Acta Polytechnica CTU Proceedings **42**:87–93, 2023
https://doi.org/10.14311/APP.2023.42.0087

Published by the Czech Technical University in Prague

SEGMENTATION OF PORES IN CARBON FIBER REINFORCED POLYMERS USING THE U-NET CONVOLUTIONAL NEURAL NETWORK

Miroslav Yosifov[a,b,*], Patrick Weinberger[b], Bernhard Plank[b], Bernhard Fröhler[b], Markus Hoeglinger[b], Johann Kastner[b], Christoph Heinzl[c]

[a] *University of Antwerp, imec-Visionlab, Department of Physics, Universiteitsplein 1, 2610 Antwerpen, Belgium*
[b] *University of Applied Sciences Upper Austria, Research Group X-ray Computed Tomography, Stelzhamerstraße 23, 4600 Wels, Austria*
[c] *University of Passau, Faculty of Computer Science and Mathematics, Innstraße 43, 94032 Passau, Germany*
[*] corresponding author: miroslav.yosifov@fh-wels.at

Abstract. This study demonstrates the utilization of deep learning techniques for binary semantic segmentation of pores in carbon fiber reinforced polymers (CFRP) using X-ray computed tomography (XCT) datasets. The proposed workflow is designed to generate efficient segmentation models with reasonable execution time, applicable even for users using consumer-grade GPU systems. First, U-Net, a convolutional neural network, is modified to handle the segmentation of XCT datasets. In the second step, suitable hyperparameters are determined through a parameter analysis (hyperparameter tuning), and the parameter set with the best result was used for the final training. In the final step, we report on our efforts of implementing the testing stage in open_iA, which allows users to segment datasets with the fully trained model within reasonable time. The model performs well on datasets with both high and low resolution, and even works reasonably for barely visible pores with different shapes and size. In our experiments, we could show that U-Net is suitable for pore segmentation. Despite being trained on a limited number of datasets, it exhibits a satisfactory level of prediction accuracy.

Keywords: Deep learning, segmentation, U-Net, computed tomography, pores, carbon fiber reinforced polymers.

1. Introduction

Fiber reinforced polymers are getting increasingly important in our daily lives. As they are used in safety-critical areas such as aerospace, fast and reliable analysis of these materials is necessary. Industrial X-ray computed tomography is a powerful tool to facilitate such an analysis, but it is still a huge effort to analyze, visualize and quantify the data generated in an XCT scan [1].

Deep learning has been successfully employed in fields like computer vision [2], object recognition [3], medical image analysis [4], face-recognition applications, and material inspection [5–7] over the past few decades. Convolutional neural networks (CNN) are one of the highly popular classes of deep learning methods, which has showed a remarkable ability to handle classification and segmentation tasks for image, volume, and video.

Material science domain experts often need to segment pores in CFRP to characterize materials, especially with regards to pores. However, segmentation of images or volumes remains a challenging problem. Despite the availability of various segmentation and classification techniques such as k-means [8, 9], watershed [10] and thresholding techniques, there are still several research efforts to improve segmentation methods [11, 12]. Accurate and efficient segmentation plays a important role in ensuring the integrity of safety-critical materials. To achieve a robust and reliable segmentation, we have developed a modified version of the U-Net architecture. Our contributions in this work are:

- The modified 3D U-Net designed for efficient pore segmentation of XCT scans from CFRP.
- A network architecture and testing application in open_iA [13] which allows predictions on consumer-grade GPUs.
- Evaluation of the neural network's performance on unseen data with varying resolutions and sizes.
- Visual analysis and quantitative comparison of the network's predictions with the results (labels) generated from the Otsu thresholding technique.

2. Background and related work

Segmentation is defined as classification of an image into regions which contain similar characteristics (e.g., similar grey values). Several segmentation methods can be applied for segmenting pores in XCT scans of fiber-reinforced polymers. Here a short overview is given over segmentation methods employed in this

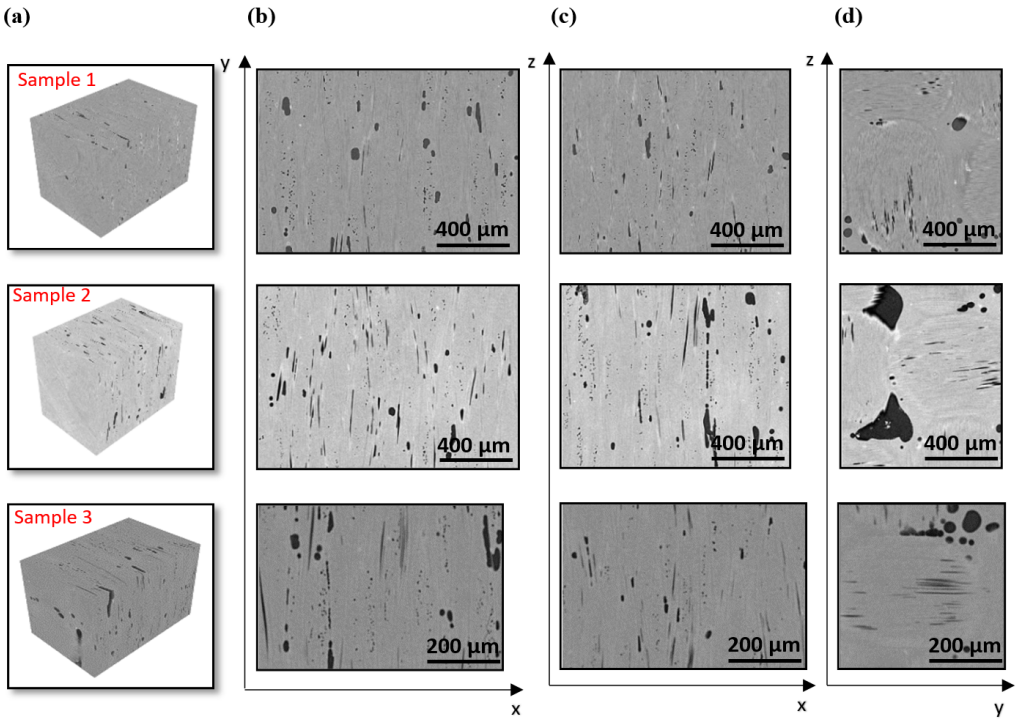

Figure 1. (a) shows 3D rendering of Sample 1, Sample 2, and Sample 3. (b), (c) and (d) respectively display axis-aligned 2D slices of Sample 1, 2 and 3.

area, the methods that we used in our work and those which we based our work on.

2.1. Thresholding-based segmentation

Global thresholding techniques are used in a wide range of diverse applications. A detailed summary on thresholding techniques has been published by Sezgin and Sankur [14]. Various kinds of threshold methods have been described and compared on carbon fiber polymer samples by Rao et. al. [12]. In this paper we have decided to use Otsu thresholding [15] to create label images for the training, testing and validation as shown by Reh et al. segmenting pores [16]. In addition, it is a non-parametric threshold method. Therefore, no additional user-depended parameters have to be defined.

2.2. Convolutional neural networks

Convolutional neural networks now are the state-of-the-art technique in biomedical image segmentation [17]. One of these CNN architectures is U-Net. U-Net was initially developed to address the segmentation task of 2D biomedical electron microscopic images and demonstrated remarkable success during the ISBI cell tracking challenge in 2015 [17]. It has been successfully applied to various segmentation tasks, including cell, organ segmentation [18] and brain tumour detection [19]. Later, U-Net was further developed to handle 3D input shapes for segmenting 3D microscopy images [18].

3. Data characteristic

In this paper, we have used two different samples of the same material which have been obtained from fiber reinforced composites. $10 \times 10 \times 2\,\mathrm{mm}^3$ carbon fiber reinforced polymer samples were cut out from an $360 \times 510 \times 2\,\mathrm{mm}^3$ plate. The plates for Sample 1 were manufactured by a Wet Lay Up process with a Vacuum Bag and for Sample 2 a Vacuum Assisted Resin Infusion (VARI) process was used. Both sample plates were manufactured of six layers of an 2×2 twill weave pattern. Figure 1 shows the training data (Sample 1), the testing data (Sample 2) and the testing data with lower resolution $10\,\mu\mathrm{m}^3$ (Sample 3). Volume Graphics was used for registration of different resolution.

XCT Scans were performed on a GE Nanotom 180 NF XCT-device. Using an ROI-CT mode, $3.3\,\mu\mathrm{m}$ voxel size can be reached. An additional low-resolution scan with $10\,\mu\mathrm{m}$ voxel size was performed on the VARI material (Sample 3). For further data processing, following cut-out dimensions of the datasets were used, shown in Table 1.

	Resolution	**Size**
Sample 1	$3.3\,\mu\mathrm{m}$	$1300 \times 900 \times 976$
Sample 2	$3.3\,\mu\mathrm{m}$	$1300 \times 900 \times 976$
Sample 3	$10\,\mu\mathrm{m}$	$366 \times 244 \times 244$

Table 1. Sample Modality

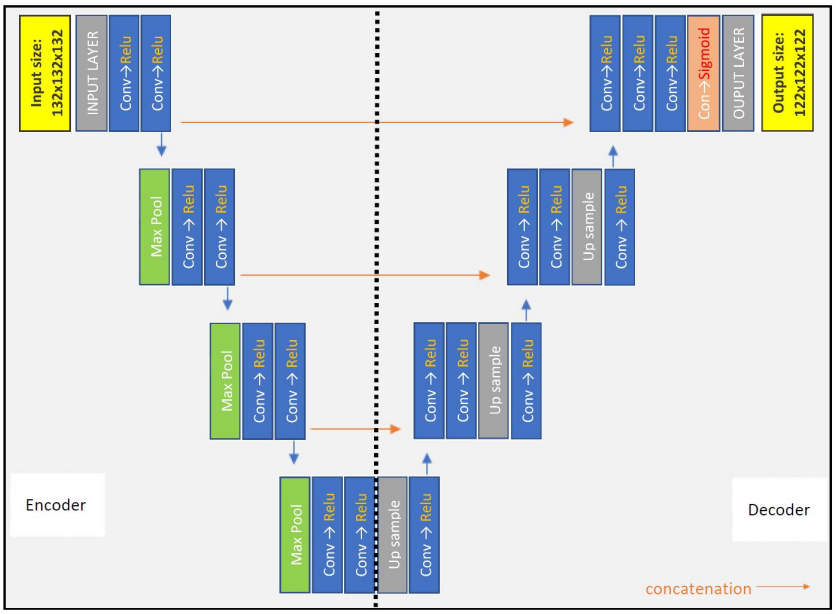

FIGURE 2. The modified 3D U-Net architecture [5].

4. METHODS

In the following section we describe our modified 3D U-Net (Section 4.1), the respective data pre-processing steps, as well as the used training process (Section 4.2).

4.1. MODIFICATION OF 3D U-NET

U-Net is a convolutional neural network that features high performance with a small amount of training, which can be applied both for 2D and 3D data. We implemented an architecture in Python [20] using Keras [21] with TensorFlow [22] as a backend which was inspired by 3D U-Net. The modified 3D U-Net network contains two paths respectively called analysis and synthesis path, with a U-shaped architecture containing in total 46 layers (see Figure 2).

The modified 3D U-Net architecture contains the input layer, encoder and decoder sections, a final convolution and transpose output layer. For up-sampling, the decoder subnetworks also contain one more convolution layer. First, encoder sections analyze the whole image, then the decoder sections produce and predict the segmented volume [5]. The modified 3D U-Net has two additional convolutional and Rectified Linear Unit (ReLU) layers in the decoder stages if we compare with original 3D U-Net. Rather than employing a softmax function, a sigmoid function was utilized in this case to reduce complexity in the final classification. Additional layers were integrated into the decoder stages with the aim of enhancing the final prediction accuracy. As demonstrated in our previous study [5], the segmentation results were improved compared to the original U-net architecture.

The modified 3D U-Net uses an overlap-tile strategy to predict the labels for each test volume. The input volume has dimensions of $132 \times 132 \times 132$, while the prediction size is $122 \times 122 \times 122$. The variation in input and output sizes is due to a 5-voxel overlap in each dimension. This overlapping strategy enhances prediction accuracy, particularly at the borders of the sub-volumes [17]. The neural network model (the weights) was saved in ONNX (Open Neural Network Exchange) format [23] for use in open_iA [13].

4.2. DATA PRE-PROCESSING, TRAINING AND TESTING

The preprocessing pipeline involved several steps, including volume normalization, mirror-padding with 5 extra voxels, and division of the CT scan into sub-volumes with a 5-voxel overlap in all directions. The labeled image, processed using Otsu thresholding, was also split into sub-volumes for training. The primary reason for splitting XCT data into sub-volumes with a size of $132 \times 132 \times 132$ is to optimize the training time and augment the training data, thereby reducing the need for scanning multiple specimens.

For training, only the sub-volumes from Sample 1 were used, which were further divided into three parts: 64 % (358 samples), for training, 20 % (112 samples) for testing, and 16 % (90 samples) for validation. To identify optimal training parameters, a heuristic hyperparameter tuning with Talos [24] was employed. In the hyperparameter tuning of our training we tested various combinations of learning rates, optimizers, and epochs to achieve an optimum between accuracy and training time. The hyperparameter tuning process yielded the best outcomes when employing the Adam optimizer [25] with a learning rate of 4e-5. The training process, conducted on a workstation powered by an Nvidia Quadro RTX 6000 with Keras 2.3.0 and TensorFlow-GPU 2.1.0, was successfully completed in approximately 1 hour, involving 10 epochs and a batch size of 3. To distinguish classes with significant imbalances during training, the dice coefficient [26]

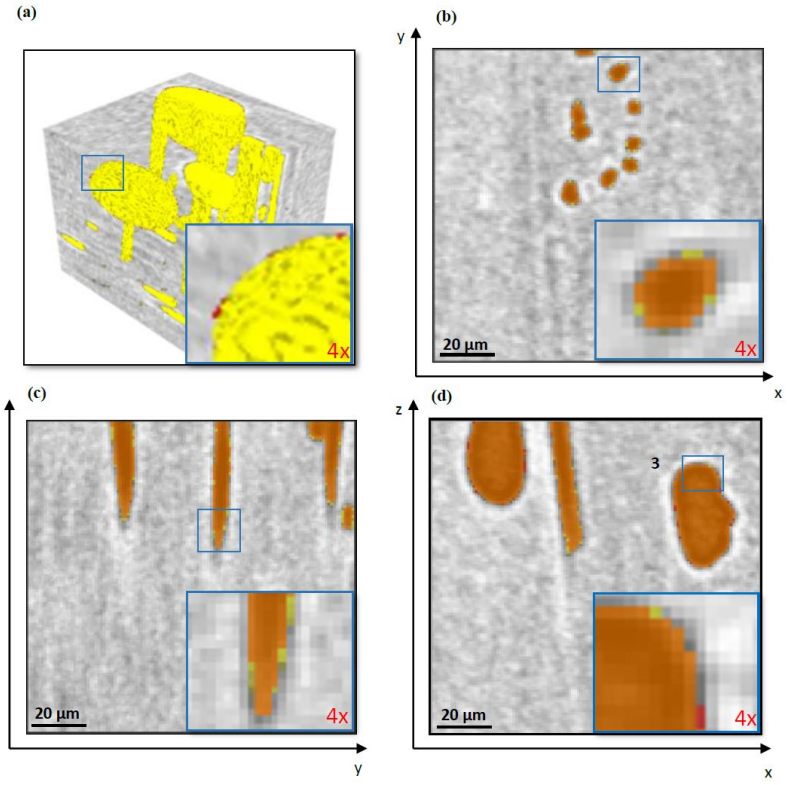

Figure 3. The visualization consists of 2D and 3D representations of a sub-volume extracted from Sample 2. (a) displays the superimposed predictions (yellow) on the input volume and the Otsu thresholding (red) in a 3D visualization. (b), (c) and (d) respectively display axis-aligned 2D slices of predictions (yellow) overlaid on the input volume and Otsu thresholding (red). Orange colour represents the agreement between the prediction and Otsu thresholding regions. Additionally, the detail images provide a closer look at a specific region marked in blue, magnified four times.

function was employed as loss function. Following the training process, the obtained results were evaluated using the test dataset. In the training process, the input size is fixed at $132 \times 132 \times 132$; however, during the testing stage, data of any size can be applied. In our case, we segmented the entire XCT volumes with the trained model, as well as exemplary sub-volumes.

For the prediction phase, a different system was employed, featuring a consumer-grade GPU (Nvidia GeForce GTX 1080). Predicting the Sample 2 ($1300 \times 900 \times 976$) using open_iA and the ONNX runtime took approximately four minutes. This time includes the normalization of the input, splitting it into sub-volumes, performing the prediction, and merging the sub-volumes into the final result.

5. Results and discussion

In this section, we present some of the results of the predictions in comparison to Otsu thresholding. We measure the segmentation accuracy using the dice coefficient function [26]. This function computes the dice similarity coefficient between the prediction and reference segmentations. Training and validation accuracy were both determined at appr. 99 %. On the training process, those values were continuously growing, and the final model achieved satisfactory results.

The result of the evaluation with the test dataset was a dice coefficient of 0.9822 and an accuracy of 0.9990. The grey values were normalized to be in the range from zero to one for all of the samples before the training and testing process. Otsu thresholding segmentations (labels) are binarized to zero or one (white values = 1, black values = 0) for all the samples. For each voxel, the prediction delivers a value between 0 and 1, denoting the probability for this voxel to be a pore. We have used our prediction without any post-processing to show the probability of the predicted voxels.

5.1. Sub-volume prediction and visualization

The neural network model in open_iA was utilized to perform segmentation on a sub-volume extracted from Sample 2, which had dimensions of $122 \times 122 \times 122$. The obtained segmentation results are presented in Figure 3. This subsection of the dataset has been chosen since it contains many different shapes of pores.

According to the predictions, all kind of pores with different shapes are segmented both with Otsu thresholding and with our modified 3D U-Net approach. For bigger pores, the overall results seem quite similar with both methods. In the magnifications, it becomes obvious that the Otsu thresholding segmentations are

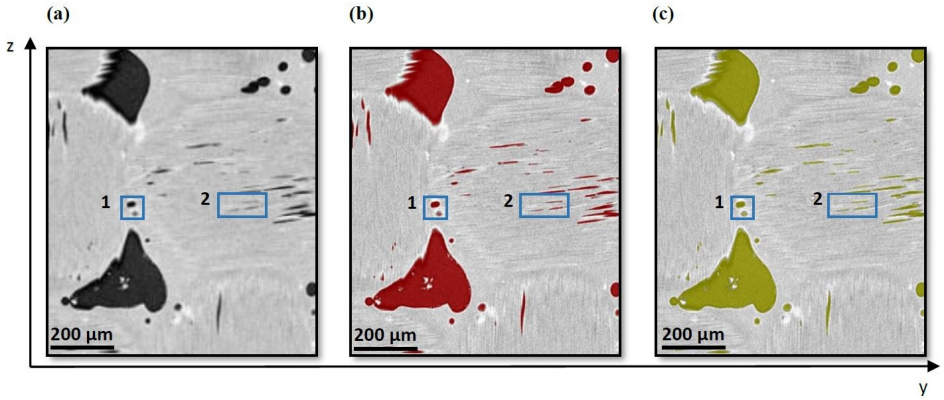

FIGURE 4. Original input data slice from Sample 2 (a), respectively show 2D slices of Otsu thresholding (red) (b) and prediction (yellow) (c) overlaid on an input volume slice.

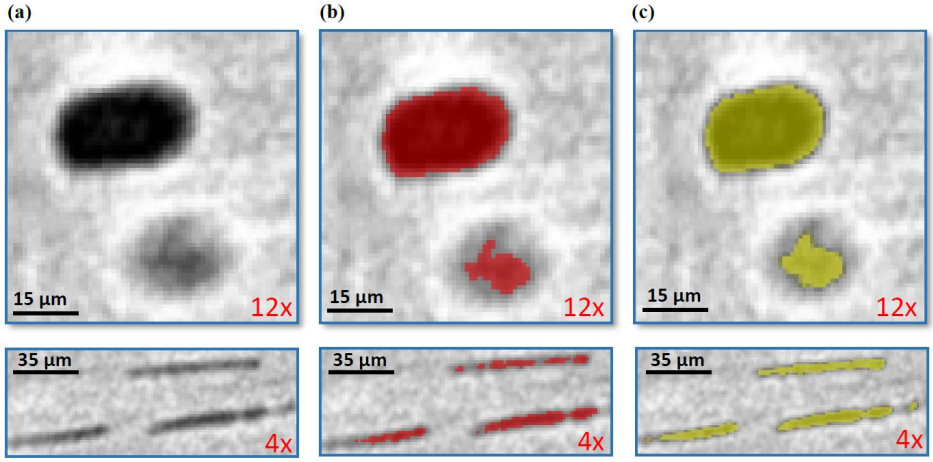

FIGURE 5. Zoomed version of Region 1 and 2: Input data slice (a), Otsu thresholding (red) segmentation results overlaid on input data slice (b), prediction (yellow) overlaid on input data slice (c). The detail images showcase a closer view of the edge region marked in blue, magnified 12 times and 4 times respectively.

under-segmented, especially with long and thin (Figure 3c pores and small pores (see Figure 3b. The neural network predictions are performing better than Otsu thresholding as seen by visual inspection.

3D visualization of the sub-volume and its prediction with overlaid Otsu thresholding is shown in Figure 3a. Differences between Otsu thresholding and predictions are clearly visible in the zoomed region in the 3D image. The predicted area covers most of the matching Otsu thresholding region. According to the 2D and 3D visualizations of the sub-volume shown in Figure 3, both show that the results from Otsu thresholding are under-segmented when compared to predictions of the neural network model.

5.2. Prediction and visualization of sample 2

This section presents the prediction outcome for Sample 2 (see Figure 4). The goal is here to show that the model is also able to effectively segment unseen data. Train data and test data have an isotropic resolution of 3.3 μm. In Figure 4c, yellow colors (prediction) show that long thin pores are segmented much better with the neural network. Otsu thresholding is not as effective for this kind of pore segmentation as the neural network. In Figure 4b, Region 2 clearly shows that Otsu thresholding produces under-segmented results.

The qualitative result of prediction is similar to the sub-volume segmentation presented in section Section 5.1. Sample 2 has many different shapes of pores such as rounded pores (big and small), long thin and some big complex pores. Segmentation results differ substantially between the Otsu technique and the neural network prediction depend on the shapes of the pores.

Region 1 and 2 of Figure 4 are shown in a magnified version in Figure 5. The original slice images are transparently overlaid with the Otsu thresholding segmentation and the neural network prediction. Region 1 contains two pores with different characteristics: one is clearly visible, while the other one is only barely visible. Segmentations are almost the same and reasonably good for the clearly visible pore in both methods. For barely visible pores, it is hard to evaluate which method corresponds better with reality by visual inspection. There is not a sharp edge between pore and background transitions and thus it is also hard to make decisions for the network. Even so, it is

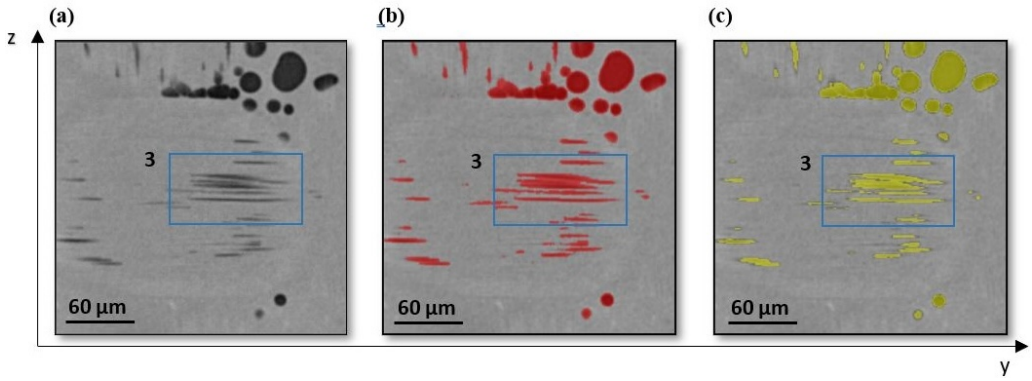

Figure 6. Input data slice from Sample 3. (a), and (b), (c) respectively show 2D slices of Otsu thresholding and prediction overlaid on an input volume slice from Sample 2 (red= Otsu thresholding, yellow=predictions).

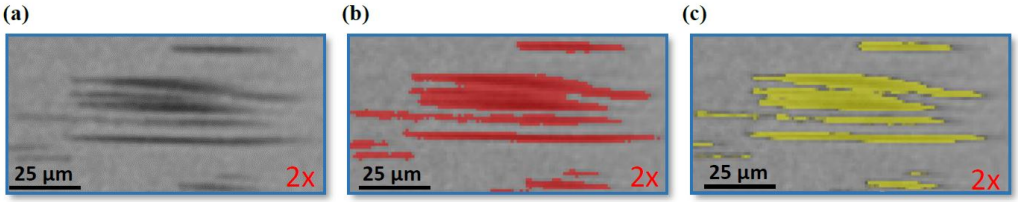

Figure 7. Zoomed version of Region 3: Input data slice (a), Otsu thresholding overlaid on input data slice (b), prediction overlaid on input data slice (c) (red= Otsu thresholding, yellow=predictions). The detailed images display a 2x zoom of the edge region highlighted in blue.

also hard to decide which area should be segmented as pore for a human.

5.3. Prediction and visualization of sample 3

The model trained on 3.3 μm data is also successfully working on datasets with different resolution. To show that, Sample 3 is segmented in open_iA with the neural network and Otsu thresholding method, the resulting segmentation outcomes are presented in Figure 6. However, there are some limitations with applying the neural network on the different resolution data: the pores (long and thin) are barely visible, much less visible than in the datasets with 3.3 μm resolution, and the transition is more blurred between pore and background.

The results show that the prediction of the neural network is also successfully working for the segmentation of different shapes of pores on different resolutions (see Figure 6). Another result is that the Otsu thresholding over-segments pores (see Figure 7b).

The main challenge in this scenario is the segmentation of long and thin pores, particularly in cases where they are in close proximity to one another (see Figure 7). Proper segmentation is challenging for the network model and other methods. Nevertheless, results are impressive and promising for future improvements.

6. Conclusions

This paper demonstrates that the modified 3D U-Net architecture achieves satisfactory performance, even when trained on a limited training dataset. Through visual inspection and a comparative assessment with the findings, the modified 3D U-Net performs better in certain areas compared to Otsu thresholding. Significantly, the generated model with 3 μm resolution also demonstrated a satisfactory performance on the datasets with 10 μm resolution. It is important to note that during training, only the 3 μm resolution (Sample 1) was utilized. Quality assessment and comparisons in this workflow are conducted through manual visual inspection.

For future work, we plan to measure the quality numerically by using XCT simulated data on CFRP. However, to measure the quality of estimations and the performance of the deep learning model, the generation of accurate ground truth is a necessity.

Therefore, we will improve our reference segmentation by incorporating both XCT images segmented by human experts and XCT simulation techniques. Another important point is the uncertainty determination for generation of XCT images (radiographs), reconstruction, segmentation and deep learning training. Each of these steps can influence the final image segmentation or estimation. To improve segmentation accuracy, uncertainty sources and their levels have to be identified. Additionally, we intend to explore the assessment of algorithm reliability through the creation of artificial XCT images containing known pore sizes. Moreover, we will investigate the impact of introducing noise or image blurring on the algorithm's sensitivity to various side effects.

Acknowledgements

This research was co-financed by the European Union H2020-MSCA-ITN-2020 under grant agreement no. 956172 (xCTing).

References

[1] C. Heinzl, S. Stappen. Star: Visual computing in materials science. *Computer Graphics Forum* **36**:647–666, 2017. https://doi.org/10.1111/cgf.13214

[2] D. Cireşan, U. Meier, J. Schmidhuber. Multi-column deep neural networks for image classification. *Proceedings / CVPR, IEEE Computer Society Conference on Computer Vision and Pattern Recognition IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 2012. [2023-06-30]. https://doi.org/10.48550/ARXIV.1202.2745

[3] J. Ba, V. Mnih, K. Kavukcuoglu. Multiple object recognition with visual attention. *arXiv* 2014. [2023-06-30]. https://doi.org/10.48550/ARXIV.1412.7755

[4] J. Ker, L. Wang, J. Rao, T. Lim. Deep learning applications in medical image analysis. *IEEE Access* **6**:9375–9389, 2018. https://doi.org/10.1109/ACCESS.2017.2788044

[5] M. Yosifov. *Extraction and Quantification of Features in XCT Datasets of Fibre Reinforced Polymers using Machine Learning Techniques*. Master's thesis, Umeå University, Department of Computing Science, 2020.

[6] M. Yosifov, M. Reiter, S. Heupl, et al. Probability of detection applied to x-ray inspection using numerical simulations. *Nondestructive Testing and Evaluation* **0**(0):1–16, 2022. https://doi.org/10.1080/10589759.2022.2071892

[7] M. Yosifov, P. Weinberger, M. Reiter, et al. Defect detectability analysis via probability of defect detection between traditional and deep learning methods in numerical simulations. *e-Journal of Nondestructive Testing* **28**, 2023. https://doi.org/10.58286/27716

[8] S. Ray, R. H. Turi. Determination of number of clusters in k-means clustering and application in colour image segmentation. In *International Conference on Advances in Pattern Recognition and Digital Techniques*, pp. 137–143. 1999.

[9] J. A. Hartigan, M. A. Wong. Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society Series C (Applied Statistics)* **28**(1):100–108, 1979. https://doi.org/10.2307/2346830

[10] L. J. Belaid, W. Mourou. Image segmentation: a watershed transformation algorithm. *Image Analysis & Stereology* **28**(2):93–102, 2009. https://doi.org/10.5566/ias.v28.p93-102

[11] J. Kastner, B. Plank, D. Salaberger, J. Sekelja. Defect and porosity determination of fibre reinforced polymers by x-ray computed tomography. *2nd International Symposium on NDT in Aerospace* 2010.

[12] G. Rao, B. Plank, J. Kastner. Comparison of different segmentation methods for porosity evaluation in cfrp-reference samples with real porosity samples. In *Conference on Industrial Computed Tomography*, pp. 1–8. 2016.

[13] B. Fröhler, J. Weissenböck, M. Schiwarth, et al. open_iA: A tool for processing and visual analysis of industrial computed tomography datasets. *Journal of Open Source Software* **4**(35):1185, 2019. https://doi.org/10.21105/joss.01185

[14] M. Sezgin, B. Sankur. Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging* **13**:146–168, 2004. https://doi.org/10.1117/1.1631315

[15] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics* **9**(1):62–66, 1979. https://doi.org/10.1109/TSMC.1979.4310076

[16] A. Reh, B. Plank, J. Kastner, et al. Porosity maps–interactive exploration and visual analysis of porosity in carbon fiber reinforced polymers. *Computer Graphics Forum* **31**:1185–1194, 2012. https://doi.org/10.1111/j.1467-8659.2012.03111.x

[17] O. Ronneberger, P. Fischer, T. Brox. U-Net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 234–241. Springer, Cham, 2015. https://doi.org/10.1007/978-3-319-24574-4_28

[18] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, et al. 3D U-Net: Learning dense volumetric segmentation from sparse annotation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 424–432. Springer, Cham, 2016. https://doi.org/10.1007/978-3-319-46723-849

[19] J. Walsh, A. Othmani, M. Jain, S. Dev. Using u-net network for efficient brain tumor segmentation in mri images. *Healthcare Analytics* **2**:100098, 2022. https://doi.org/https://doi.org/10.1016/j.health.2022.100098

[20] G. van Rossum, P. D. Team. *Python 3.6 Language Reference*. Samurai Media Limited, 2016.

[21] Keras. F. Chollet and others. *Keras: Deep Learning for humans*. 2015. https://keras.io.

[22] *TensorFlow: Large-scale machine learning on heterogeneous systems*. 2015. https://tensorflow.org.

[23] ONNX. *Open Neural Network Exchange: The open standard for machine learning interoperability*. 2017. https://onnx.ai.

[24] Autonomio Talos [Computer software]. 2020. http://github.com/autonomio/talos.

[25] D. P. Kingma, J. Ba. Adam: A method for stochastic optimization. In *International Conference for Learning Representations*. 2017. [2023-06-30]. https://doi.org/10.48550/arXiv.1412.6980

[26] T. Sørensen. A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on danish commons. *Kongelige Danske Videnskabernes Selskab, Biologiske Skrifter* pp. 1–34, 1948.