



Accelerated semantic segmentation of additively manufactured metal matrix composites: Generating datasets, evaluating convolutional and transformer models, and developing the MicroSegQ+ Tool

Mutahar Safdar^{a,b}, Yi Fan Li^a, Randy El Haddad^a, Max Zimmermann^c, Gentry Wood^d, Guy Lamouche^b, Priti Wanjara^b, Yaoyao Fiona Zhao^{a,*}

^a Department of Mechanical Engineering, McGill University, Montreal, QC H3A 0C3, Canada

^b National Research Council Canada, Montreal, QC H3T 1J4, Canada

^c Fraunhofer Institute for Laser Technology ILT, Aachen 52074, Germany

^d Apollo-Clad Laser Cladding, a division of Apollo Machine and Welding Ltd., Leduc, AB T9E 0B3, Canada



ARTICLE INFO

Keywords:

Semantic segmentation
Additive manufacturing metallographs
Class imbalance
Microstructure quantification
Vision transformers
Prediction fusion
Industrial application

ABSTRACT

Data-driven applications are penetrating every aspect of Additive Manufacturing (AM) to enable efficient resolution of key existing challenges. One such application is semantic segmentation which automatically quantifies post-process structure data from graphic or 3D computer representations. In industrial settings, these models can expedite the development of new materials or processes. However, Additively Manufactured (AMed) materials offer unique challenges, which require new models to be developed and trained. These challenges include out-of-balance classes (e.g., defects), massive labeling efforts, and significant data preparation costs. Recent applications of semantic segmentation in AM have shown the potential of ensemble-based approaches at the expense of increased parameters and computational burden while ignoring the minority classes. In this work, we propose a reproducible method and develop an associated tool to rapidly segment and subsequently quantify industrial AMed metallographic images. First, two new datasets representing AMed materials are generated from extensive experimentation. Subsequently, state-of-the-art models from convolutional and self-attention categories are evaluated in their ability to segment the datasets of interest. Finally, a modular software tool is developed for industrial applications. The developed Microstructure Segmentation, Quantification, and Fusion (+) tool or MicroSegQ+ enables a weighted pairing of predictions from different semantic segmentation models. The pairing strategy helps to exploit the complimentary performance of convolutional and transformer models on multi-class metallographic images from a Metal Matrix Composite (MMC) material system. In addition to developing models on this MMC dataset, the proposed strategy is evaluated on an open-source metal AM dataset. We also evaluate the generalizability of the best-performing model on a second MMC dataset from a different industrial setting that shows an overall accuracy of 93 % without fine-tuning. The proposed tool enables rapid quantification under 1 min, unlike the existing semi-automatic approach that takes 1–2 h of segmentation effort on single bead cross-sections. The tool is wrapped into a single executable file for industrial deployment. The work also contributes to two annotated datasets from a direct energy deposition (DED) AM process. The datasets can be used to develop and validate new semantic segmentation models of AMed microstructures, whereas the functionality of the tool can be expanded upstream to include data and modeling steps, enabling a no-code pipeline for the industry.

1. Introduction

Additive Manufacturing (AM) or 3D printing is an advanced layer-

upon-layer fabrication technique where the material and part are manufactured simultaneously (Pei, 2023). Direct Energy Deposition (DED) is an AM process where a focused thermal energy source is used to

* Corresponding author.

E-mail addresses: mutahar.safdar@mail.mcgill.ca (M. Safdar), yi.fan.li@mail.mcgill.ca (Y.F. Li), randy.elhaddad@mail.mcgill.ca (R. El Haddad), max.zimmermann@ilt.fraunhofer.de (M. Zimmermann), gentry.wood@apollomachine.com (G. Wood), guy.lamouche@cnrc-nrc.gc.ca (G. Lamouche), priti.wanjara@cnrc-nrc.gc.ca (P. Wanjara), yaoyao.zhao@mcgill.ca (Y.F. Zhao).

fuse materials (e.g., wire or powder) by melting as they are being deposited (ASTM, 2022). The DED process, offering flexibility in material deposition, is well-suited for repair and surface modification applications. It is well documented that the lengthy fabrication for metal AM can lead to different anomalies and defects in the resulting parts (Zhu et al., 2021; Oleff, 2021). For instance, DED-based processes can deposit Nickel Tungsten Carbide (Ni-WC) Metal Matrix Composite (MMC) overlays to enhance wear resistance. However, the lack of process development can lead to non-uniform particle-reinforced MMCs that directly affect the wear resistance performance. Moreover, these MMCs can suffer from porosity, and as excessive porosity can render the overlays ineffective, further process optimization is needed to reach near-full density. It is, therefore, imperative to analyze and quantify the presence of reinforcement particles and porosities (Murr, 2018).

The constituents in the microstructure of materials determine their properties and hence play a key role in their performance. In the case of Ni-WC overlays, the individual constituents complement each other by reducing material wastage, while allowing reasonable plastic deformation. Quantitative microscopy can help differentiate the presence of different constituents and subsequently support decision-making during process development. The resulting characteristics (constituents' size, volume fraction, mean free path, and distribution) relate directly to wear resistance performance (Gurland, 1984). Moreover, these analysis techniques hold considerable significance within the data-driven AM community for the provision of labels or targets to support the development of supervision-based Machine Learning (ML) models – a theme that has become a hot research topic in recent years (Zhang, 2022).

Additively Manufactured (AMed) microstructures can be analyzed and quantified through various tools and techniques. Manual approaches to quantify and distinguish different constituents of these microstructures are commonly used in industry. Although these techniques can generate high-quality analysis results, they are highly labor-intensive and time-consuming. Computer-assisted or semi-automated methods can expedite the process, but require the same effort each time a new analysis is needed (Rose, 2022). This has inspired the use of image-processing or computer vision (CV) approaches to fully automate the process of quantifying microstructures. A spectrum of such methodologies exists, ranging from simple to advanced in terms of the complexity of the underlying algorithms. While useful, CV approaches suffer from a lack of generalization and are unable to handle inherent variability introduced in AMed metallographs.

Semantic segmentation or dense prediction is a technique for pixel-wise classification of images and has become popular in several domains, including autonomous driving, medical imaging, robotics, and object recognition (Guo, 2018). The conventional ML models popular for semantic segmentation tasks are based on the well-known Convolutional Neural Network (CNN) architectures. These semantic segmentation models (e.g., U-Nets) have been proven to detect non-overlapping regions in the images through pixel-level grouping (Guo, 2018; Ronneberger et al., 2015). Even though U-Net models have become ubiquitous, the ML community has recently extended state-of-the-art vision transformers (ViTs) to semantic segmentation tasks leading to models with competitive performance (Thisanke, 2023). This highlights the continuous need to improve accuracy, develop domain-specific models, increase generalization, and generate high-quality annotated datasets.

Although convolution-based U-Net and self-attention guided ViTs perform well on a range of semantic segmentation tasks, these models suffer from domain-specific limitations and challenges (Thisanke, 2023; Hao et al., 2020). Most of the challenges in materials characterization also hold relevance for semantic segmentation of AMed microstructures. First, AM can result in complex microstructures that are difficult to contextualize and segment. Moreover, labeling or annotating the complex microstructures of AMed materials from scratch is challenging and requires significant initial effort (Rose, 2022). The AM-based anomalies and defects in the microstructures are particularly interesting during

materials development but pose an additional segmentation challenge due to their sparse presence, especially in near-full density deposits (Zhang, 2022; Safdar, 2023).

This work aims to tackle the challenge of rapidly quantifying industrial AMed metallographs while solving class imbalance and overcoming the lack of annotated multi-class MMC datasets. We start by generating two AMed MMC datasets. In addition to generating data and evaluating models, this work is also aimed at supporting the AM industry's need to rapidly segment and quantify microstructures to support the development of new processes and materials. To this end, we have developed "MicroSegQ+", a tool for microstructure segmentation, quantification, and fusion to support operators, technicians, process engineers, and materials scientists in analyzing the microstructure through segmentation and quantification in industry. The overall development pipeline is reproducible. The long-term vision is a no-code reproducible expert system expanding to data preparation and model fine-tuning.

This work contributes to:

- Two new multi-class annotated datasets from DED-based AMed metallographs
- An extension of ViTs through the Segformer model to AMed metallographic segmentation tasks
- A pairing scheme between CNNs and ViTs to overcome the class imbalance in sparse datasets
- A reproducible modeling pipeline and a modular software tool for rapid segmentation and quantification of AMed metallographs in industry

The rest of the paper is arranged as follows. Section 2 provides background information on relevant research on semantic segmentation, both in general and within the AM domain. Section 3 presents the overall framework proposed in this work. Section 4 introduces the generated AM datasets and provides details on their characteristics, preparation, processing, and usage. Section 5 details the evaluated models providing a self-contained introduction. It also contains the details of semantic segmentation modeling experiments, their results, and the relevant discussions. Section 6 demonstrates the tool developed to support technicians for rapid microstructure quantification. Section 7 concludes the work and lists avenues for future research directions.

2. Background

This section is divided into two parts. The first part briefly covers the research and evolution of ML-based semantic segmentation algorithms, starting with the fully convolutional and basic U-Net models and ending with the latest research trend of developing transformer variants. The second part highlights the applications of semantic segmentation in AM, leading to recent trends and existing gaps in research.

2.1. ML-based semantic segmentation

The ability to segment knowledge in images at the pixel level can be encoded in ML models through different mechanisms. Fully Convolutional Networks (FCNs) represent the first major transition to ML-based semantic segmentation models (Long et al., 2015). FCNs employed regular CNN architecture to accomplish the semantic segmentation task. The fully connected layers were replaced entirely with the convolutional layers to enable end-to-end segmentation on images of arbitrary size. The top-performing and representative architecture in this category was proposed by Long et al. (Long et al., 2015). The skip architecture helped recover the lost information in the down sampling process. At the time of its release, their architecture showed impressive performance on the PASCAL VOC dataset (Hoiem et al., 2009), beating state-of-the-art.

The next era of semantic segmentation models started with the encoder-decoder architectures from the U-Net series. These models

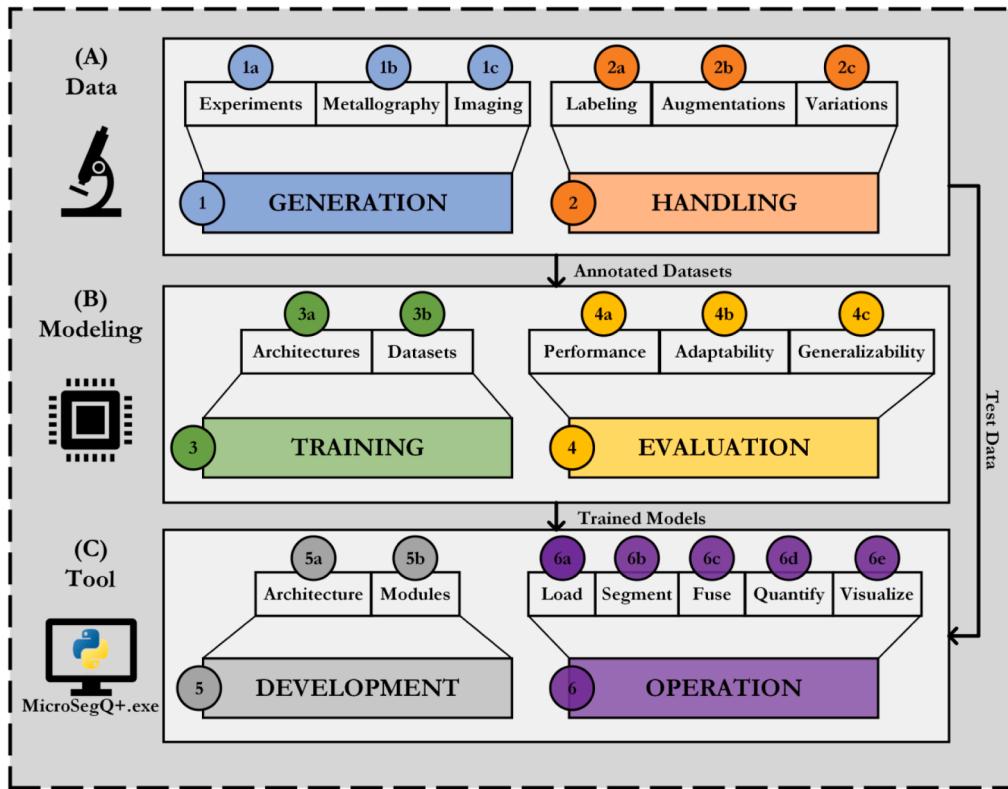


Fig. 1. The overall framework to develop *MicroSegQ+*. Data, modeling, and tool represent the major steps. Each step is divided into sub-steps that are grouped under dataset generation, dataset handling, model training, model evaluation, application development, and application operation. The sub-steps are numbered and referenced in the text.

focused on preserving the fine-grain information using skip connections. Ronneberger et al. (Ronneberger et al., 2015) gave the representative architecture of this series. The encoder portion was aimed at the contraction which was meant to capture the context of pixels, while the decoder portion was aimed at the expansion which was meant for precise localization. The architecture also relied on skip connections between the contractive and the expansive parts. These skip connections helped the model in the accurate segmentation of the pixels. While the model aimed to segment medical images, it became the de facto segmentation model in many other application areas and remains a popular choice.

Sultana et al. (Sultana et al., 2020) reviewed and summarized the semantic segmentation models based on convolutions before the advent of Vision Transformers (ViTs) and their use for image segmentation tasks. Their taxonomy of CNN-based semantic segmentation models provides five major architecture categories namely (i) networks with fully convolutional layers, (ii) networks with dilated/atrous convolutions, (iii) networks with top-down/bottom-up approach, (iv) networks based on global context and (v) networks based on receptive field enlargement and multi-scale context incorporation. Similarly, the representative architectures are (i) FCN, (ii) Dialatednet and DeepLab, (iii) Deconvnet, U-Net, SegNet, and FC-DenseNet, (iv) ParseNet, GCN, and EncNet, (v) DeepLabV2, DeepLabV3, PSPNet, and Gated-SCNN. Their work provides a broader summarization of the field. Interested readers are referred to their work for a detailed comparison of the architectures along with the strengths and weaknesses of each category.

The advent of transformers and their applications to graphic data has led to their use in image semantic segmentation tasks. Transformer models use self-attention mechanisms in their architectures, allowing them to process the data in parallel and to capture long-term dependencies. This key attention feature enabled the transformer models to focus on relevant information. Dosovitskiy et al. (Dosovitskiy, et al.,

2010) extended the self-attention mechanism to images by treating an image as a sequence of tokens. This breakthrough in applying transformers to vision data has gained traction in recent years. The first application of ViTs to semantic segmentation was reported in the work of Zheng et al. (Zheng, 2021). Segmentation Transformer or SETR model replaced the encoder part of the architecture with a pure transformer where the spatial resolution of the input was gradually reduced. Since then, different researchers have extended transformers to CV tasks, including semantic segmentation. Thisanke et al. (Thisanke, 2023) reviewed and compared major ViT architectures for semantic segmentation tasks where interested readers can get insights into the evolving applications.

2.2. Applications in Additive Manufacturing

Semantic segmentation applications in AM are primarily constrained to in-process and post-process datasets. In-process applications concern the rapid detection of defects and anomalies during the process where the pixels of interest are segmented. Scime et al. (Scime, 2020) developed a CNN-based semantic segmentation architecture to segment surface visible anomalies from camera images across powder bed fusion machines of different process technologies. Some recent applications of semantic segmentation have also focused on monitoring anomalies in the powder bed (Schmitt, 2023). Other in-process applications include segmenting images of different printing conditions in fused filament fabrication (Jin, 2021) and segmenting regions of interest from in-situ sensing modalities (Zhang, 2022). A recent application of semantic segmentation to detect in-situ defects focussed on the issue of imbalanced datasets. Wang proposed and developed a class-aware semantic contrast and attention amalgamation model for semantic segmentation (Wang, 2023). The network was shown to perform well for in-situ defect detection in data imbalance scenarios.

Similar to the in-process applications, semantic segmentation techniques are becoming popular for post-process AM datasets as these can enable rapid segmentation of the features of interest. Scott et al. (Scott et al., 2023) fused Scanning Electron Microscopy (SEM) images with synthetic thermal tomography images in a shared U-Net encoder. The network was used both in the regular fashion to segment the defects as well as to classify their parameters by extracting the encoded features for a downstream fully connected network. As a result, their work led to the improvement in segmenting thermal tomography results by incorporating the images from SEM. This multi-task learning approach performed better than existing conventional techniques, but the issues of data scarcity and class imbalance were not addressed. Rose et al. (Rose, 2022) developed a convolution-based semantic segmentation model to automate segmenting NiCrBSi-WC MMC metallographic images. Their model was trained to segment carbide particles, whereas the matrix was treated as the background. While the studied material in this previous work is similar to the datasets in the current research, their work is constrained to binary segmentation. Hence, it doesn't cover the class imbalance aspect in AMed metallographs.

Two recent works on the segmentation of AMed metallographic images have proposed ensemble approaches to meet the challenge of segmenting multi-phase scenarios. Luengo et al. (Luengo, 2022) presented extensive experimentation of CNN-based architectures on their MetalDAM dataset that was publicly released as part of the study. The work also proposed a new ensemble model specialized for semantic segmentation tasks. Their stacking-based semantic segmentation ensemble approach outperformed single models. Similar work from Biswas et al. (Biswas, 2023) proposed an ensemble of three dilated attention-guided U-Net models. The prediction of each model was added in a pixel-wise manner to get the final mask. Both ensemble-based approaches show promising results but omit the minority and difficult-to-label precipitate class of the MetalDAM dataset from predicted targets.

3. The proposed framework

This section introduces the overall framework, which is divided into three tiers, namely Data (A), Modeling (B), and Tool (C). Each tier is further divided into steps and sub-steps. The development frameworks (data preparation, model training, app development) and the tools (e.g., support software) for each tier are explained in the respective sections while the description of each step and sub step is provided in the subsequent text of the section. The proposed framework shown in Fig. 1 can be reproduced in different industries aiming to quantify classes of interest in their metallographs.

Data (A) from light-based microscopy represents the first major component of the framework. Data generation (1) and data handling (2) are two main steps concerning this component of the framework. The scope of generation concerns the metallography (1b) procedure leading to the panoramic images (1c), whereas the needs of the ML models define the scope of preparation. We conducted extensive experiments (1a) to prepare two datasets for development and used an open-source third AM dataset for validation. The two MMC datasets prepared in-house were obtained with the exact composition of Ni-WC. However, two different preparation setups were used, leading to variations in intensity distribution with visibly different images. The methodology used to generate the datasets was based on light-based microscopes, since these can suitably highlight the features (porosities and reinforcement particles in MMCs) of interest for the relevant application during the industrial process development. The data preparation pipeline consists of labeling or annotation (2a), augmentations (2b), and variations (2c), where the variations are aimed at testing the capacity of the models to segment imbalanced classes in increasingly smaller datasets. The steps are explained in Section 4 on datasets.

ML modeling (B) represents the second component of our framework. We experimented with two types of semantic segmentation architectures (3a), namely convolution-based models and self-attention-

based models. The convolution-based models represent the well-known techniques to segment across scientific fields and application domains, whereas the self-attention models represent the state-of-the-art ViTs. The key architectures in these categories were evaluated, and the final selection represents the top-performing models for our datasets (3b). As a result, the strategy adopted is modular, and the selected model from each architecture can be first fine-tuned before their use in the tool. In this regard, the fine-tuning experiments are discussed in terms of their hyper-parameters and development settings in Section 6. Three different datasets were used in the experimentations involving training (3) and evaluation (4): an in-house dataset that was generated and annotated to test the training performance (4a), an open-source AM dataset, MetalDAM, to compare the performance of our models with state-of-the-art results (4b), and another in-house dataset with a different preparation setup to test the generalizability of our top performing model (4c).

The tool represents the final component of the framework with basic functionalities (5) to cater to the needs of the industry. MicroSegQ+ supports hosting different model types (5a) and enables the fusion of results from two different model categories. The existing modules (5b) in its architecture can be extended to include new functionalities. The usage (6) of the tool is described in detail in Section 7. It allows users to load models and images, segment images, fuse individual models' segmentation results, and quantify and visualize them (6a-6e). The results and segmented masks can be saved as well. The modular architecture can be extended to accommodate different models and data types. The details on architecture and functionality are discussed in Section 7.

4. Additive manufacturing datasets

Metallography datasets play a crucial role in materials development. By analyzing the information provided by a microstructure, one can optimize the process parameters, leading to an improvement in the production quality. In AM, an essential component of the microstructure is pixels belonging to defects (e.g., porosities) or anomalies (non-uniform particle distribution). In this regard, we work with three different industrial datasets to develop and validate the results of semantic segmentation. The first two single-bead Direct Energy Deposition AM (DEDAM) datasets are prepared by the authors whereas the third one is an open-source AM dataset (Luengo, 2022). The MMC datasets represent coated or repaired components for enhanced wear resistance applications. As such, there is a great interest in the ability to rapidly quantify the reinforcement area or fraction. The information on the WC reinforcement particles, porosity, and matrix can mainly highlight the performance of the deposited MMCs. The metal AM dataset, MetalDAM, represents AMed steel micrographs. Similar to the constituents in Ni-WC MMC, differentiating the fraction of the constituents (later referred to as classes) in steels is important for developing, controlling, and monitoring the AM process and resulting properties. In a way, steels are very much like composites with hard and soft phases that contribute (e.g., rule of mixtures) to the mechanical response and performance of the alloy. Thus, selecting MetalDAM for our analysis was opportune. More information on the MetalDAM can be found in the source (Luengo, 2022).

4.1. Preparation procedure

For the metallographic analysis of the individual beads in DEDAM-1 dataset, they are separated perpendicular to the application direction. The separated samples are then embedded in Bakelite to facilitate further metallographic preparation. Following embedding, the samples are ground successively with increasingly fine sandpaper (80 grit, 120 grit, 220 grit, 500 grit, 1200 grit) to produce a flat surface and minimize grinding marks. The Saphir 550 grinding and polishing machine, manufactured by ATM Qness GmbH, is used to prepare the samples. The machine applies a constant force of 200 N for 2 min per abrasive grain. After grinding, the samples are polished for 8 min with a force of 25 N to



Fig. 2. A sample bead cross-section (7321 by 3185) depicting a panoramic view after stitching individual cross-sectional images taken from optical microscopy in DEDAM-1 dataset at Fraunhofer Institute of Laser Technology ILT. The information on different constituents needs to be analyzed and quantified to assess the effectiveness of process parameters during development. The phases of interest in the Ni-WC MMC beads captured with optical microscopy include the background, porosity, carbide, matrix, HAZ, and substrate. For training purposes, we remove the scale bar and crop the image to exclude irregular borders leading to 7221 by 3085 cross-section for the sample shown here.

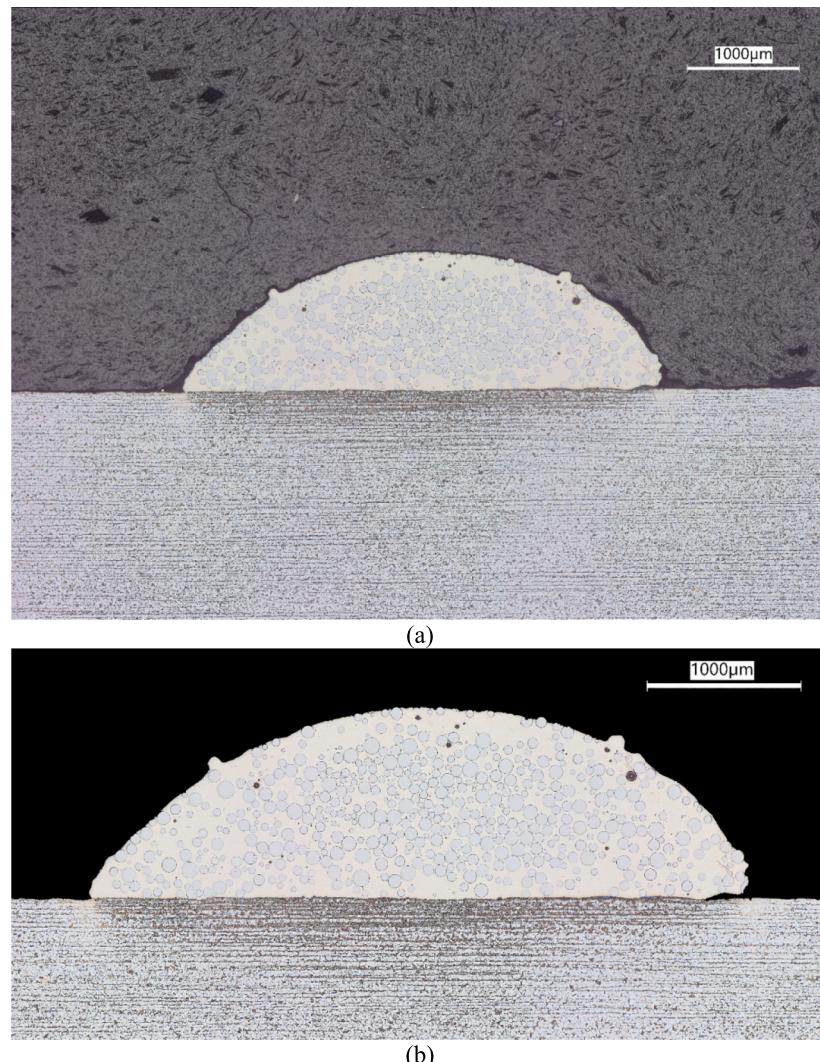


Fig. 3. (a) A sample bead cross-section depicting a panoramic view after stitching individual cross-sectional images taken from optical microscopy in the DEDAM-2 dataset at Apollo-Clad Laser Cladding. (b) Post-processing of the original sample using Adobe Photoshop to highlight features of interest. The constituents of interest remain the same as DEDAM-1 dataset presented earlier.

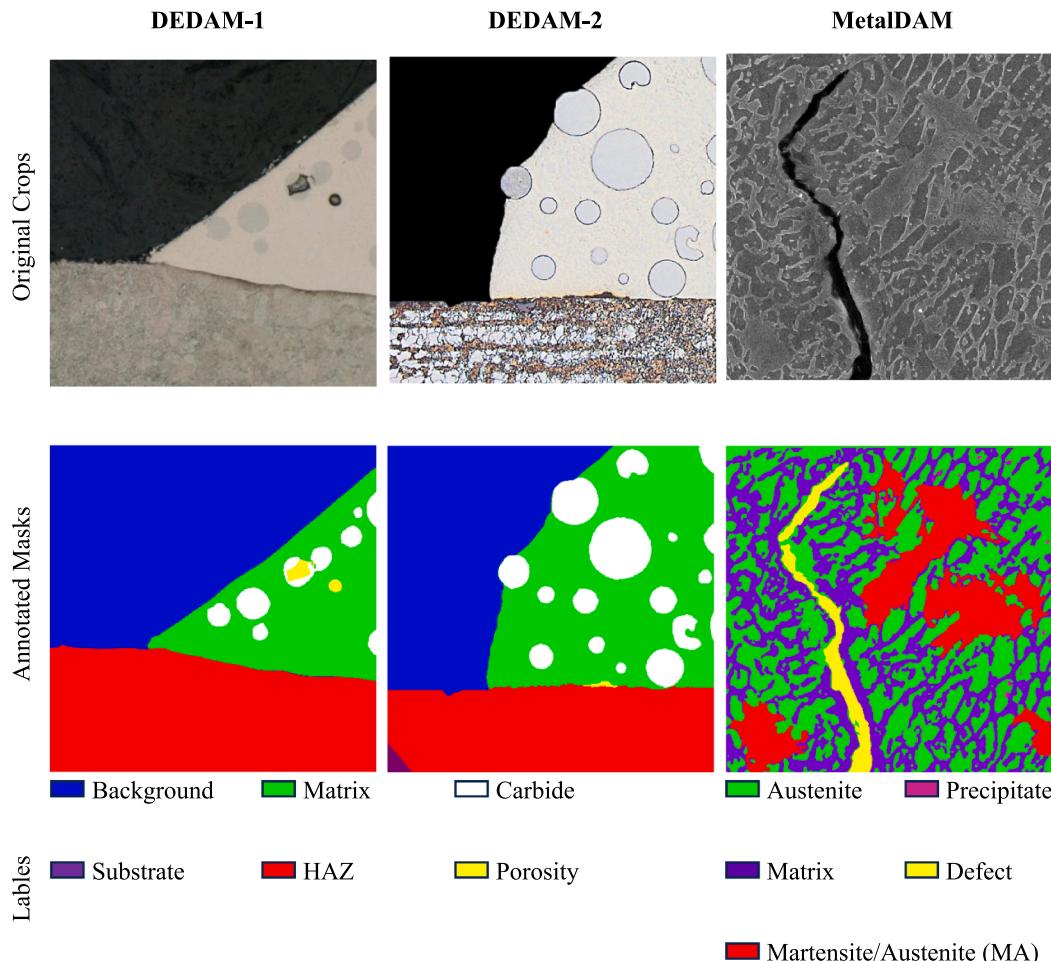


Fig. 4. 512 by 512 crop samples from the metallography datasets. Top row shows raw micrographs from DEDAM-1, DEDAM-2, and MetalDAM datasets (left to right). Bottom row shows corresponding ground truth masks with color-coded class annotations. For Ni-WC MMC datasets, the anomalies of the preparation process such as pull-out carbides and dirt particles are ignored to simplify the training process.

remove any grinding marks. To highlight the Heat-Affected Zone (HAZ) in the cross-sections, the samples are etched with Nital (alcoholic nitric acid $\sim 3\%$) for 10 s and then cleaned with ethanol to remove any residues. After preparing the samples, they are examined under an Olympus BX53M microscope with 10x magnification. The Olympus Stream Motion software is then used to automatically capture microscopic images using an autofocus and stitching process, resulting in a resolution of $0.97\text{ }\mu\text{m}/\text{pixel}$. A typical resulting cross-section is shown in Fig. 2.

Metallographic analysis of the single beads in DEDAM-2 dataset consists of sectioning the beads transverse to the travel direction at the midpoint of the clad pass. The cross-sections are mounted in Bakelite and manually ground finer successively with resin-bonded diamond abrasive pads (80 grit, 180grit, 220grit, 500grit) to produce a flat surface using a Struers LaboPol-2 grinding and polishing machine equipped with a LaboDoser. After grinding, the samples are polished to a mirror finish using the same Stuers machine in progressively finer steps using diamond slurry ($9\text{ }\mu\text{m}$, $6\text{ }\mu\text{m}$, $3\text{ }\mu\text{m}$) followed by a finishing $0.04\text{ }\mu\text{m}$ step with an alumina suspension. The HAZ is revealed in the coupon with a 5 s Nital etch (alcoholic nitric acid $\sim 3\%$), which is subsequently cleaned with high purity ethanol. After preparation, the samples are examined under a Keyence VHX-7000 microscope at 50X magnification. The Keyence software is then used to automatically stitch the images together, creating a large panorama of the full cross-section with a resolution of $0.20\text{ }\mu\text{m}/\text{pixel}$. A typical cross-section is shown in Fig. 3.

The images are then further post-processed in Adobe Photoshop to

improve the image quality and consistency. The procedure was as follows: straighten the image relative to the fusion line between the bead and the base material, crop to an appropriate size, mask the Bakelite using a black background, increase contrast to improve the visual difference between the tungsten carbides and the surrounding nickel-based metal matrix, apply an “Unsharp Mask” filter, and place the scale bar in the corner of the image. Fig. 3b shows the results of the post-processing on the image from Fig. 3a.

4.2. Dataset characteristics

Following the discussion on their preparation, we present the datasets used for the semantic segmentation of the metallographic images. The first DEDAM dataset, DEDAM-1, was provided by the Fraunhofer Institute of Laser Technology (ILT) based in Aachen, Germany, and was used to conduct our main experiments. This dataset was originally composed of 42 unlabeled cross-sectional images of DED-based single 1D beads in Portable Network Graphics (PNG) format, with resolutions ranging from 5548 by 1969 to 7509 by 3698. We selectively chose four images from this collection, prioritizing those with the highest resolutions and most intricate contextual details. Those images were then cropped into 512 by 512 tiles, with a 50 % overlap in both vertical and horizontal directions, resulting in a total of 1093 images available for pixel-wise annotations. The second DEDAM dataset, DEDAM-2, was provided by Apollo-Clad Laser Cladding based in Edmonton, Canada, and was used to validate and reinforce the experimental findings of our

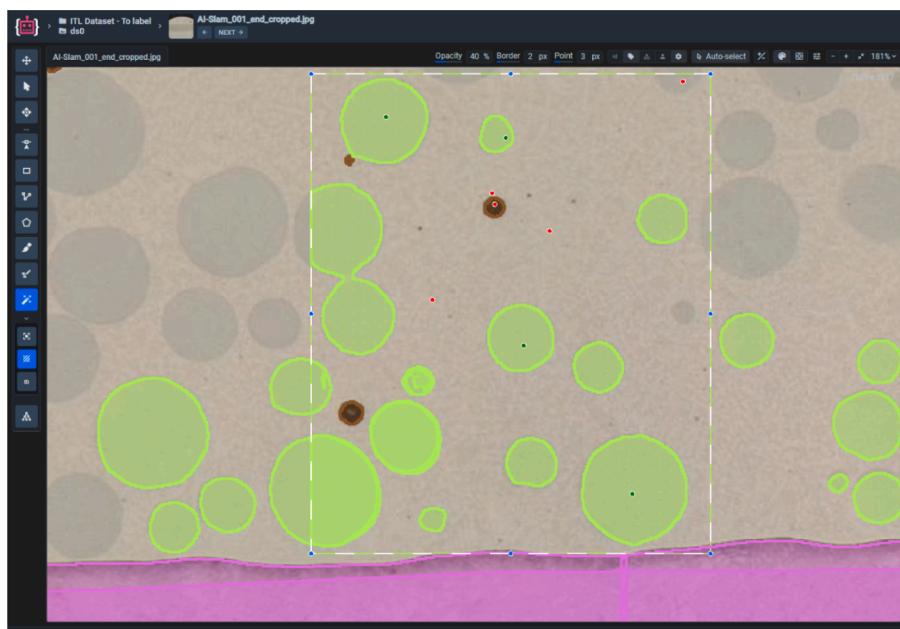


Fig. 5. Supervised interface with RITM Interactive Segmentation. Green and red dots indicate positive and negative user feedback for smart tool-assisted annotations. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1

Variations for DEDAM-1 dataset used in the experiments. Class 3 or porosities represent the minority.

Dataset Variation	Relative Size	Size of Variation	Class 0	Class 1	Class 2	Class 3	Class 4	Class 5
α	100 %	4 Images, 1093 Crops	23.15 %	30.01 %	13.39 %	0.10 %	5.41 %	27.94 %
β	47 %	2 Images, 513 Crops	20.66 %	32.29 %	14.11 %	0.06 %	5.58 %	27.29 %
γ	33 %	1 Image, 364 Crops	29.04 %	25.69 %	15.89 %	0.12 %	6.06 %	23.20 %
δ	20 %	1 Image, 216 Crops	19.13 %	31.85 %	7.48 %	0.14 %	3.93 %	37.48 %
ε	6 %	16 Images (partial), 66 Crops	14.58 %	27.20 %	24.82 %	0.31 %	11.66 %	21.42 %

models. Following a selection and cropping process similar to DEDAM-1, we selected three cross-sectional bead images and ended up with 609 cropped images for analysis. Fig. 4 displays the resultant crops alongside their pixelwise annotations.

An open-source third dataset was used to test the semantic segmentation models with a focus on minority classes and was developed by ArcelorMittal Global R&D (MetalDAM) (Luengo, 2022). This Metallography Dataset for AM or shortly MetalDAM represents micrographs of Steel and has five classes, namely matrix (class 0), austenite (class 1), martensite/austenite (class 2), precipitate (class 3) and defects (class 4). The original dataset contained 42 greyscale SEM images in Joint Photographic Expert Group (JPEG) format with two resolutions namely 1280 by 895 and 1024 by 703. Readers may consult the work of Luengo and co-workers, who introduced the dataset and provided details on its preparation and usage (Luengo, 2022). Like DEDAM datasets, the original MetalDAM dataset was partitioned into 20 labeled 512 by 512 crops for the test set and 136 crops of the same dimensions for the training set.

4.3. Labeling

Although there are a few publicly available labeled datasets, as reviewed and compared in (Luengo, 2022), they contain less than a hundred labeled images. This scarcity is due to the labor-intensive and time-consuming nature of manual annotations. We addressed the latter by using a semi-automatic approach: click-based interactive segmentation. In interactive segmentation, the algorithm is guided by human

input to recognize and delineate a target region within an image (Sofiiuk et al., 2021). We used the Revising Iterative Training with Mask (RITM) Interactive Segmentation tool integrated into the Supervisely Image Annotator Version 6.8.22 (Liu, 2022). Supervisely is a web-based platform designed to provide tools for collaborative annotation, data management, and training of ML models, particularly for CV applications. As seen in Fig. 5, the labeling of WC particles is facilitated by the user's placement of positive feedback points (green dots), which indicate the target areas for segmentation, while negative feedback points (red dots) are used to refine the selection by signaling regions to exclude. This interactive method not only saves time but also ensures precision and consistency in the annotation process.

To assist readers in applying this semi-automatic segmentation approach to their datasets, we outline the main steps and considerations from our labeling process:

- **Set-up:** team creation and image uploading
- **Defining Labeling Order (larger to smaller class):** background, substrate, HAZ, matrix, carbides, and porosities
- **Adapting Smart Labeling Strategies:** Mask pen tool and bounding box functionalities
- **Exporting Masks:** choosing version 2.0.6 for smooth exporting

4.4. Dataset size variations

The AMed metallographic datasets for semantic segmentation are

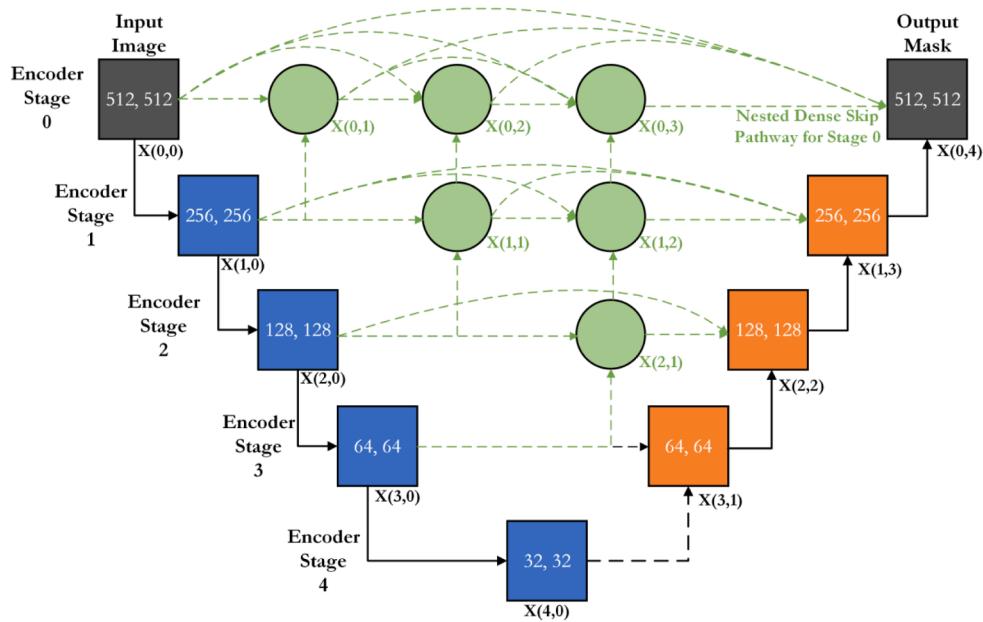


Fig. 6. The modified U-Net architecture (e.g., U-Net++) from (Zhou, 2018) with the encoder (blue), decoder (orange) and the added nested dense skip pathways (green). The added convolution blocks between the connected feature maps of the encoder and the decoder are meant to minimize the semantic gap in the original U-Net. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

sparingly labeled and rarely made open-source. To our knowledge, our annotated dataset is the largest open-source dataset for AMed metallographic images. We experimented with different dataset size variations to judge the impact on training effort and performance. Table 1 lists these size variations, assigning a Greek alphabet to each. The same naming scheme is used throughout this work. The number of crops and the resulting class proportions for each variation are also highlighted.

The variations derived from the original dataset are curated to reflect realistic scenarios typically seen in practical applications. In essence, we investigate the optimal balance between reduced labeling effort and the maintenance of high segmentation quality. Dataset α serves as the baseline, representing the DEDAM-1 dataset. The crops in this dataset are derived from four cross-sectional images, capturing both the starting point and the end point of beads A & B, with the images named "A_start", "A_end", "B_start", and "B_end". Dataset β comprises nearly half of the original dataset, including both "A_start" and "B_end". This subset is notable for its substantially lower porosity, intensifying the challenge of class imbalance. It is considered the high-quantity, low-quality variation. Dataset γ includes image "B_start", while dataset δ comprises image "A_start". Their class distribution closely aligns with the original dataset, primarily focusing on the impact of varying quantities, whilst maintaining similar quality. Lastly, dataset ϵ presents a compilation of carefully picked high-context crops from over 16 distinct beads, marked by a porosity rate nearly triple that of the original dataset. It stands as the low-quantity, high-quality variant. Detailed experimentation and analysis of these dataset variations are presented in the experiments section.

The dataset used to validate our models is also augmented using traditional augmentation techniques. Mirroring the augmentations used on the validation dataset MetalDAM in the work of (Luengo, 2022), the original dataset's size is expanded using techniques such as horizontal flip, vertical flip, and both horizontal and vertical flip combined. This workflow enabled a direct comparison with previous segmentation works involving MetalDAM. Additionally, we also applied scaling and brightness-based augmentations to MetalDAM.

5. Model evaluation

This section describes the ML models used in the learning

experiments for AMed metallographs. The choice of the models and the key features of their architectures are explained.

5.1. Convolution-based models

The convolution-based models are added in the experiments since these are known to perform well both on AM-based semantic segmentation tasks and the general image segmentation applications (Qin, 2022). The selected models represent improved architectures from the dominant encoder-decoder and atrous convolutions categories. Specifically, an enhanced U-Net and an enhanced DeepLab model are selected.

The enhanced U-Net model named U-Net++ improves the fusion of feature maps between the encoder backbone and the upsampling decoder through nested dense skip pathways (Zhou, 2018). This arrangement is inspired by the need to minimize the semantic gap between the feature maps of the encoder and the decoder in the original U-Net architecture. The nested dense skip pathways enhance the regular skip connections using deep convolutional blocks for each skip pathway between the encoder and the decoder. Fig. 6 highlights the dense skip pathways (in green) to minimize the semantic gap in the encoder and the decoder at different stages of the U-Net++ architecture.

According to the original paper on U-Net++ (Zhou, 2018), the first pathway using dense skip connections can be analyzed to explain the enhancement of fusion between the encoder and the decoder. $X(i,j)$ represents the convolution throughout the network (decoder, encoder, and the skip connections), with i indexing the downsampling layer along the encoder and j indexing the convolution layer of the dense block along the skip pathway. The skip pathway can be formulated using $x(i,j)$ as the output of node $X(i,j)$. Accordingly, the resulting feature maps at any node $x(i,j)$ along the nested dense skip pathway can be generalized.

$$x^{i,j} = \begin{cases} H(x^{i-1,j}), j = 0 \\ H\left(\left[x^{i,k}\right]_{k=0}^{j-1}, U(x^{i+1,j-1})\right), j > 0 \end{cases} \quad (1)$$

In Equation (1), the function $H(\cdot)$ represents a convolution operation applied after the activation, whereas $U(\cdot)$ denotes an upsampling layer, and braces $\{\}$ denote the concatenation operation. In the downsampling

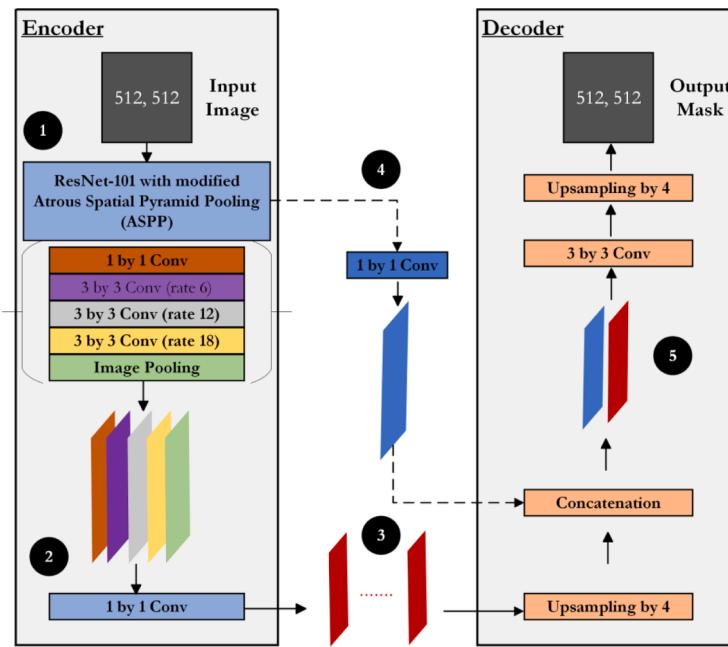


Fig. 7. The modified DeepLabv3 architecture (e.g., DeepLabv3+) from (Chen, 2018). The encoder is borrowed from the DeepLabv3 architecture (Chen, et al., 2017) and applies an augmented atrous spatial pyramid pooling (1) at varying rates with image level features (2, 3). The proposed decoder upsamples the encoded output by 4 and concatenates it with the low-level image features (4,5). The concatenated features are convolved, upsampled, and converted into predicted masks.

path of the encoder (e.g., $j = 0$), each node receives only one input, which comes from the previous node. In the first convolution layer (e.g., $j = 1$), each node receives two inputs, one from the feature map being upsampled and one directly from the previous node of the encoder, and so on. The convolution, upsampling, and concatenation operations of the nested dense skip pathway of Stage 0 can be represented by Equations (2) to (5).

$$x^{0,1} = H[x^{0,0}, U(x^{1,0})] \quad (2)$$

$$x^{0,2} = H[x^{0,0}, x^{0,1}, U(x^{1,1})] \quad (3)$$

$$x^{0,3} = H[x^{0,0}, x^{0,1}, x^{0,2}, U(x^{1,2})] \quad (4)$$

$$x^{0,4} = H[x^{0,0}, x^{0,1}, x^{0,2}, x^{0,3}, U(x^{1,3})] \quad (5)$$

The enhanced DeepLab model named DeepLabv3+ improves the atrous (also known as dilated) convolution based on DeepLabv3 by adding a simple decoder (Chen, 2018). The dilated or atrous convolution operation adapted for the architectures in the DeepLab series is well-known in general (Chen, 2017). For a 2D input feature map x , the atrous convolution operation can be defined as in Equation (6).

$$y(i) = \sum_k x[i + r.k]w[k] \quad (6)$$

where y represents a 2D output feature map, i a singular output location, w a convolution filter, and k its dimension. The key parameter controlling the convolution operation is atrous rate r leading to different fields of view. The DeepLabv3 paper provides visual examples of a 3D atrous filter with different rates and interested readers may consult it for more details (Chen, et al., 2017). The improvement is inspired by the need to refine the segmentation results around the objects' boundaries. DeepLabv3 model applies atrous or dilated convolutions in parallel at varying rates and later fuses these feature maps with image-level features to enhance the results through global context. The application of dilated convolutions at multiple scales helps to capture the context of multi-scale features in the input images. However, the model struggles with the segmentation of object boundaries. The decoder portion in the

encoder-decoder architecture gradually upsamples the features and hence has the capacity to learn sharper object boundaries. As a result, the DeepLabv3+ model brings a simple decoder module for refining the segmentation results. Fig. 7 shows the added decoder module in a base DeepLabv3 architecture.

5.2. Self-attention-based models

In this section, we describe the image segmentation models based on the transformer architecture used in the experiments. While CNNs have remained the most common architectures for state-of-the-art models in CV tasks, transformers have most recently had a breakthrough with the ViT model (Dosovitskiy, et al., 2010). The ViT model has recently achieved state-of-the-art performance in image segmentation, and its structure has been adapted for image segmentation tasks. Many new semantic segmentation models based on the transformer architecture have come out since, such as SETR (Zheng, 2021), BEiT (Bao, et al., 2021), and Segmener (Strudel, 2021). However, we will focus our coverage on Segformer (Xie, 2021), the model that performed the best in our experiments.

The Segformer model has an encoder-decoder structure, with the hierarchical encoder to generate multi-scale features and a simple all Multi-Layer Perceptron (MLP) decoder to fuse these features to produce the final semantic segmentation mask. It takes an image of size $H * W * 3$, where H represents height and W represents width and subdivides it into patches of $4 * 4$ pixels. Hierarchical features are created through overlapped patch merging during the encoder phases, to reproduce a feature map F_i with a resolution:

$$R = H/2^{i+1} \times W/2^{i+1} \times C_i, \text{ where } i = [1, 2, 3, 4] \quad (7)$$

Positional information between patches is preserved using a Mix Feed Forward Network (FFN) made up of a 3 by 3 convolution and an MLP operation (Oleff, 2021). The hierarchical feature map is decoded using multiple layers of MLP to unify the channel dimension, upsample, concatenate and finally predict. The final produced segmentation mask M has a dimension of $H/4 * W/4 * N$, where N represents the number of classes. Fig. 8 highlights the proposed architecture for Segformer.

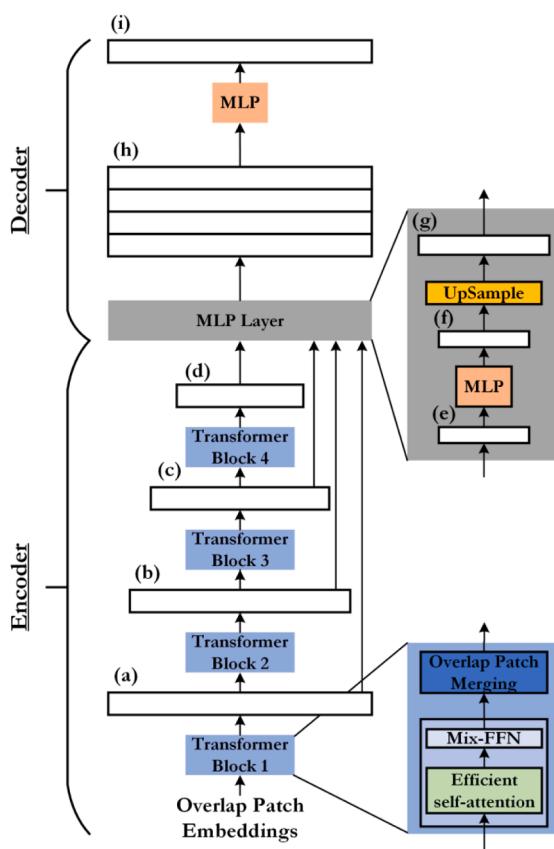


Fig. 8. Segformer architecture comprising encoder and decoder modules. The encoder (blue blocks) extracts multi-scale (coarse and fine) features whereas an all MLP decoder (grey blocks) predicts the segmentation mask directly by fusing these features. The resulting feature maps (white blocks) across the network are represented in terms of their resolution (lowercase letters in parentheses). The letters a through i represent feature maps and interested readers can explore their resulting dimensions in the source (Xie, 2021). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 2
Segformer versions and associated parameters.

Model Name	Parameters (Millions)
MiT-b0	3.7
MiT-b1	14.0
MiT-b2	25.4
MiT-b3	45.2
MiT-b4	62.6
MiT-b5	82.0

Table 3
Hyperparameters for the experiments.

	U-Net++	DeepLabv3+	Segformer B0
Input Size (Height x Width)	512 by 512	512 by 512	512 by 512
Model Size (Millions)	9.04	22	3.7
Epochs	50–200	50–200	50–200
Learning Rates	0.01–0.0001 (0.0001)	0.01–0.0001 (0.0001)	0.000006
Optimizer (B)	Adam	Adam	Adam
Batch Size	Training = 4, Validation = 1	Training = 4, Validation = 1	Training = 4, Validation = 1

Segformer models are divided into five versions, with varying parameters. The smallest version, SegFormer-B0, is made up of a total of 3.8 million parameters, and the largest version, SegFormer-B5, is made up of 84.7 million parameters. A comparison of model size is shown in Table 2.

5.3. Experimentation and results

The experiments for convolution-based models are all performed using the Segmentation Models PyTorch (SMP) library. Google Collaboratory (Colab) Pro Plus was used as the compute environment, which handles the environment variables and associated libraries. The Graphics Processing Unit chip type is not specified since it was assigned randomly. The Segformer model was implemented using the Hugging Face Transformer library from Python. It was forked from the Nvidia/mit-b0 model page, which features a lighter pre-trained Segformer encoder version and was fine-tuned on two separate datasets, namely DEDAM-1 and MetalDAM. All the models are pre-trained on the ImageNet dataset as a starting point. We also considered the possibility of training a model from scratch. We found that performance was significantly poorer than with pretraining. It required a very large number of training epochs to obtain even passable performance on the easier classes (e.g., substrate, background) of the DEDAM-1 dataset.

Table 3 specifies the parameter settings used in the fine-tuning process. All models operate on the same input size of 512 by 512 during training. The model parameters indicate that DeepLabv3+ has the largest size given a ResNet50 encoder “resent50_32x4d”. The transformer model is understandably lighter as the authors aimed to propose a simple and efficient vision transformer architecture (Xie, 2021). The range of epochs indicates that different values were used for different dataset variations, with the goal of keeping the total number of images seen in the training constant across all training. Similarly, different values of the learning rates were evaluated. The best learning rate value for each model is included in parentheses. The optimizer used for the convolutional models is Adaptive Moment Estimation (Adam). A batch size of 4 was used in the training, whereas a batch size of 1 was used for the validation. The loss function used for the Segformer model was binary cross entropy whereas dice loss function for convolutional models was used.

In order to validate the results on the lowest sized dataset, we repeated the experiments with an open-source AM dataset, namely MetalDAM. In this scenario, the MetalDAM dataset exhibits both class imbalance (for precipitates), as well as sparse labeling (only a handful of crops). DeepLabv3+ and Segformer B0 models are fine-tuned on the dataset augmented with classical augmentation methods and the results are evaluated individually. In this case, the model with the best performance (e.g., Segformer) from our experiments is also compared with state-of-the-art models from Luengo et al. (Luengo, 2022) (e.g., Semantic Segmentation Ensemble) and Biswas et al. (Biswas, 2023) (Attention-guided U-Net Ensemble) for MetalDAM dataset. Finally, a third dataset named DEDAM-2, representing the same material system from a different preparation setup is used to test the generalizability potential of models fine-tuned on DEDAM-1 dataset. For this purpose, a full bead is labeled, and the predictions are evaluated for the two models. The results of the experiments are presented in the next section.

The models are evaluated on the confusion matrix and per class intersection over union (IoU). The confusion matrix shows correctly and incorrectly predicted pixels, giving insight into what a model predicts well and where it struggles. The IoU metric (also known as Jaccard Index) is a class-informed measure, not as dependent on the number of pixels and provides an effective way to quantify the overall performance of the model without overshadowing the minority classes. Given the importance of minority classes to our primary task, we use both per class IoU and the average IoU as the metrics to compare models.

Tables 4–8 show the results in terms of IoU and F1 scores (e.g., the harmonic mean of precision and recall) for the DEDAM-1 dataset

Table 4

Mean IoU and F1 scores for variation α (original dataset) of DEDAM-1 across the convolution (U-Net++, DeepLabv3+) and transformer (Segformer B0) models.

	U-Net++		DeepLabv3+		Segformer B0	
	IoU	F1	IoU	F1	IoU	F1
Background	97.95 %	98.96 %	97.84 %	98.91 %	99.52 %	99.76 %
HAZ	83.27 %	90.87 %	91.67 %	95.65 %	87.14 %	93.13 %
Matrix	68.06 %	81.00 %	87.62 %	93.40 %	96.59 %	98.27 %
Porosity	0.00 %	0.00 %	0.01 %	0.01 %	47.35 %	64.27 %
Carbides	90.26 %	94.88 %	93.50 %	96.64 %	94.63 %	97.24 %
Substrate	82.57 %	90.46 %	94.06 %	96.94 %	90.72 %	95.13 %
Mean	70.35 %	76.03 %	77.45 %	80.26 %	85.99 %	91.30 %

Table 5

Mean IoU and F1 scores for variation β (47 % original) of DEDAM-1 across the convolution (U-Net++, DeepLabv3+) and transformer (Segformer B0) models.

	U-Net++		DeepLabv3+		Segformer B0	
	IoU	F1	IoU	F1	IoU	F1
Background	97.94 %	98.96 %	97.62 %	98.80 %	99.33 %	99.66 %
HAZ	72.71 %	84.20 %	78.25 %	87.80 %	58.29 %	73.65 %
Matrix	86.04 %	92.50 %	92.99 %	96.37 %	95.83 %	97.87 %
Porosity	1.01 %	2.00 %	0.00 %	0.00 %	33.72 %	50.43 %
Carbides	92.87 %	96.30 %	92.43 %	96.07 %	93.10 %	96.43 %
Substrate	83.24 %	90.85 %	88.79 %	94.06 %	74.53 %	85.41 %
Mean	72.30 %	77.47 %	75.01 %	78.85 %	75.80 %	83.91 %

Table 6

Mean IoU and F1 scores for variation γ (33 % original) of DEDAM-1 across the convolution (U-Net++, DeepLabv3+) and transformer (Segformer B0) models.

	U-Net++		DeepLabv3+		Segformer B0	
	IoU	F1	IoU	F1	IoU	F1
Background	97.84 %	98.91 %	97.60 %	98.78 %	99.43 %	99.71 %
HAZ	79.58 %	88.63 %	81.86 %	90.02 %	88.50 %	93.90 %
Matrix	81.31 %	89.69 %	82.88 %	90.64 %	96.47 %	98.21 %
Porosity	0.09 %	0.19 %	0.01 %	0.03 %	66.67 %	80.01 %
Carbides	94.45 %	97.15 %	92.64 %	96.18 %	93.89 %	96.85 %
Substrate	84.24 %	91.45 %	86.80 %	92.93 %	91.76 %	95.70 %
Mean	72.92 %	77.67 %	73.63 %	78.10 %	89.45 %	94.06 %

Table 7

Mean IoU and F1 scores for variation δ (20 % original) of DEDAM-1 across the convolution (U-Net++, DeepLabv3+) and transformer (Segformer B0) models.

	U-Net++		DeepLabv3+		Segformer B0	
	IoU	F1	IoU	F1	IoU	F1
Background	97.94 %	98.96 %	97.70 %	98.84 %	99.38 %	99.69 %
HAZ	84.11 %	91.37 %	87.61 %	93.40 %	89.69 %	94.56 %
Matrix	88.24 %	93.75 %	73.34 %	84.62 %	95.01 %	97.44 %
Porosity	0.65 %	1.29 %	0.00 %	0.00 %	51.23 %	67.75 %
Carbides	91.87 %	95.76 %	83.60 %	91.07 %	91.79 %	95.72 %
Substrate	92.27 %	95.98 %	88.21 %	93.74 %	92.81 %	96.27 %
Mean	75.85 %	79.52 %	71.74 %	76.94 %	86.65 %	91.91 %

Table 8

Mean IoU and F1 scores for variation ϵ (6 % original) of DEDAM-1 across the convolution (U-Net++, DeepLabv3+) and transformer (Segformer B0) models.

	U-Net++		DeepLabv3+		Segformer B0	
	IoU	F1	IoU	F1	IoU	F1
Background	97.92 %	98.95 %	97.67 %	98.82 %	99.39 %	99.70 %
HAZ	64.99 %	78.78 %	90.07 %	94.77 %	14.31 %	25.04 %
Matrix	93.82 %	96.81 %	94.80 %	97.33 %	95.55 %	97.72 %
Porosity	2.54 %	4.96 %	0.01 %	0.01 %	55.13 %	71.08 %
Carbides	93.40 %	96.59 %	91.51 %	95.57 %	93.32 %	96.55 %
Substrate	76.57 %	86.73 %	95.26 %	97.57 %	62.15 %	76.65 %
Mean	71.54 %	77.14 %	78.22 %	80.68 %	69.98 %	77.79 %

variations on the test set image. The convolutional models were given the full image as input, while the Segformer model was given 512 by 512 crops of the full image, then stitched together to obtain the full image mask. The convolutional models perform handsomely on majority classes (e.g., background, substrate or HAZ) where they beat the

transformer model on some datasets (α , β , and ϵ) as shown by the emboldened IoU. On the other hand, the transformer model performs consistently across the dataset sometimes outperforming the convolutional models on all classes (δ). More importantly, Segformer is the only model that showed capacity to learn minority class across the dataset

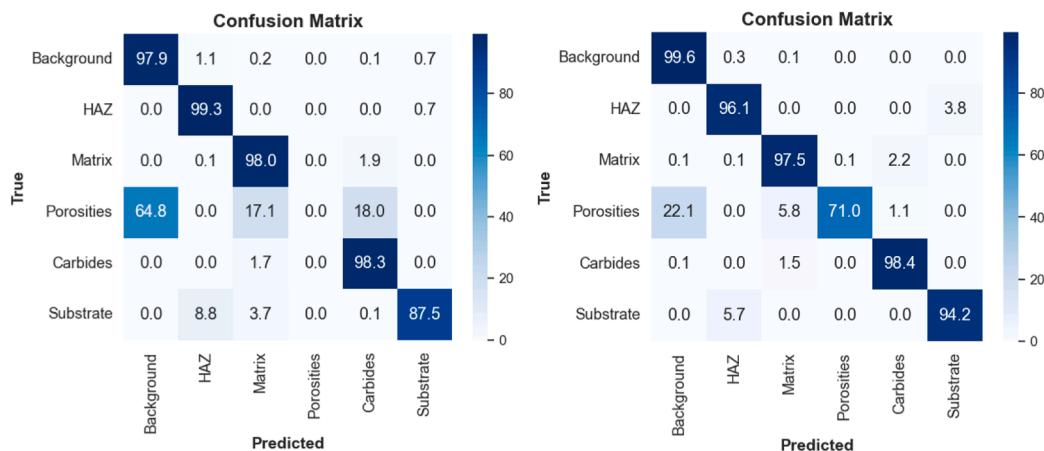


Fig. 9. Overall confusion matrix for top performing convolution model (e.g., left, DeepLabv3+) corresponding to top performing self-attention model (e.g., right, Segformer B0) on DEDAM-1 (variation δ).

Table 9

Class-wise and Mean IoU on MetalDAM dataset for two state-of-the-art models from literature and the Segformer B0 model.

	Stacking-based Semantic Segmentation Ensemble (Luengo, 2022)	Attention-guided U-Net Ensemble (Biswas, 2023)	Dominant Segformer B0 in the Weighted Pair
Matrix IoU	75.46	61.20	83.36
Austenite IoU	86.75	74.82	89.72
Martensite/ Austenite IoU	40.56	54.84	77.94
Precipitate IoU	Not Predicted	Not Predicted	33.29
Defect IoU	68.34	75.57	87.82
Mean IoU	67.78	66.61	74.43

variations, whereas convolutional models showed near-zero performance and confused it with background and neighboring classes (e.g., matrix, carbides).

The quality of the dataset also influenced the results where a

seemingly larger dataset led to low performance as compared to smaller datasets (e.g., β vs γ and δ). The effect of higher quality data over more data can be seen in the weaker performance on porosity with the Segformer model for datasets α and β when compared to datasets γ and δ . While datasets γ and δ were smaller, they had a higher ratio of porosity, as shown earlier in Table 1. And while dataset α contains dataset γ and δ , it also contains dataset β , which has a very poor ratio of porosity. This is even more apparent with dataset ϵ , which is made up of higher quality crops and, as a result, manages to obtain good performance on porosity, while being 16 times smaller. It, however, performs poorly on HAZ as the variation ϵ has very few crops that purely represent HAZ and substrate classes. Since Segformer predicts on crops, this leads to a lack of contextual information during the training process.

Fig. 9 shows the overall confusion matrix for the test set from dataset variation (e.g., δ) with the best-performing Segformer B0 model and the top-performing convolutional model from the same setting. While the Segformer model clearly shows the capacity to learn minority class and performs well on the remaining classes, it still struggles with distinguishing close to 22 % of porosity pixels from the background class for visibly similar intensity distributions. On the other hand, trained on

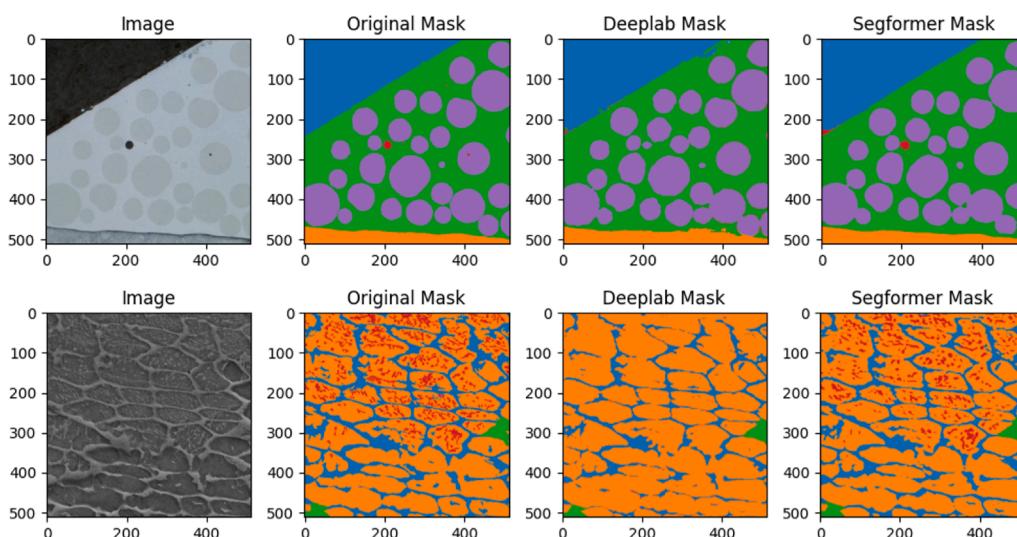


Fig. 10. A sample segmented crop for DEDAM-1 dataset using DeepLabv3+ and the Segformer B0 model (top) and a sample segmented crop for the MetalDAM dataset using the two models (below). The red color in the original mask represents the minority class (e.g., porosities and precipitates respectively) for both datasets. Transformer model clearly shows the capacity to learn the minority class under same data conditions for both cases. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

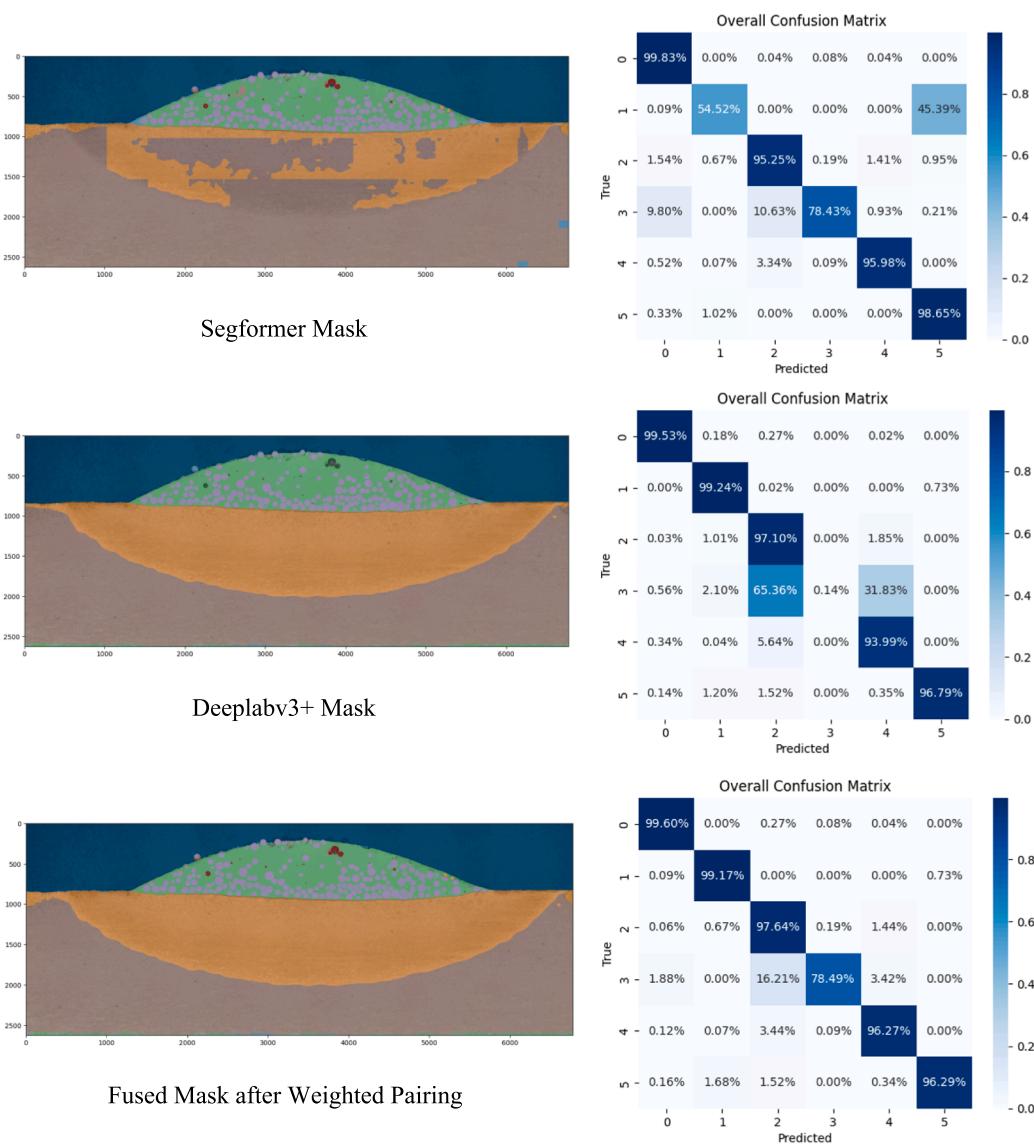


Fig. 11. Results from the proposed pairing (weighted) strategy on large-scale industrial beads (e.g., 7000 x 3000) to support rapid quantification. The transformer mask and the associated confusion matrix at the top highlight the weakness associated with the crop-wise prediction on large phases having intensity variations. The middle mask and associated confusion matrix show the weakness of the convolutional model in segmenting minority class. As a result, instead of large datasets, excessive training, or bulky ensembles, the two predictions are fused to exploit best aspects of the two models. In the confusion matrices, 0 corresponds to background, 1 corresponds to HAZ, 2 corresponds to matrix, 3 corresponds to porosities, 4 corresponds to carbides and 5 corresponds to the substrate classes.

the same dataset variation, the DeepLabv3+ model does not learn porosity at all. In addition to mislabeling the majority of the porosity pixels as background (64.8 %), the model also poorly contextualizes the pixels in the bead reinforcement area and misclassifies the remaining porosity pixels as either matrix (17.1 %) or carbides (18.0 %). Notably, the DeepLabv3+ model still outperforms the Segformer model on HAZ and matrix classes, due to the lack of contextual information for Segformer as mentioned previously. The minor interplay of misclassification between HAZ and substrate indicates that the transition from one phase or class to the other is not always easy to distinguish.

The results of validating DeepLabv3+ and Segformer B0 on the open-source MetalDAM dataset led to the transformer model dominating in all classes (e.g., like variation δ of dataset DEDAM-1). Moreover, the convolutional counterpart performed poorly on the minority class. This highlights the ability of Segformer to learn minority class distributions in imbalanced and sparsely labeled datasets where convolutional models fail to show competitive performance. We compare the dominant Segformer B0 model to the two state-of-the-art models in (Luengo, 2022) &

(Biswas, 2023) on the MetalDAM dataset in Table 9 based on class IoU and mean IoU. Based on the class-wise and mean IoU across classes, Segformer outperforms the two ensemble-based approaches. It is important to note that the apparent poor performance on the minority class could be due to several issues.

Luengo et al. (Luengo, 2022), while releasing the MetalDAM dataset mentioned the difficulty in labeling the precipitate class and hence not all pixels representing precipitates were labeled. This could have led to confusing the model and could be the reason that it was not included in the analysis by (Luengo, 2022) and (Biswas, 2023). Fig. 10 shows the two sample crops from DEDAM-1 and MetalDAM datasets alongside the ground truths and predicted masks by the top-performing models. Segformer's ability to segment minority class (in red) in both datasets can be seen in terms of segmented porosities and precipitates.

The fusion strategy was based off the results of the models on the test set. The fusion model is a weighted sum of the class predictions from the Segformer and DeepLabv3+ models. The weight is obtained from the confusion matrix of the models on the test set by taking the normalized

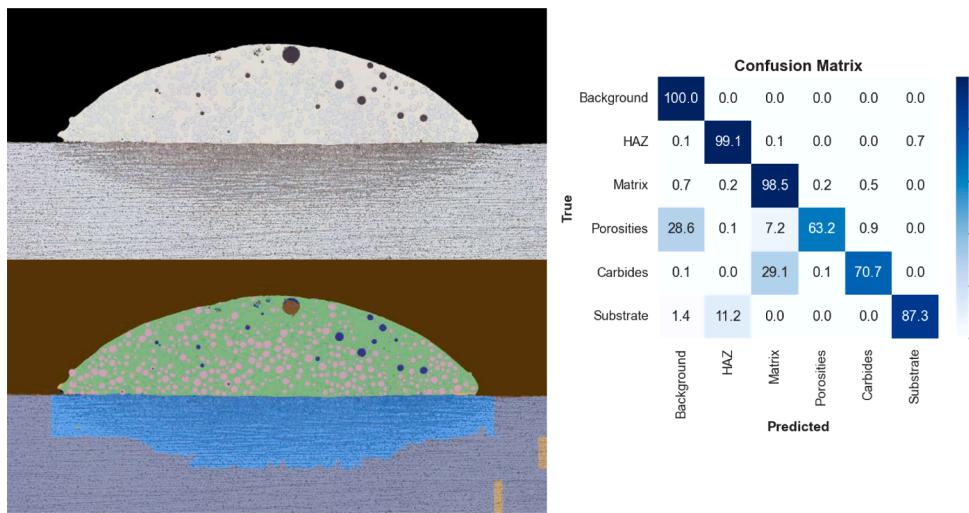


Fig. 12. Results of the generalization test. The Segformer B0 model is tested on an annotated full bead from DEDAM-2 dataset. Despite the difference in bead appearance and in intensity distribution for the same classes, the model achieves an overall accuracy of 93% with an accuracy of 63% on the minority class. As a result, the existing model can be easily fine-tuned to metallographic images from different contexts.

Table 10
Development environment.

Environment Variable	Value
Language	Python (Python, 2021)
GUI Library	Tk Interface (Dalgard, 2001)
Supported ML Frameworks	PyTorch, SMP
Distribution Package	PyInstaller
Distribution Format	Single File
Supported OS	Windows 11

percentage of the true background divided by the normalized percentage of all other classes predicted as background. The exact formula for the class weight per model is as follows:

$$\text{class weight} = \frac{(\text{normalised percentage of true positive})}{\text{sum of normalised percentage of positive}} \quad (8)$$

The class weights are then used in the calculation for the fusion matrix as follows:

$$P = w_1 \cdot p_1 + w_2 \cdot p_2 \quad (9)$$

where P is a $1 \times C$ matrix of predicted weight, w_1 is a $1 \times C$ matrix of the calculated class weights for the first model, and p_1 is a $C \times 1$ one-hot encoding of the prediction of the first model. The resulting prediction is an argmax of the matrix P . This method allows for a fusion model to obtain a higher overall IoU, and the capacity to predict porosities in Ni-WC MMC. The results of the fusion model compared to the results of the Segformer and the DeepLabv3+ models are shown in Fig. 11, where a second test set is used for the comparison.

Finally, we annotated a test bead of the same material system from a different industrial partner (e.g., DEDAM-2 dataset) to test the generalizability of the best-performing model to changing industrial conditions. Fig. 12 shows the original bead cross-section, the same cross-section with the segmentation mask, and the corresponding confusion

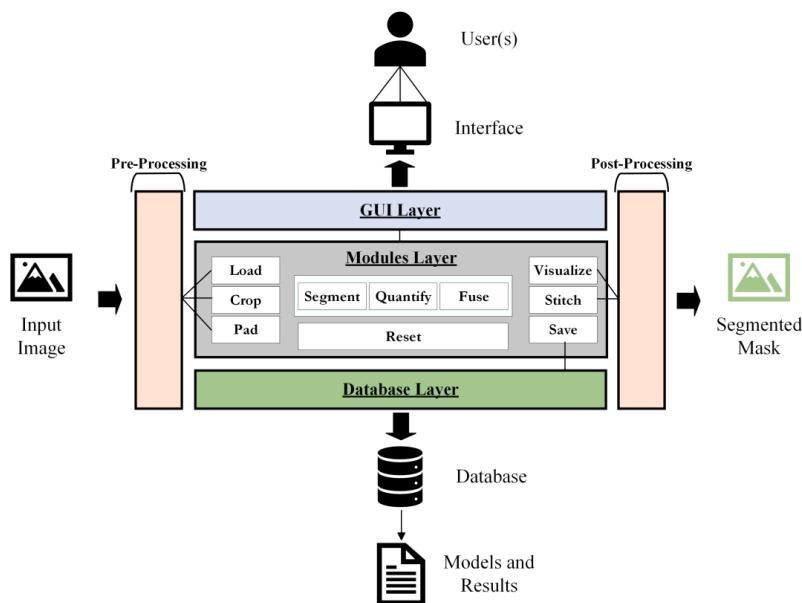


Fig. 13. Overall architecture of MicroSegQ+ with pre-processing, GUI, Modules, Database, and Post-Processing layers.

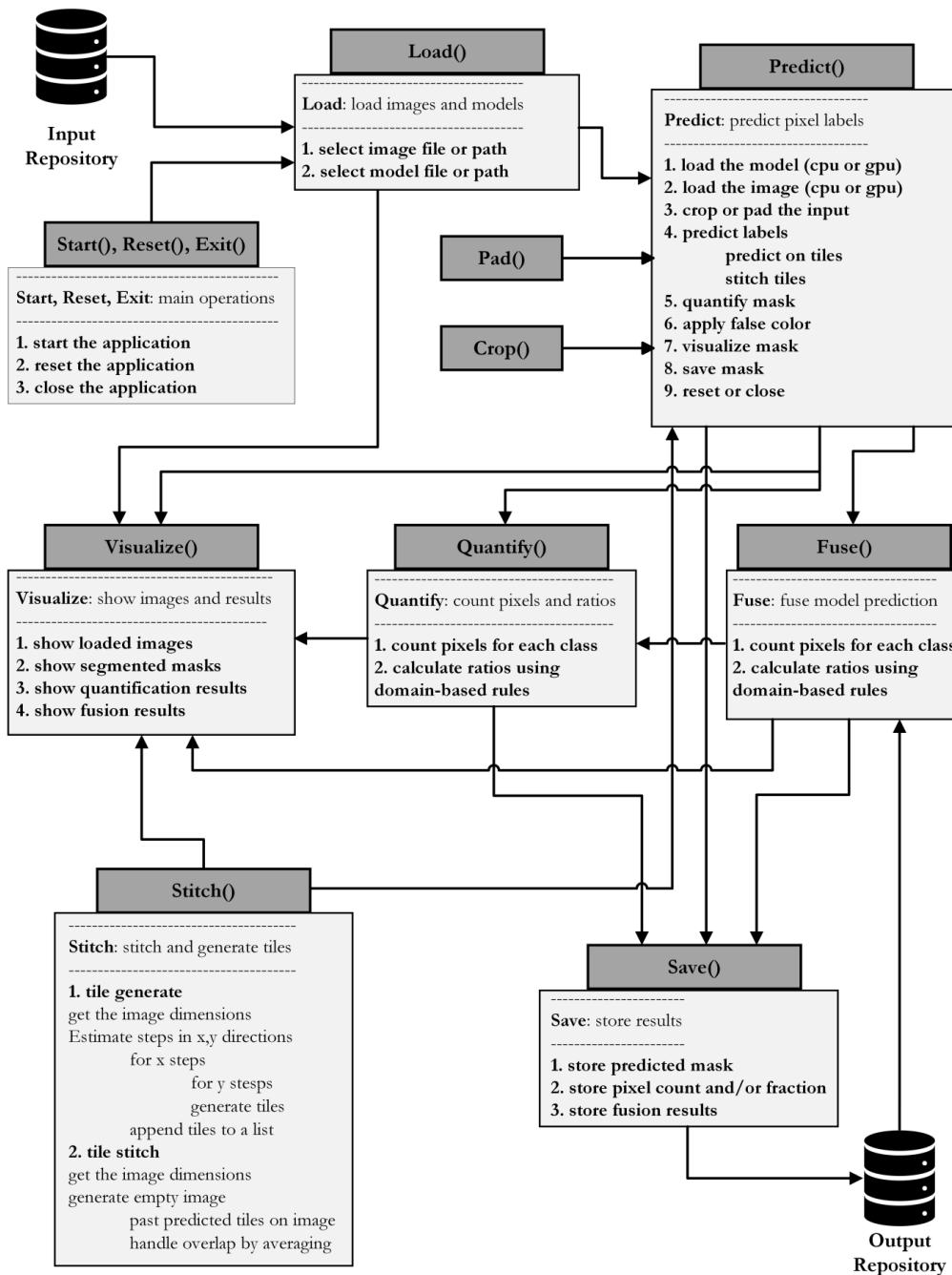


Fig. 14. Relationship between different functional modules in MicroSegQ+. The modules are arranged in the order of their use during segmentation. The upper block highlights the name of each module, while the body text specifies the subfunctions available.

matrix. Without any fine-tuning on this new dataset, the model does extremely well (e.g., 93 % accuracy) in segmenting the classes including porosities. This was a challenging sample given the presence of porosities in different sizes and shapes. Most of the errors in segmenting porosities come from the large void in the middle indicating the model lacks contextualization for large porosities and confuses them with the background. Nearly 30 % of carbide pixels are predicted as matrix indicating poor localization at the boundaries. Nonetheless, the model shows promise to overcome segmentation errors through further fine-tuning.

6. Developing the MicroSegQ+ Tool

To meet the needs of the industry to rapidly segment and quantify either random-sized crops or complete cross-sections of AMed metallographs, we developed a simple and lightweight tool based on Python. In addition to segmentation and quantification, the tool also enables the fusion of predictions from different models to improve the segmentation results. The key idea is to bring flexibility in the choice of models (e.g., attention-based, convolution-based, or more). The models can be selected on the go and their results can be stored, visualized, and quantified, whereas the stored results can be later fused with other predictions using the fusion logic presented earlier.

The Python-based tool is developed using different libraries

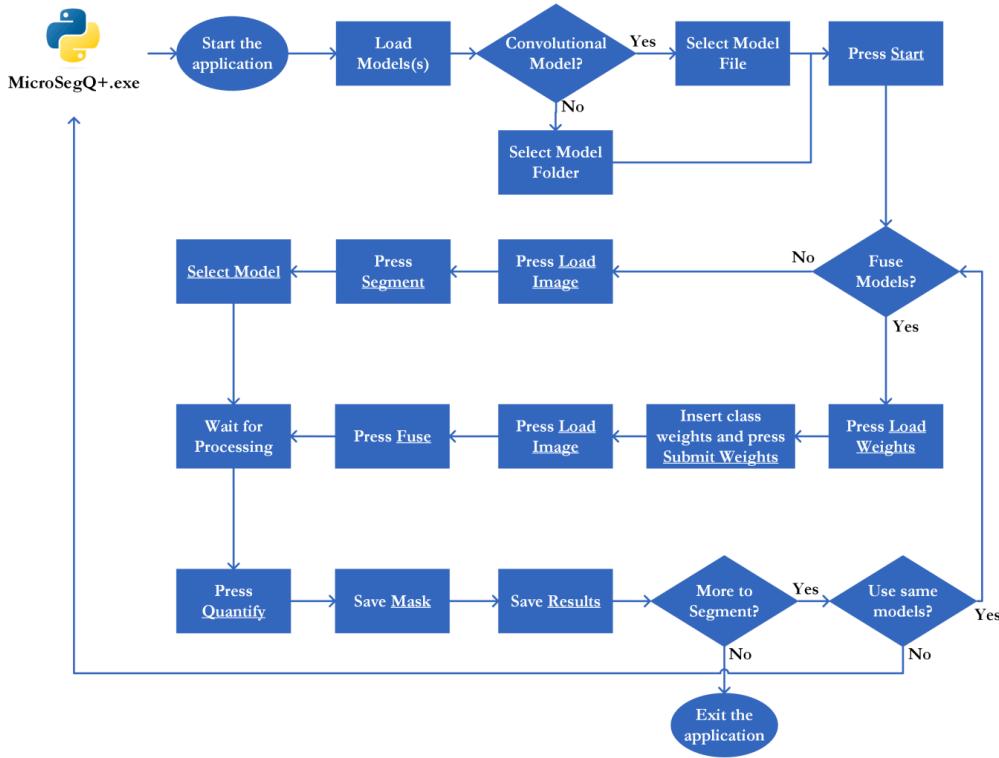


Fig. 15. Steps to segment with and without fusion using Version 1.0 of MicroSegQ+.

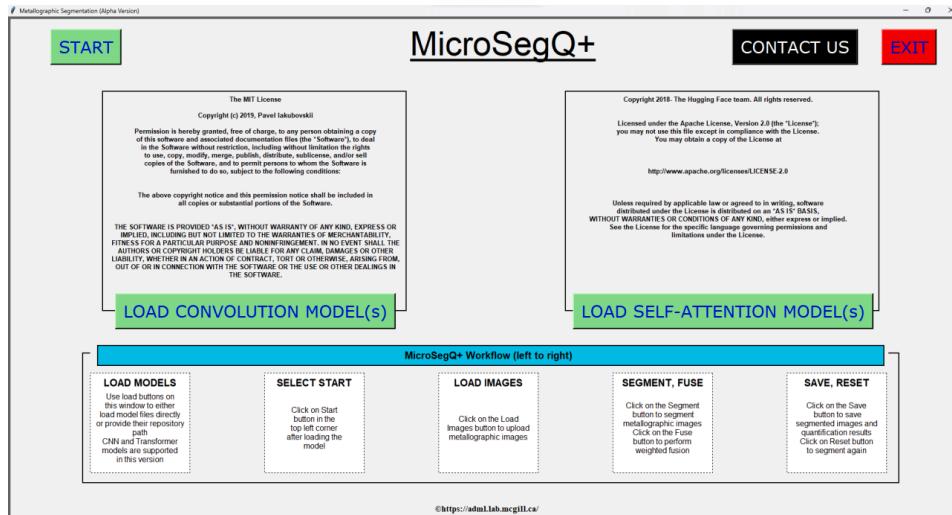


Fig. 16. Welcome window depicting segmentation workflow, license information, and model loading buttons, the Start button on top left takes the user to the segmentation window after the segmentation model(s) have been uploaded.

(Table 10). Tkinter or Tk interface is used to later wrap the functionalities in a GUI. The pipeline results in a ready-to-use execution file (.exe) in a distribution folder (dist.zip) that can be zipped and released to the industry. The main modules of the tool architecture can be arranged according to its functionalities. Fig. 13 shows the architecture of the tool in terms of its modules. The key modules include load, crop, pad, stitch, store/save, quantify, fuse, visualize, and predict.

Fig. 14 illustrates the algorithmic flow and relationship among the main modules of MicroSegQ+ alongside their major pseudocode steps. Once the application is started, the user is asked to provide the path of the trained segmentation model for inference. Following the successful loading of the model, the user is prompted to input the image file for

segmentation. These functionalities are grouped in the load module. The key module of the application is the prediction module, which makes use of the remaining functionalities. Once the user selects the segmentation command in the GUI, the inference operation is run to segment the image. For this to work, the model and image are loaded as objects. For fully convolutional models, the whole image may be segmented without the need of crops, whereas the transformer models depend on the tile generation function to perform patch-wise segmentation that are later stitched together. Once the segmentation is performed, the results can be visualized using a color map, quantified into total or class ratios, and stored. The fusion operation enables the fusion of predictions from different models in a pair-wise weighted fashion to enhance the



Fig. 17. Segmentation window. The green buttons represent major implemented functions in this version. On this screen, users can load images, insert fusion weights, segment with or without fusion, and quantify the results of segmentation. Moreover, the segmented masks and counted pixels can be saved locally. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

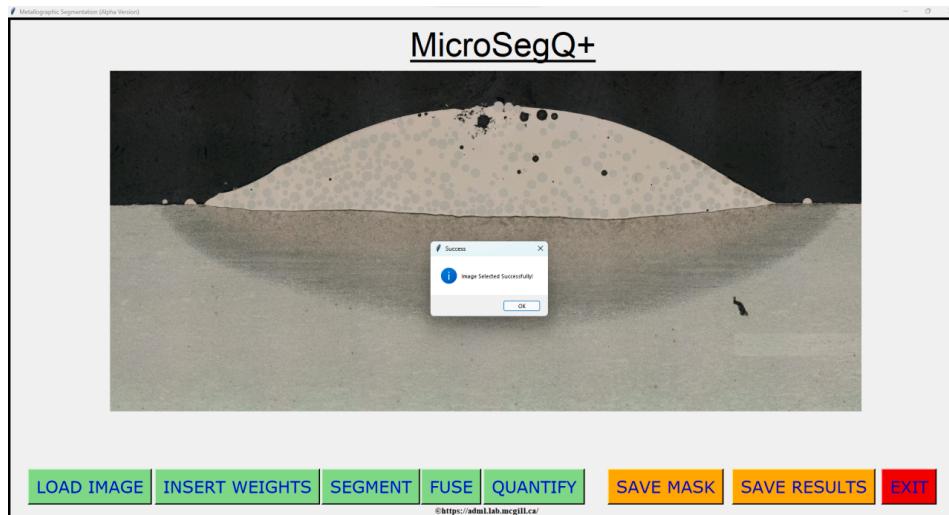


Fig. 18. A sample loaded bead cross-section in Segmentation Window. The message box highlights whether an operation was successful or not.

segmentation results.

Fig. 15 outlines a step-by-step workflow to use the tool. Figs. 16 to 21 show different windows of MicroSegQ+ and highlight their functionalities. The startup window in Fig. 16 for loading the models also provides instructions to the users and contact information for questions or help. The segmentation window shown in Fig. 17 lets the user select and upload metallographs of random size that are resized to fit the available screen resolution. Fig. 19 highlights the options to visualize and save the results with MicroSegQ+. In addition to the segmentation mask being pasted on the original image, the results can be saved in an Excel file with time and date stamps. Finally, the color-converted segmented mask can be saved locally.

The current version of the MicroSegQ+ can handle a variety of situations and further improvements are planned for future versions. First, the models can be expanded to more than two categories and the tool can be programmed to detect the model type based on some set rules. Moreover, the phases to segment can be selected on the go depending on the data and model. Currently, this is fixed for the two MMC datasets used in the experiments. The tool can also be expanded to fuse predictions from more than two models in a majority voting-based

ensemble model.

7. Conclusions and future works

Inspired by the challenges of segmenting and quantifying large-scale industrial metallographic images from AM processes, this work explored state-of-the-art convolutional and vision transformer models. First, we prepared and annotated two datasets representing the microstructure of an additively deposited MMC material. Key model types from the two categories were explored to shortlist three models namely U-Net++ (convolution-based), DeepLabv3+ (convolution-based), and the Segformer B0 (transformer). We systematically prepared dataset variations to investigate the performance of the models at varying levels of quantity and quality. Convolutional architectures performed well on large and continuous phases, while struggling to segment minority defect class. The transformer model on the other hand gave competitive performance on large phases and showed the capacity to learn the minority class. This complimentary performance across the additively deposited MMC dataset settings led to a pairing strategy for rapid quantification. The best-performing models were also fine-tuned and tested on an open-



Fig. 19. A segmented bead cross section with segmentation mask in the segmentation window.

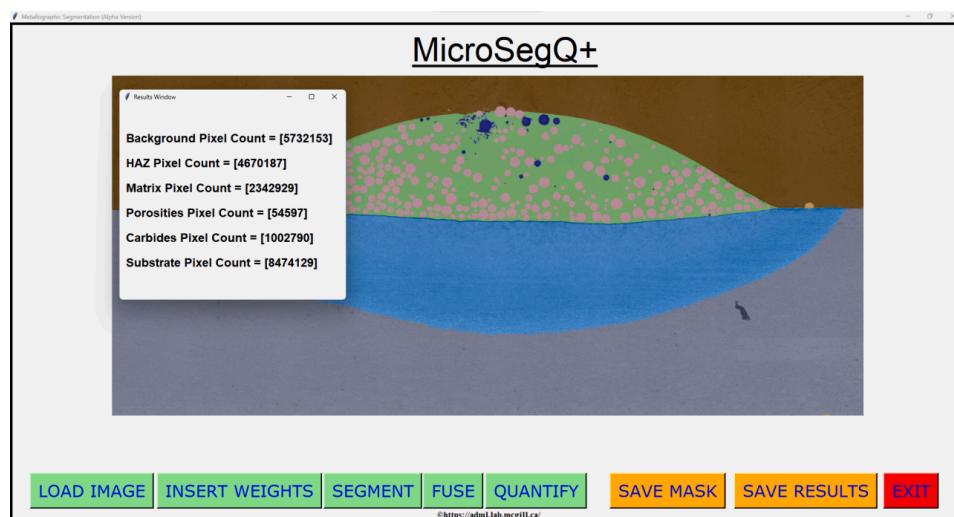


Fig. 20. Results window with counted pixels from segmentation mask.

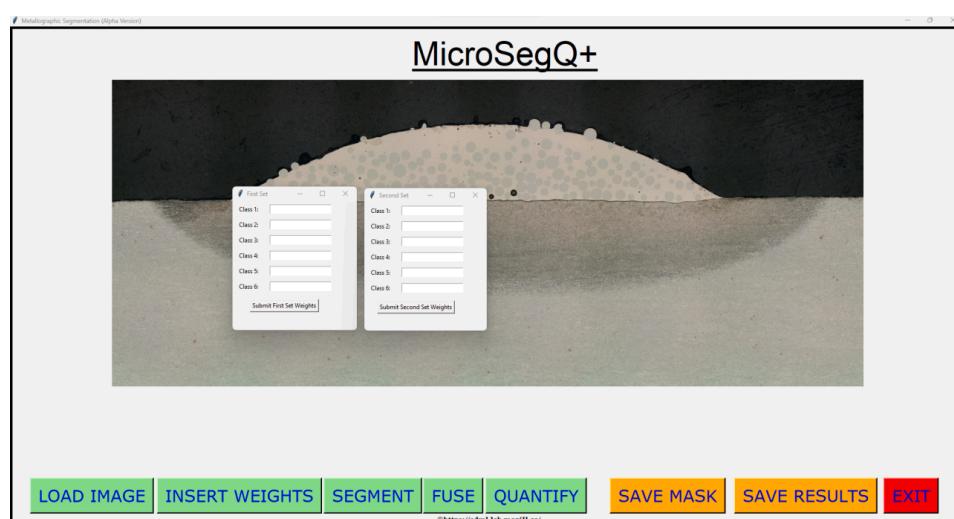


Fig. 21. A view of the model weight windows through the insert weight button.

source metal AM dataset containing minority defect class. Since industrial datasets are prone to conditions that can introduce changes in their (intensity) distribution, we tested the generalizability potential of our best model. The extensive experiments on dataset variations indicated the potential of small-size and high-quality datasets for the fusion strategy.

In order to make the models usable in industry, a simple and efficient tool with basic functions to segment and quantify AMed metallographic images was developed using Python. The tool, MicroSegQ+, is based on a modular concept to support diverse datasets and models for the segmentation task. The tool was packaged into a single standalone executable (.exe) file for industrial deployment, while the segmentation models were saved as individual files or sets of files. The minimalist and modular approach makes the tool efficient and easy to use for operators and technicians. The tool also enables visualization of segmentation and quantification results. These results can also be saved for future analysis. Finally, the tool also supports the fusion strategy adopted in this work. The updated weights for each model in the pair can be loaded in the tool before fusion. The fusion function uses the weight arrays to fuse segmentation results from individual models before visualization.

Future work can be divided between modeling and tool development, where datasets can be prepared on demand. As a result, the directions to pursue for future studies include:

- Fine-tuning Segformer B0 model on DEDAM-2 dataset to assess the extent of annotated data and the learning effort needed when moving from one type of bead cross-section to another.
- Expanding to include recently proposed ViTs in the experiments to compare their performance with the Segformer model.
- Expanding the developed models and fusion strategy to other AM datasets with minority defect classes.
- Integrating the model fine-tuning functionality within the Ver 2.0 of MicroSegQ+ to transition towards a no-code ML pipeline for industrial applications.
- Enhancing the GUI of MicroSegQ+ by enabling class and data-specific settings (e.g., prompt for the labels to segment).
- Expanding the applications of MicroSegQ+ to design and process-based AM datasets (e.g., segmentation of in-process defects and anomalies).

CRediT authorship contribution statement

Mutahar Safdar: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing – original draft. **Yi Fan Li:** Conceptualization, Methodology, Validation, Investigation, Data curation, Writing – original draft. **Randy El Haddad:** Conceptualization, Methodology, Validation, Investigation, Data curation, Writing – original draft. **Max Zimmermann:** Resources, Data curation, Writing – original draft. **Gentry Wood:** Resources, Data curation, Writing – original draft. **Guy Lamouche:** Conceptualization, Methodology, Writing – review & editing, Supervision, Project administration, Funding acquisition. **Priti Wanjara:** Conceptualization, Methodology, Writing – review & editing, Supervision, Project administration, Funding acquisition. **Yaoyao Fiona Zhao:** Conceptualization, Methodology, Writing – review & editing, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The annotated datasets DEDAM-1 (https://huggingface.co/datasets/ironchancellor/DEDAM_1)

and DEDAM-2 (https://huggingface.co/datasets/ironchancellor/DEDAM_2) have been publicly released on the Hugging Face Hub. An executable version of the tool can be made available upon request to the corresponding author.

Acknowledgements

The authors are grateful to Digital Research Alliance of Canada (RRG# 4294) for providing computational resources to support this research. We would like to acknowledge the Artificial Intelligence Enhancement of Process Sensing for Adaptive Laser Additive Manufacturing (AI-SLAM) consortium between Canada and Germany for their support that made this work possible.

Funding

McGill Engineering Doctoral Award (MEDA) fellowship for M.S. is acknowledged with gratitude. M.S. also received financial support from National Research Council Canada (Grant# NRC INT-015-1).

References

- Pei, E., et al. (2023). *Springer Handbook of Additive Manufacturing*. Springer Nature.
- ASTM. (2022). *Additive manufacturing — General principles — Fundamentals and vocabulary*. American Society of Testing Materials.
- Zhu, K., Fuh, J. Y. H., & Lin, X. (2021). Metal-based additive manufacturing condition monitoring: A review on machine learning based approaches. *IEEE/ASME Transactions on Mechatronics*.
- Oleff, A., et al. (2021). Process monitoring for material extrusion additive manufacturing: A state-of-the-art review. *Progress in Additive Manufacturing*, 1–26.
- Murr, L. E. (2018). A metallographic review of 3D printing/additive manufacturing of metal and alloy products and components. *Metallography, Microstructure, and Analysis*, 7, 103–132.
- Gurland, J. (1984). Application of quantitative microscopy to cemented carbides. *ASTM International*.
- Zhang, Y., et al. (2022). A systematic review on data of additive manufacturing for machine learning applications: The data quality, type, preprocessing, and management. *Journal of Intelligent Manufacturing*, 1–36.
- Rose, D., et al. (2022). Automated semantic segmentation of NiCrBSi-WC optical microscopy images using convolutional neural networks. *Computational Materials Science*, 210, Article 111391.
- Guo, Y., et al. (2018). A review of semantic segmentation using deep neural networks. *International Journal of Multimedia Information Retrieval*, 7, 87–93.
- Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. *International Conference on Medical image computing and computer-assisted intervention*. Springer.
- Thisanek, H., et al. (2023). Semantic segmentation using Vision Transformers: A survey. *Engineering Applications of Artificial Intelligence*, 126, Article 106669.
- Hao, S., Zhou, Y., & Guo, Y. (2020). A brief survey on semantic segmentation with deep learning. *Neurocomputing*, 406, 302–321.
- Safdar, M., et al. (2023). *Engineering of additive manufacturing features for data-driven solutions: Sources, techniques, pipelines, and applications*. Springer Nature.
- Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. *Proceedings of the IEEE conference on computer vision and pattern recognition*.
- Hoiem, D., Divvala, S. K., & Hays, J. H. (2009). Pascal VOC 2008 challenge. *World Literature Today*, 24(1).
- Sultana, F., Sufian, A., & Dutta, P. (2020). Evolution of image segmentation using deep convolutional neural network: A survey. *Knowledge-Based Systems*, 201, Article 106062.
- Dosovitskiy, A., et al., *An image is worth 16x16 words: Transformers for image recognition at scale*. arXiv preprint arXiv:2010.11929, 2020.
- Zheng, S., et al. (2021). Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.
- Scime, L., et al. (2020). Layer-wise anomaly detection and classification for powder bed additive manufacturing processes: A machine-agnostic algorithm for real-time pixel-wise semantic segmentation. *Additive Manufacturing*, 36, Article 101453.
- Schmitt, A.-M., et al. (2023). Powder bed monitoring using semantic image segmentation to detect failures during 3D metal printing. *Sensors*, 23(9), 4183.
- Jin, Z., et al. (2021). Precise localization and semantic segmentation detection of printing conditions in fused filament fabrication technologies using machine learning. *Additive Manufacturing*, 37, Article 101696.
- Zhang, J., et al. (2022). Image segmentation for defect analysis in laser powder bed fusion: Deep data mining of X-ray photography from recent literature. *Integrating Materials and Manufacturing Innovation*, 11(3), 418–432.
- Wang, K. (2023). Contrastive learning-based semantic segmentation for In-situ stratified defect detection in additive manufacturing. *Journal of Manufacturing Systems*, 68, 465–476.

- Scott, S., Chen, W.-Y., & Heifetz, A. (2023). Multi-task learning of scanning electron microscopy and synthetic thermal tomography images for detection of defects in additively manufactured metals. *Sensors*, 23(20), 8462.
- Luengo, J., et al. (2022). A tutorial on the segmentation of metallographic images: Taxonomy, new MetalDAM dataset, deep learning-based ensemble model, experimental analysis and challenges. *Information Fusion*, 78, 232–253.
- Biswas, M., et al. (2023). Microstructural segmentation using a union of attention guided U-Net models with different color transformed images. *Scientific Reports*, 13(1), 5737.
- Sofiiuk, K., I. Petrov, and A. Konushin, *Reviving iterative training with mask guidance for interactive segmentation*. arXiv 2021. arXiv preprint arXiv:2102.06583.
- Liu, Q., et al. (2022). PseudoClick: Interactive image segmentation with click imitation. *European conference on computer vision*. Springer.
- Qin, J., et al. (2022). Research and application of machine learning for additive manufacturing. *Additive Manufacturing*, Article 102691.
- Zhou, Z., et al. (2018). Unet++: A nested u-net architecture for medical image segmentation. *Deep learning in medical image analysis and multimodal learning for clinical decision support: 4th International workshop, DLMIA 2018, and 8th international workshop, ML-CDS 2018, Held in conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings* 4. Springer.
- Chen, L.-C., et al. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. *Proceedings of the European conference on computer vision (ECCV)*.
- Chen, L.-C., et al. (2017). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4), 834–848.
- Chen, L.-C., et al., *Rethinking atrous convolution for semantic image segmentation*. arXiv preprint arXiv:1706.05587, 2017.
- Bao, H., et al., *Beit: Bert pre-training of image transformers*. arXiv preprint arXiv: 2106.08254, 2021.
- Strudel, R., et al. (2021). Segmenter: Transformer for semantic segmentation. *Proceedings of the IEEE/CVF international conference on computer vision*.
- Xie, E., et al. (2021). SegFormer: Simple and efficient design for semantic segmentation with transformers. *Advances in Neural Information Processing Systems*, 34, 12077–12090.
- Python, W. (2021). Python. *Python Releases for Windows*, 24.
- Dalgaard, P. (2001). The R-tcl/tk interface. *Proceedings of DSC*.