be attributed to the presence of variations and noise within the background pixels at a granular level specific to this sample. Additionally, several pixels belonging to the non-defect area have a similar pixel intensity as the pores, contributing to the observed fluctuations. However, while our model effectively segments the pores within the solid regions, it occasionally misclassifies background artifacts as pores in a few images, causing the fall of the DSC score. Furthermore, we note that, for Sample 4, the most accurate results are obtained from the masks segmenting the sub-parts, from the multi-mask outputs of SAM. This can be attributed to the limited number of pores with very small sizes i.e. equivalent spherical diameter less than 0.06 mm to segment in this sample, resulting in parts and objects being indistinguishable. Overall, our framework has demonstrated excellent segmentation performance with Sample 5, and moderate performance with Samples 3 and 6 respectively while encountering challenges with Sample 4. Previous studies²² have also reported similar challenges, noting that pre-trained U-Net models struggle with background noise in Sample 4. These findings highlight the necessity of enhancing our unsupervised prompt generation approach with a prompt optimization strategy, either in the original or embedded space, to improve the efficiency of prompt utilization in our framework.

This framework is not intended to compete with existing supervised approaches for fully labeled datasets, which have already shown high performances. For example, the recent work by²² reported an average IoU score exceeding 95% across all samples using a fully supervised learning approach. Their method relied on custom label creation to train a Random Forest and a pre-trained U-Net, resizing images to 256 × 256 pixels to suit the model architecture. While this led to impressive segmentation accuracy, such an approach has practical drawbacks. Generating ground truth labels manually can be challenging and time-consuming, and resizing can lead to a loss of critical information, particularly in high-resolution datasets like XCT scans. Moreover, their results showed a significant drop in IoU scores for out-ofdistribution data, with notable examples being Sample 3 at 35%, Sample 4 at 43%, and Sample 6 at 75%, underscoring the limitations of supervised models when dealing with unseen variations in data. These findings emphasize the challenge of generalizing beyond the training dataset. In contrast, our approach avoids the need for fine-tuning or label generation by deriving contextual prompts directly from the data to guide SAM in segmentation, achieving comparable results without labeled data (see Table 1). To our knowledge, this is the first unsupervised approach for porosity segmentation in unlabeled XCT data using SAM, offering a solution for cases where labeled data is scarce or unavailable. Our results provide a viable alternative to existing supervised techniques for effective segmentation in challenging, label-constrained environments.

Accurate detection and localization of porosity are essential for ex-situ characterization in PBF-L. While XCT imaging provides detailed insights into internal artifacts in 3D-printed parts, identifying pores remains challenging due to their variability. Although deep learning-based porosity segmentation has advanced significantly in recent years, traditional supervised models are hindered by the scarcity of labeled data and poor performance during domain shifts. Additionally, these models often demand substantial data and computational resources. To address these limitations, in this paper, we introduce a novel porosity segmentation framework leveraging the Segment-Anything Foundation Model. The major findings of our work are summarized below:

- I. We successfully minimize the computational overhead associated with supervised training or fine-tuning, as well as the need for labeled data in real-world manufacturing scenarios, by incorporating the Segment Anything Foundation Model and by introducing a novel prompt generation scheme. Instead of generating labels or prompts for individual pores in each image, our framework utilizes representative images from unsupervised clustering of XCT images to create contextual prompts. As a result, we achieve a DSC score exceeding 80% during inference, without having performed any supervised training or fine-tuning.
- II. Using centroid images from both K-means and K-medoids clustering as representatives for each data group yielded similar results to



prompts derived from individual ground truth (reference binary masks). This highlights the effectiveness and generalizability of our approach.

- III. Our framework performs well on samples with a moderate pore count and a uniform pore distribution, in contrast to those with a heavytailed pore distribution or complex pore morphology. Additionally, we observe consistent performance across different sets of prompts for the same sample, as confirmed by uncertainty quantification through bootstrapping.
- IV. While the current methodology has been applied to a case study of exsitu characterization of porosity from XCT images of PBF-L printed samples, the framework holds potential for other LAM processes such as Directed Energy Deposition. As long as internal defects like porosity are captured by imaging techniques, our approach can be adapted to detect and segment pores in various LAM processes and materials. This makes it a flexible tool for porosity segmentation across different manufacturing conditions.

Our study focuses on porosity segmentation using XCT imaging of samples produced with different process parameters, laying the groundwork for advancing the use of promptable Foundation Models, such as SAM, in the additive manufacturing domain. However, the relatively large number of multi-point prompts required in our case may benefit from optimization within the embedding space, as suggested by⁶⁸, potentially through few-shot training. Moving forward, we plan to explore the selection of representative images, analyze the embedded space, and refine prompt generation to enable prompt optimization. Additionally, we aim to extend our work to other datasets within the LAM domain including the XCT imaging of airmaterial boundary, pores near the surface, and XCT images with artifacts such as beam-hardening and dead pixels exploring embedded space representations and integrating robust noise mitigation. In summary, our centroid-based prompt generation method for defect segmentation, coupled with SAM's zero-shot generalization capability as mentioned in³⁵, offers a simple and efficient framework that reduces human involvement and holds promise for real-time anomaly detection⁶⁹ in LAM.

Methodology

In this section, we briefly describe the dataset and discuss the key components of the Foundation model used in this study i.e., the Segment Anything Model (SAM).

Dataset

The data presented in this study is derived from⁵, where Cobalt-chrome (CoCr) alloy specimens were fabricated using a EOS M270 Direct Metal Laser Sintering System (DMLS) via the Powder Bed Fusion-Laser (PBF-L) process. High-resolution XCT scans were conducted on the CoCr

Fig. 7 | XCT images showing the surface topology of defects of four CoCr alloy cylindrical discs; from left to right respectively from⁵.



 Table 4 | Porosity information of the XCT images of the samples at varying process parameters

Samples	lmage size (px)	Scanning Speed (mm/s)	Hatch space (mm)	Porosity (%)
Sample 3	(988, 1013)	3200	0.1	18.2
Sample 4	(984, 1010)	800	0.2	2.07
Sample 5	(984, 1010)	800	0.4	10.19
Sample 6	(984, 1013)	3200	0.4	72.0

specimens to examine how processing parameters, specifically scan speed and hatch spacing, affect pore structure formation and overall porosity. The XCT scans were performed using a ZEISS Versa XRM500 system with a tube voltage of 155 kV and a power of 10 W, achieving \simeq 10% transmission. Cylindrical cores with a 5 mm diameter had been extracted from larger cobalt-chrome alloy disks (40 mm diameter, 10 mm height). The voxel sizes used were approximately 2.44 μ m for most samples and 0.87 μ m for highresolution scans of Samples 3 and 5. To improve beam transmission and harden the X-ray spectrum, the highest X-ray attenuation filter was applied. The images were reconstructed using a cone-beam CT reconstruction algorithm with a smoothing filter (kernel size: 0.7), and no additional preprocessing, ring removal, or beam hardening corrections were applied. It is to be noted that the XCT scans do not include the external surface in this study. The completely black air background and the lack of beam hardening in the images are consistent with this focus on internal structures. Layer thickness is nominally 20 μ m, and powder particle sizes range from 5 μ m to 80 μ m, peaking at 30 μ m. Therefore, the dataset comprises 8-bit grayscale images of 2D slices from the top surface of cylindrical 3D samples. We use data from four specimens (Samples 3, 4, 5, and 6), as defined in the original study⁵, selected based on their variations in porosity volume, size, and distribution (Fig. 7 and Table 4).

Example images of Samples 3 to 6 (Fig. 7) show notable differences in pore structures due to varying processing parameters, with pores influenced by hatch space and scanning speed according to⁵. As seen in Table 4, Sample 4 has the least porosity and smaller, closed pores without trapped powder, due to its low hatch space. Samples 3 and 5, despite having similar porosity percentages, exhibit distinct pore structures-Sample 3 shows smaller, more connected pores, while Sample 5 has larger, disconnected pores, both containing unmelted particles due to higher hatch space. Sample 6, with 72% porosity, displays disconnected solid parts, largely due to its high porosity.

Segment Anything Inference Model

The architecture of SAM consists of an image encoder, a prompt encoder, and a mask decoder (Fig. 8). The main components are discussed as follows.

Image Encoder. The image encoder takes RGB images, rescaling them to 1024×1024 with padding, and processes them using a pre-trained Masked Autoencoder (MAE)⁷⁰. It outputs a downscaled embedding of the input image, reduced by a factor of 16 in both dimensions, and

reduces the dimensionality to 256 channels at a resolution of 64×64 . Following the notations from³⁷, first, an RGB image *I* is converted into image embeddings, *e* by a convolutional operation, CONV as below where e^{i} represents each of the patch embeddings.

$$e = \text{CONV(I)}, \text{ where } I \in \mathbb{R}^{3 \times 1024 \times 1024}, e = \{e^{j} \in \mathbb{R}^{64 \times 64}, 1 \le j \le 768\}$$
(4)

Subsequently, these patch embeddings e are processed through a stack of 12 Transformer blocks. The final output embedding from the image encoder *Encoder¹* is:

$$E = \text{Encoder}^{I}(e), \text{ where } E = \{e^{q} \in \mathbb{R}^{64 \times 64}, 1 \le q \le 256\}$$
(5)

where e^q represents each of the final patch embeddings.

Prompt Encoder. The prompt encoder accommodates different types of prompts: sparse prompts (bounding boxes, points) and dense prompts (masks) embedded via convolution and element-wise summation using the image embeddings and text prompts³⁵. In our work, we use sparse prompts in the form of points to perform image segmentation. For point prompts, the representation of each point comprises a positional encoding of its location which is combined with embeddings denoting the location class of the point.

From the prompts p and labels l provided by the user, the prompt embeddings generated by the prompt encoder, $Encoder^{p}$, are:

$$[x^*, P] = \text{Encoder}^P([l, p]), \text{ where, } P \in \mathbb{R}^{256}$$
(6)

Here, x^* represents a learned embedding indicating the location class of a point which can be either the foreground or background, and is inserted along with the prompt embeddings *P*, which are termed as 'tokens' in the original SAM paper³⁵.

Mask Decoder. During inference, the mask decoder takes 256dimensional image embeddings and prompt tokens as input, producing masks through two transformer layers. The process involves selfattention on tokens, cross-attention between tokens and image embeddings, and updates via point-wise MLPs, with residual connections and layer normalization for stability. The enhanced embeddings are then upscaled using transposed convolutions, and the updated tokens pass through MLP heads to compute mask probabilities. Finally, the decoder predicts multiple masks and generates a confidence score, estimating the IoU for each mask and assessing their quality.

For the *N* number of *Layers* in a transformer block,

$$[x_i^*, E_i] = \text{Layer}_i([x_{i-1}^*, P_{i-1}, E_{i-1}]), \text{ where } 1 \le i \le N, N = 2$$
(7)

Here, x^* represents the learnable class token⁷¹, and *P* and *E* as the prompts and image embeddings respectively. The final output of the Mask Decoder



of SAM, $Decoder^{M}$ is denoted as below:

$$(M, S) = \text{Decoder}^{M}([\mathbf{x}^*, \mathbf{P}, \mathbf{E}])$$
(8)

Here, *M* denotes the final predicted mask and *S* is the predicted IoU score by the model. x^* denotes the learnable class token, *P* denotes the prompts from prompt encoder *Encoder*^{*P*} and *E* denotes the image embeddings from image encoder *Encoder*^{*I*}.

SAM is trained on the SA-1B dataset³⁵, which consists of over 1 billion masks across 11 million diverse images. A high-performance computing system with a total number of 256 A100 GPUs over approximately is used in training SAM for 68 hours for 90,000 iterations. The training setup includes Adam optimizer, a batch size of 256, and a learning rate of $8e^{-4}$ with a weight decay by a factor of 10.

Data Availability

The dataset used and analyzed during the current study is available in NIST Engineering Laboratory's repository, https://www.nist.gov/el/intelligent-systems-division-73500/cocr-am-xct-data.

Code availability

The underlying code for this study is not publicly available but can be made available on reasonable request to the corresponding authors.

Received: 1 November 2024; Accepted: 11 February 2025; Published online: 30 March 2025

References

- 1. ISO/ASTM 52900, I. Additive manufacturing–general principles– fundamentals and vocabulary (2021).
- King, W. E. et al. Laser powder bed fusion additive manufacturing of metals; physics, computational, and materials challenges. *Applied Physics Reviews* 2 (2015).
- Zhang, Y., Yang, S. & Zhao, Y. F. Manufacturability analysis of metal laser-based powder bed fusion additive manufacturing-a survey. *Int J. Adv. Manuf. Technol.* **110**, 57–78 (2020).

- Fu, Y. et al. Machine learning algorithms for defect detection in metal laser-based additive manufacturing: A review. *J. Manuf. Process* 75, 693–710 (2022).
- Kim, F., Moylan, S., Garboczi, E. & Slotwinski, J. Investigation of pore structure in cobalt chrome additively manufactured parts using x-ray computed tomography and three-dimensional image analysis. *Addit. Manuf.* 17, 23–38 (2017).
- Al-Maharma, A. Y., Patil, S. P. & Markert, B. Effects of porosity on the mechanical properties of additively manufactured components: a critical review. *Mater. Res Express* 7, 122001 (2020).
- Du Plessis, A., Yadroitsev, I., Yadroitsava, I. & Le Roux, S. G. X-ray microcomputed tomography in additive manufacturing: a review of the current technology and applications. *3D Print. Addit. Manuf.* 5, 227–247 (2018).
- 8. Davies, E. R. *Machine vision: theory, algorithms, practicalities* (Elsevier, 2004).
- Oztemel, E. & Gursev, S. Literature review of industry 4.0 and related technologies. J. Intell. Manuf. 31, 127–182 (2020).
- Park, C. & Ding, Y. Data Science for Nano Image Analysis. International Series in Operations Research & Management Science (Springer Cham, 2021).
- 11. Yan, H., Grasso, M., Paynabar, K. & Colosimo, B. M. Real-time detection of clustered events in video-imaging data with applications to additive manufacturing. *IISE Trans.* **54**, 464–480 (2022).
- Mou, S. et al. Paedid: P atch a utoencoder-based d eep i mage d ecomposition for pixel-level defective region segmentation. *IISE Trans.* 56, 917–931 (2024).
- Tabernik, D., Šela, S., Skvarč, J. & Skočaj, D. Segmentation-based deep-learning approach for surface-defect detection. *J. Intell. Manuf.* 31, 759–776 (2020).
- Yu, H. et al. Methods and datasets on semantic segmentation: A review. *Neurocomputing* **304**, 82–103 (2018).
- Witten, I. H., Frank, E., Hall, M. A., Pal, C. J. & Data, M. Practical machine learning tools and techniques. In *Data mining*, vol. 2, 403–413 (Elsevier Amsterdam, The Netherlands, 2005).
- Schindelin, J. et al. Fiji: an open-source platform for biological-image analysis. *Nat. methods* 9, 676–682 (2012).

- Arganda-Carreras, I. et al. Trainable weka segmentation: a machine learning tool for microscopy pixel classification. *Bioinformatics* 33, 2424–2426 (2017).
- Jolley, B. R., Uchic, M. D., Sparkman, D., Chapman, M. & Schwalbach, E. J. Application of serial sectioning to evaluate the performance of x-ray computed tomography for quantitative porosity measurements in additively manufactured metals. *JOM* **73**, 3230–3239 (2021).
- Miers, J. C., Moore, D. G. & Saldana, C. Modeling strain localization in laser powder bed fusion 316l stainless steel with porous defects. *Available at SSRN 4363069*.
- Ronneberger, O., Fischer, P. & Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18, 234–241 (Springer, 2015).
- Scime, L., Siddel, D., Baird, S. & Paquit, V. Layer-wise anomaly detection and classification for powder bed additive manufacturing processes: A machine-agnostic algorithm for real-time pixel-wise semantic segmentation. *Addit. Manuf.* **36**, 101453 (2020).
- Ouidadi, H., Xu, B. & Guo, S. Defect segmentation from x-ray computed tomography of laser powder bed fusion parts: A comparative study among machine learning, deep learning, and statistical image thresholding methods. In *International Manufacturing Science and Engineering Conference*, vol. 87233, V001T01A012 (American Society of Mechanical Engineers, 2023).
- Wong, V. W., Ferguson, M., Law, K., Lee, Y.-T. & Witherell, P. Automatic volumetric segmentation of additive manufacturing defects with 3d u-net. In: AAAI 2020 Spring Symposium Series. AAAI Press, Palo Alto-VIRTUAL, CA, US (2020).
- García-Moreno, A.-I., Alvarado-Orozco, J.-M., Ibarra-Medina, J. & Martínez-Franco, E. Image-based porosity classification in al-alloys by laser metal deposition using random forests. *Int J. Adv. Manuf. Technol.* **110**, 2827–2845 (2020).
- 25. Malarvel, M. & Singh, H. An autonomous technique for weld defects detection and classification using multi-class support vector machine in x-radiography image. *Optik* **231**, 166342 (2021).
- Singh, A., Rabbani, A., Regenauer-Lieb, K., Armstrong, R. T. & Mostaghimi, P. Computer vision and unsupervised machine learning for pore-scale structural analysis of fractured porous media. *Adv. Water Resour.* **147**, 103801 (2021).
- 27. Bernsen, J. Dynamic thresholding of grey-level images fcv. In Proceeding of the 8 International Conference O11 Pattern Rec-Gn Ition, 125I–1255 (1986).
- Otsu, N. et al. A threshold selection method from gray-level histograms. *Automatica* 11, 23–27 (1975).
- Alqahtani, N., Alzubaidi, F., Armstrong, R. T., Swietojanski, P. & Mostaghimi, P. Machine learning for predicting properties of porous media from 2d x-ray images. *J. Pet. Sci. Eng.* 184, 106514 (2020).
- Gobert, C. et al. Porosity segmentation in x-ray computed tomography scans of metal additively manufactured specimens with machine learning. *Addit. Manuf.* 36, 101460 (2020).
- Bellens, S., Probst, G. M., Janssens, M., Vandewalle, P. & Dewulf, W. Evaluating conventional and deep learning segmentation for fast x-ray ct porosity measurements of polymer laser sintered am parts. *Polym. Test.* **110**, 107540 (2022).
- Van Opbroek, A., Ikram, M. A., Vernooij, M. W. & De Bruijne, M. Transfer learning improves supervised image segmentation across imaging protocols. *IEEE Trans. Med imaging* 34, 1018–1030 (2014).
- Mazurowski, M. A. et al. Segment anything model for medical image analysis: an experimental study. *Med Image Anal.* 89, 102918 (2023).
- Wu, J. et al. Medical sam adapter: Adapting segment anything model for medical image segmentation. arXiv preprint arXiv:2304.12620 (2023).
- Kirillov, A. et al. Segment anything. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4015–4026 (2023).

- 36. Bommasani, R. et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258* (2021).
- 37. Jia, M. et al. Visual prompt tuning. In *European Conference on Computer Vision*, 709–727 (Springer, 2022).
- Liu, P. et al. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput Surv.* 55, 1–35 (2023).
- Shi, P. et al. Generalist vision foundation models for medical imaging: A case study of segment anything model on zero-shot medical segmentation. *Diagnostics* 13, 1947 (2023).
- 40. Zhang, K. & Liu, D. Customized segment anything model for medical image segmentation. *arXiv preprint arXiv:2304.13785* (2023).
- 41. Ma, J. et al. Segment anything in medical images. *Nat. Commun.* **15**, 654 (2024).
- 42. Tang, L., Xiao, H. & Li, B. Can sam segment anything? when sam meets camouflaged object detection. *arXiv preprint arXiv:2304.04709* (2023).
- Yang, J. et al. Track anything: Segment anything meets videos. arXiv preprint arXiv:2304.11968 (2023).
- 44. Ahmadi, M. et al. Application of segment anything model for civil infrastructure defect assessment. *arXiv preprint arXiv:2304.12600* (2023).
- Chen, J. & Bai, X. Learning to" segment anything" in thermal infrared images through knowledge distillation with a large scale dataset satir. arXiv preprint arXiv:2304.07969 (2023).
- Wang, T. et al. Caption anything: Interactive image description with diverse multimodal controls. *arXiv preprint arXiv:2305.02677* (2023).
- 47. Ji, G.-P. et al. Sam struggles in concealed scenes empirical study on "segment anything". *Sci. China Inf. Sci.* **66**, 226101 (2023).
- Zhou, Z.-H. A brief introduction to weakly supervised learning. *Natl Sci. Rev.* 5, 44–53 (2018).
- Madhulatha, T. S. Comparison between k-means and k-medoids clustering algorithms. In *International Conference on Advances in Computing and Information Technology*, 472–481 (Springer, 2011).
- 50. Ding, C. & He, X. K-means clustering via principal component analysis. In *Proceedings of the twenty-first international conference on Machine learning*, 29 (2004).
- Park, H.-S. & Jun, C.-H. A simple and fast algorithm for k-medoids clustering. *Expert Syst. Appl* 36, 3336–3341 (2009).
- Das, S., Mhaskar, H. N. & Cloninger, A. Kernel distance measures for time series, random fields and other structured data. *Front Appl Math. Stat.* 7, 787455 (2021).
- 53. Jain, A. K. Data clustering: 50 years beyond k-means. *Pattern Recognit. Lett.* **31**, 651–666 (2010).
- 54. Danielsson, P.-E. Euclidean distance mapping. *Computer Graph image Process* **14**, 227–248 (1980).
- 55. Müller, M. Dynamic time warping. *Information retrieval for music and motion* 69–84 (2007).
- Bahng, H., Jahanian, A., Sankaranarayanan, S. & Isola, P. Exploring visual prompts for adapting large-scale models. *arXiv preprint arXiv:2203.17274* (2022).
- 57. Seelaboyina, R. & Vishwakarma, R. Different thresholding techniques in image processing: A review. In *ICDSMLA 2021: Proceedings of the 3rd International Conference on Data Science, Machine Learning and Applications*, 23–29 (Springer, 2023).
- Rahman, M. A. & Wang, Y. Optimizing intersection-over-union in deep neural networks for image segmentation. In *International symposium on visual computing*, 234–244 (Springer, 2016).
- 59. Efron, B. & Tibshirani, R. J.*An introduction to the bootstrap* (Chapman and Hall/CRC, 1994).
- Efron, B. Bootstrap methods: another look at the jackknife. In Breakthroughs in statistics: Methodology and distribution, 569–593 (Springer, 1992).

- Das, S. & Politis, D. N. Predictive inference for locally stationary time series with an application to climate data. *J. Am. Stat. Assoc.* **116**, 919–934 (2021).
- Lee, S. M. S. & Pun, M. On m out of n bootstrapping for nonstandard m-estimation with nuisance parameters. *J. Am. Stat. Assoc.* **101**, 1185–1197 (2006).
- 63. Carass, A. et al. Evaluating white matter lesion segmentations with refined sørensen-dice analysis. *Sci. Rep.* **10**, 8242 (2020).
- Powers, D. M. W. Evaluation: From precision, recall and f-measure to roc, informedness, markedness & correlation. *J. Mach. Learn. Technol.* 2, 37–63 (2011).
- 65. Hoeffding, W. Probability inequalities for sums of bounded random variables. *The collected works of Wassily Hoeffding* 409–426 (1994).
- Amirkhani, D., Allili, M. S., Hebbache, L., Hammouche, N. & Lapointe, J.-F. Visual concrete bridge defect classification and detection using deep learning: A systematic review. *IEEE Transactions on Intelligent Transportation Systems* (2024).
- Rosenfeld, A. & Pfaltz, J. L. Sequential operations in digital picture processing. J. ACM (JACM) 13, 471–494 (1966).
- Huang, J. et al. Learning to prompt segment anything models. arXiv preprint arXiv:2401.04651 (2024).
- Jiang, Y., Wang, W. & Zhao, C. A machine vision-based realtime anomaly detection method for industrial products using deep learning. In 2019 Chinese Automation Congress (CAC), 4842–4847 (IEEE, 2019).
- 70. He, K. et al. Masked autoencoders are scalable vision learners 2111.06377 (2021).
- Dosovitskiy, A. et al. An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2021).

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. 2119654. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. The authors would like to acknowledge the Pacific Research Platform, NSF Project ACI-1541349, and Larry Smarr (PI, Calit2 at UCSD) for providing the computing infrastructure used in this project.

Author contributions

I.Z.E.: Data Preprocessing; Code and Methodology, Formal Analysis, Validation, Original Draft; I.A.: Conceptualization, Methodology, Discussion, Supervision; Z.L.: Conceptualization, Supervision, Domain Knowledge, Research Funding; S.D.: Conceptualization, Methodology and Statistical Analysis, Discussion, Supervision, Computational Resources. All authors contributed to manuscript editing and revision.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to Imtiaz Ahmed, Zhichao Liu or Srinjoy Das.

Reprints and permissions information is available at http://www.nature.com/reprints

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

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, 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 you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. 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/bync-nd/4.0/.

© The Author(s) 2025