Detecting micro fractures: A comprehensive comparison of conventional and machine-learning based segmentation methods

Dongwon Lee[1], Nikolaos Karadimitriou[1], Matthias Ruf[1], and Holger Steeb[1,2]

[1]Institute of Applied Mechanics (CE), Pfaffenwaldring 7, University of Stuttgart,70569, Germany
[2]Stuttgart Center for Simulation Science, Pfaffenwaldring 5a, University of Stuttgart,70569, Germany

Correspondence: Dongwon Lee (dongwon.lee@mechbau.uni-stuttgart.de)

Abstract. Studying porous rocks with X-Ray Computed Tomography (XRCT) has been established as a standard procedure for the non-destructive characterization of flow and transport in porous media. Despite the recent advances in the field of XRCT, various challenges still remain due to the inherent noise and imaging artefacts in the produced data. These issues become even more profound when the objective is the identification of fractures, and/or fracture networks. One challenge is the limited contrast between the regions of interest and the neighboring areas, which can mostly be attributed to the minute aperture of the fractures. In order to overcome this challenge, it has been a common approach to apply various digital image processing steps, such as filtering, to enhance the signal-to-noise ratio. Additionally, segmentation methods based on threshold/morphology schemes have been employed to obtain enhanced information from the features of interest. However, this workflow needs a skillful operator to fine-tune its input parameters, and the required computation time significantly increases due to the complexity of the available methods, and the large volume of an XRCT data-set. In this study, based on a data-set produced by the successful visualization of a fracture network in Carrara marble with μXRCT, we present the results from five segmentation methods, three conventional and two machine learning-based ones. The objective is to provide the interested reader with a comprehensive comparison between existing approaches, while presenting the operating principles, advantages and limitations, to serve as a guide towards an individualized segmentation workflow. The segmentation results from all five methods are compared to each other in terms of quality and time efficiency. Due to memory limitations, and in order to accomplish a fair comparison, all the methods are employed in a 2D scheme. The output of the 2D U-net model, which is one of the adopted machine learning-based segmentation methods, shows the best performance regarding the quality of segmentation and the required processing time.

1 Introduction

The adequate characterization and knowledge of the geometrical properties of fractures has a strong constructive impact on various disciplines in engineering and geosciences. In civil engineering, detecting fractures on building materials Yamaguchi and Hashimoto (2010); Xing et al. (2018), or road pavements Nguyen et al. (2009) is important for maintenance and inspection. In the field of reservoir management, due to the fact that fractures strongly influence the permeability and porosity of a reservoir, their properties, like network connectivity and mean aperture, play a decisive role Berkowitz (1995); Jing et al.

(2017); Karimpouli et al. (2017); Suzuki et al. (2020) in the production and conservation processes of resources, such as methane Jiang et al. (2016) and shale oil Dong et al. (2019); van Santvoort and Golombok (2018). Additionally, the geometry of the fracture network is an equally important property that needs to be addressed in our efforts to properly characterize the stresses developing between tectonic plates Lei and Gao (2018) which, eventually, have a strong effect on the permeability of a reservoir Crawford et al. (2017); Huang et al. (2019). As one of the most straight-forward and effective ways of investigating such fracture properties, 3D visualization and consequent efficient extraction of various fractures, or fracture network, spatial properties, like length, aperture, tortuosity etc., has been increasingly employed.

X-Ray Computed Tomography (XRCT) has been adopted as a non-destructive method to monitor and quantify such spatial properties, and it has become a standard procedure in laboratory studies Weerakone and Wong (2010); Taylor et al. (2015); De Kock et al. (2015); Saenger et al. (2016); Halisch et al. (2016); Chung et al. (2019). More specifically, lab-based micro-X-Ray Computed Tomography (μXRCT) offers the ability to visualize features on the pore-scale at micrometer resolution. However, as with every other visualization method, there are some drawbacks which are inherent to the method itself. This visualization method is based on the interaction of an, otherwise, opaque sample with a specific type of radiation. Occasionally, there is a low contrast between distinct features, either belonging to different materials, or having spatial characteristics which are close to the resolution of the method itself. Such an example is the distinction between the solid matrix and an existing fracture, which commonly has a very small aperture Lai et al. (2017); Kumari et al. (2018); Zhang et al. (2018); Furat et al. (2019); Alzubaidi et al. (2022). Consequently, a lot of effort has to be made to "clean" the acquired data from potential noise, independently of its source Frangakis and Hegerl (2001); Behrenbruch et al. (2004); Christe et al. (2007). As a means to tackle this issue, there have been many studies focusing on the enhancement and optimization of digital image processing with the use of filtering methods Behrenbruch et al. (2004); Coady et al. (2019) which could reduce the inherent noise of the image-based data while keeping the features of interest intact. In this way "clean" images, meaning with a low signal-to-noise ratio, can be obtained with an enhanced contrast between the features of interest and their surroundings.

In the work of Poulose (2013), the authors classified the available filtering methods. Their classification can be narrowed down into two major categories, namely the linear and the non-linear ones Ahamed et al. (2019). The latter type of filtering methods makes use of the intensities of the neighboring pixels in comparison to the intensities of the ones of interest. There are two well-known filtering methods of this type; the first one is commonly referred to as "anisotropic diffusion" Perona and Malik (1990), where a blurring scheme is applied in correlation to a threshold gray-scale value. The other is commonly referred to as the "non-local means filter" Buades et al. (2011). This method uses mean intensities to compute the variance (similarity) at the local area surrounding a target pixel. As an improvement to this filter, the "adaptive manifolds non-local means filter" Gastal and Oliveiray (2012) has been recently developed to reduce the data noise with the help of an Euclidean-geodesic relation between pixel intensities and the calculated local mean intensities of the created manifolds, accounting for the grayscale value as a separate dimension for each pixel. On the other hand, the linear type of filtering method makes use only of fixed (pixel-oriented) criteria. This, in practice, means that it simply blurs the image without distinguishing between generic noise and the boundaries of an object of interest. Although there are no prerequisites in the selection of the appropriate

noise filtering method, the non-linear type of filtering is more frequently employed with XRCT due to its edge conserving characteristic Sheppard et al. (2004); Taylor et al. (2015); Ramandi et al. (2017).

The extraction of features of interest in images ought to be carried out for further evaluations. This can be achieved by performing image segmentation, which is a clear pixel-wise classification of image objects. For instance, the classification of the pixels belonging to the pore space or the solid matrix, results in the effective estimation of porosity. Conventionally, image segmentation can take place simply by applying a certain threshold to classify the image corresponding to pixel intensity. The most well-known approach based on this scheme is the "Otsu method" Nobuyuki Otsu (1979). In this segmentation scheme, histogram information from intensities is used to return a threshold value which classifies the pixels into two classes, one of interest and the rest. As an advanced intensity-based approach for multi-class applications, the K-mean method is often used Salman (2006); Dhanachandra et al. (2015); Chauhan et al. (2016); MacQueen (1967). Dhanachandra et al. (2015) showed their effectively segmented results of malaria-infected blood cells with the help of this method. However, this intensity-based segmentation approach could not be a general solution for XRCT data due to lab-based XRCT inherent "beam-hardening" effects, which creates imaging artefacts. These artifacts are caused by polychromatic X-rays (more than one wavelengths) which induce different attenuation rates along the thickness of a sample Ketcham and Hanna (2014). As another common approach of segmentation, the methods based on "edge-detection" are often adopted Su et al. (2011); Bharodiya and Gonsai (2019). In this scheme one could utilize the contrast of intensities within the images which would eventually outline the features of interest Al-amri et al. (2010). Based on the assumption that the biggest contrast would be detected where two differently gray-scaled features meet, there are many different types of operators in order to compute this contrast, namely the Canny Canny (1986), Sobel Dura and Hart (1973) and Laplacian Marr and Hildreth (1980) ones. However, there is an inherent limitation to apply the method to XRCT applications, since the image has to have a high signal-to-noise ratio, which is hardly ever the case when fractures are involved.

Voorn et al. (2013), adopted a Hessian matrix-based scheme and employed "Multi-scale Hessian Fracture" (MHF) filtering to enhance the narrow planar shape of fractures in 3D XRCT data from rock samples, after applying a Gaussian filter to improve fracture visibility. This resulted in the enhancement of fractures of various apertures. The authors easily binarized their images with the help of emphasized responses with structural information obtained from the Hessian matrix. This scheme was also employed in the work of Deng et al. (2016), where they introduced the "Technique of Iterative Local Thresholding" (TILT) in order to segment a single fracture in their rock image data-sets. After acquiring an initial image just to get a rough shape of the fracture by using a threshold or MHF, they applied the *Local threshold* method Li and Jain (2009), which sets a threshold individually for every pixel (cf. 2.3.1), repeatedly to narrow down the rough shape into a finer one, until its boundary reached the outline of the fracture. Despite the successful segmentation of their data-set, some challenges still remained for the segmentation of fracture networks, as expressed in the work of Drechsler and Oyarzun Laura (2010). The authors compared the famous Hessian-based algorithms from Sato et al. (1997), Frangi et al. (1998), and Erdt et al. (2008) with each other. In their study they showed that these schemes are not too effective to classify the intersections of the features of interest. Additionally, in the case of MHF, due to the used Gaussian filter which blurs the edges of a fracture, the surface roughness of the fractures

would most probably not be identified. Also, the segmentation results varied significantly corresponding to the chosen range of standard deviations Deng et al. (2016).

As a more advanced segmentation approach, the watershed segmentation method was introduced by Beucher and Meyer (1993). It allows to obtain individually assigned groups of pixels (features) as a segmentation result. This can be accomplished by following the following steps:

1. Find the local intensity minimums for each one of the assigned groups of pixels.

2. At each local minimum, the assignment of the nearby pixels to individual label groups is performed according to the magnitudes of the corresponding intensity gradients. A distance map, where the distance is computed from all non-zero pixels of a binarized image, could be used instead of an image which contains intensity values Maurer et al. (2003).

3. The above mentioned procedures ought to be performed repetitively until there are no more pixels to be assigned.

Taylor et al. (2015), based on the watershed method, were able to successfully visualize a segmented void space of XRCT scanned granular media (sand) which also had a narrow and elongated shape, similar to a fracture. Ramandi et al. (2017) demonstrated the segmentation result of coal cleats by applying a more advanced scheme based on the watershed method, the Converging Active Contour (CAC) method, initially developed by Sheppard et al. (2004). Their method combined two different schemes; one was the watershed method itself, and the other was the active contouring method Caselles et al. (1993), which is commonly used for finding the continuous and compact outline of features. However, there was still a fundamental limitation to cope with, to directly segment a fracture with the introduced method since the method requires a well-selected initial group of pixels, while it is hard to define those only with a threshold due to the low contrast of the feature Weerakone and Wong (2010). The same authors Ramandi et al. (2017) applied a saturation technique with an X-ray attenuating fluid. By performing this they were able to enhance the visibility of the pores and fractures within the XRCT data.

Instead of dealing with the image with numerically designed schemes, in recent years and in various fields, like remote sensing Li et al. (2019); Cracknell and Reading (2014), medical imaging Singh et al. (2020) and computer vision Minaee et al. (2020), image segmentation using machine learning techniques has been being robustly studied as a new approach due to its well-known benefits, like flexibility to all types of applications and less demanding user intervention. Based on this advantages, Kodym and Španěl (2018) introduced a semi-automatic image segmentation workflow with XRCT data by using the Random Forest algorithm Amit and Geman (1997) which statistically evaluated the output from decision trees (cf. 2.3.4). Karimpouli et al. (2019) demonstrated cleats segmentation results from XRCT with a coal data sample by applying the Convolutional Neural Networks (CNNs) approach, which is another machine learning scheme. The CNN is one of the deep artificial neural network models which consists of multi-layers with convolutional and max-pooling layers Albawi et al. (2018); Palafox et al. (2017). The former type of layers is employed to extract features out of an input image and the latter one is employed to down-sample the input by collecting the maximum coefficients which are obtained from the convolutional layers. Based on the CNNs architecture, Long et al. (2017) demonstrated the Fully Convolutional Networks (FCN) model. The model consists of an encoding part, which is of the same structure as the CNNs, and a decoding part, which up-samples the layers using

a de-convolutional operation (cf. 2.3.5). The authors also introduced a skip connection scheme, which is concatenating the previously maximum collected layers into the decoding part in order to enhance the prediction of the model.

With the help of those CNN architectures, exploratory models were proposed and studied for the detection and segmentation of cracks from image data. In the work of Rezaie et al. (2020), the U-net and VGG-16 combined architecture, called TernausNet, were employed. The VGG-16 Simonyan and Zisserman (2014) is also a CNN-type architecture consisting of sequential convolutional layers with multiple pooling layers, which can extract features hierarchically. Fei et al. (2020), introduced CrackNet-V, which utilizes a VGG-16 like architecture. The CrackNet-V contains only sequential convolutional layers without the pooling layers. The authors showed that the model was beneficial to detect fine and shallow cracks. Recently, Li et al. (2022), proposed the SoUNet (Side-output U-net) model. The model consists of two parts; one is a conventional U-net part which performs a segmentation task. The other is the side-output part which detects edges of segmented results. By utilizing both, they improved their segmentation result.

Despite these promising results from the use of machine learning schemes, there is still room for improvement. In a machine learning-based scheme, apart from improving the architecture of the training model, which significantly affects the efficiency and output of the model Khryashchev et al. (2018), one major issue is the provision of a ground-truth data-set. In most applications Shorten and Khoshgoftaar (2019); Roberts et al. (2019); Zhou et al. (2019); Alqahtani et al. (2020) this was performed based on subjective manual annotation which would consume noticeable time and effort of an expert. This manual annotation is hardly the best approach, especially in our application which contains sophisticated shapes of fracture networks and low contrast features due to the low density of air within the small volume of a fracture Furat et al. (2019).

Eventually, the introduced various segmentation techniques were used in order to characterize fractures, i.e., crack density, length, orientation and aperture. In the works of Fredrich and Wong (1986),Healy et al. (2017), the authors binarized their high resolution images by manually annotating the fracture traces. Griffiths et al. (2017) classified fractures by applying the watershed segmentation method. Arena et al. (2014) segmented fractures by using the Trainable Weka platform Arganda-Carreras et al. (2017). Given that the estimation of such parameters in the studies relied on binarized images, by following differently adopted segmentation methods, it is important to compare the effectiveness and efficiency of the relevant segmentation methods.

With the use of a data-set taken from the imaging of a real fractured porous medium, and for comparative purposes, three different segmentation techniques were adopted in this work, based on conventional means, such as the Sato, Local threshold, and active contouring method, including filtering and post-processing. Two different machine learning models were employed for this comparison as well. The well-known U-net Ronneberger et al. (2015), as well as the Random Forest model, were evaluated, with the Random Forest model being supported in Trainable Weka Arganda-Carreras et al. (2017). Due to memory limitation for the U-net model we had to dice the 2D images to smaller 2D tiles and compute sequentially. The computed prediction tiles were merged together at a later stage to reconstruct the 2D image. In order to train the adopted model, we provided a training data-set which was obtained by means of a conventional segmentation method. Note that only a small portion of the results (60 slices out of 2140 images) was used as a training data-set. For a fair comparison between methods, all segmentation schemes were applied on 2D data. Via comparison, we were able to identify that the U-net segmentation model

outperformed the others in terms of quality of the output and time efficiency. Some defects which appeared in the provided truth data by conventional segmentation means were smeared out in the predictions of the model.

2 Materials and methods

2.1 Sample preparation

The segmentation approach presented here is based on a μXRCT data-set of a thermally treated cylindrical Bianco Carrara marble core sample with a diameter of $5\,\mathrm{mm}$ and a length of $10\,\mathrm{mm}$. Bianco Carrara marble is a crystalline rock, consisting of about $98\,\%$ calcite ($CaCO_3$) Pieri et al. (2001) and is a frequently used material in experimental rock mechanics Delle Piane et al. (2015). In combination with mechanical or thermal treatment, the virgin state can be modified to achieve different characteristics of micro-fractures within the sample Lissa et al. (2020, 2021); Pimienta et al. (2019); Sarout et al. (2017); Delle Piane et al. (2015); Peacock et al. (1994).

The considered sample was extracted from a bigger cylindrical core sample which was subjected to a thermal treatment beforehand. This treatment included:

1. Heating-up from room temperature (20°C) to 600°C with a heating rate of $3\,\mathrm{K\,min^{-1}}$.
2. Holding this temperature for $2\,\mathrm{h}$ to ensure a uniform temperature distribution in the entire sample.
3. Quenching the sample in a water basin at room temperature ($20\,^{\circ}\mathrm{C}$).

The porosity of the raw material, before any treatment, was $0.57\,\%$ measured with mercury porosimetry. The porosity of the thermally treated sample was obtained from measurements of volume changes before/after quenching, and was found to be around 3 %.

The scanning of the extracted sample was performed in a self-built, modular, cone-beam µ-XRCT system, thoroughly described in Ruf and Steeb (2020b) (please refer to Figure 1). The data-set along with all meta data can be found in Ruf and Steeb (2020a). For more technical details, please refer to A

2.2 Noise reduction

In order to conserve the edge information while reducing artefacts, the Adaptive Manifolds Non-Local Mean (AMNLM) method was adopted. As shown in Fig. 2, this procedure is an essential step to obtain good segmentation results with some of the conventional segmentation methods. The adopted filtering method was applied with the help of the commercial software Avizo© (2019.02 ver.). The adopted input parameters of the filter in this study were spatial standard deviation: 5, intensity standard deviation: 0.2, size of search window: 10 and size of local neighborhood: 3. The numerical description of this filtering method in 2D is explained in B.

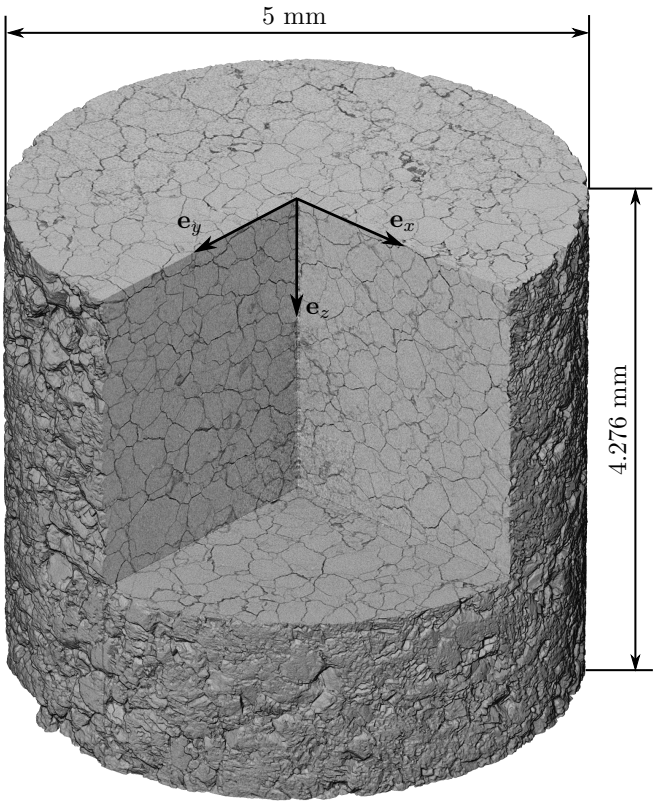

Figure 1. Illustration of the underlying raw µXRCT dataset of a thermally treated Bianco Carrara marble sample taken from Ruf and Steeb (2020a). The dataset along with all meta data can be found in Ruf and Steeb (2020a), https://doi.org/10.18419/darus-682; licensed under a Creative Commons Attribution (CC BY) license.

2.3 Segmentation methods

The adopted segmentation methods are introduced and discussed in this chapter. As conventional schemes, the Local threshold, Sato, and Active contouring methods were chosen. Note that the adopted conventional segmentation methods had to be employed with noise-reduced images and also required certain workflows (cf. Fig. 3). The implemented code with detailed parameters is available via the link in the "code availability" section.

2.3.1 Local Threshold

The Local Threshold method is a binarization method, also known as the adaptive threshold method. With this method one can create a threshold map which has the same size as this of the input image for binarization, corresponding to the intensities within the local region. The output of this procedure is a map which includes individual thresholds (local threshold) for each

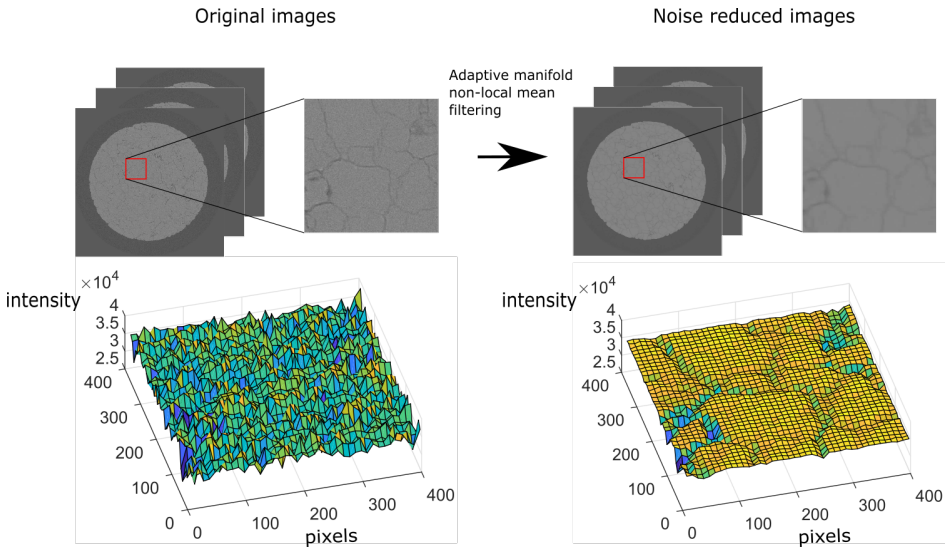

Figure 2. Filtering effects. The images shown above are 2D partial images from the original image (left side) and noise reduced image (right side) by applying 3D adaptive manifolds non-local mean filtering Gastal and Oliveiray (2012). The red box areas are zoomed (400 x 400 pixels). The intensity surfaces for both zoomed images are shown below.

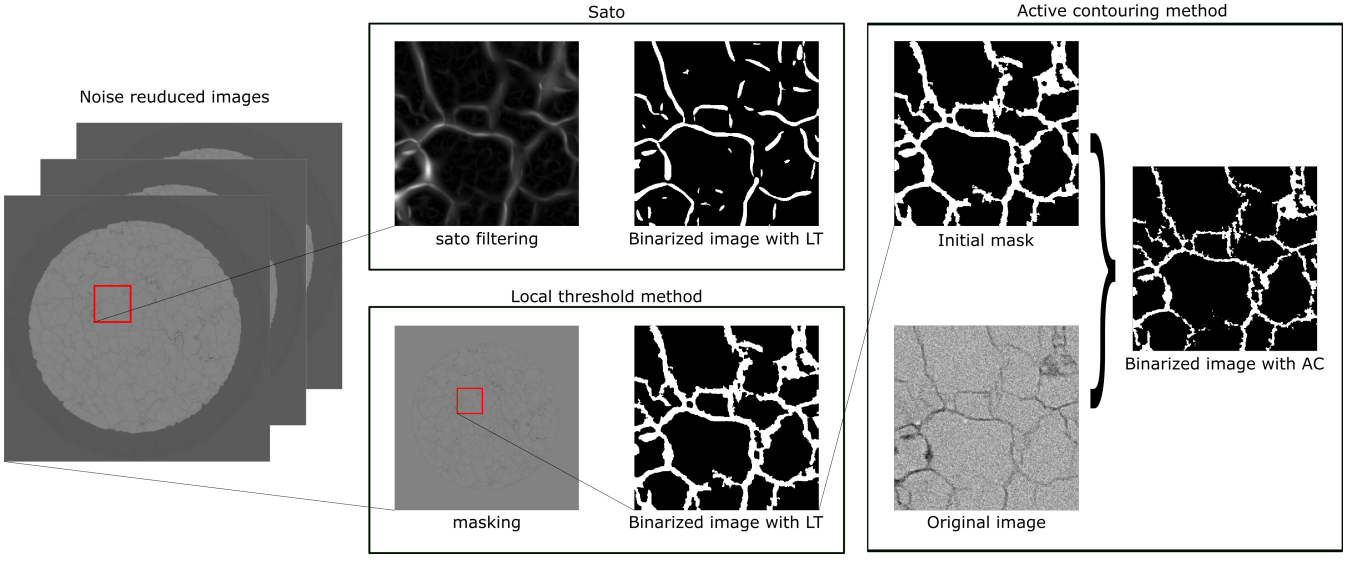

Figure 3. Workflows of the adopted conventional methods: The workflows of segmentation with Sato, Local threshold (LT) method and Active contouring (AC) method are shown with sample images from each step. The magnified region is marked in red.

pixel. From the obtained threshold map, binarization is performed by setting the pixels which contain higher intensities than the corresponding location at the threshold map as logical truth, and the others to be logical false.

For our application, this method was applied on noise-reduced data in order to minimize any artefacts. Smoothing was performed in order to obtain a threshold map with the help of Gaussian filtering Eq. C.1.

Parameters like the offset and the radius of the adjacent area were empirically selected. This procedure was performed with the `threshold_local` function from the Python *skimage* library. Small artefacts which remained were removed by employing the `remove_small_object` function from the same library. In order to cope with the modus operandi of the method, before applying the method the outer part of the sample was filled with a mean intensity value extracted from a region of the image where a fracture was not contained.

2.3.2 Sato filtering

The Sato filtering method is one of the most well-known filtering methods. The potential of this filter in finding structural information is well-matching for the segmentation of a fracture, which often has a long and thin string-like shape in two-dimensional images. An elaborate analysis of the fundamentals of the filtering method can be found in the appendix C.

2.3.3 Active contouring

In image processing, the active contouring method is one of the conventional methods applied in order to detect, smoothen, and clear the boundaries of the features of interest Caselles et al. (1993). Instead of the conventional active contouring approach, which requires distinguishable gradient information within an image, we adopted the Chan-vese method Chan and Vese (2001) which makes use of the intensity information. This has an advantage especially for XRCT applications, as in this case, due to the low signal-to-noise ratio data. An elaborate analysis of the fundamentals of the filtering method can be found in the appendix D.

2.3.4 The Random Forest scheme

The Random Forest method Amit and Geman (1997) is using statistical results from multiple classifiers, commonly referred to as decision trees. A decision tree, which is a single classifier, is made of decision nodes and branches which are bifurcated from the nodes. In image segmentation, each node represents an applied filter, such as Gaussian, Hessian, etc., in order to create branches which represent the responses of the filter. These applied filters are also called feature maps, or layers Arganda-Carreras et al. (2017). In our study, the chosen feature maps, in order to create decision trees, were edge detectors (Sobel filtering, Gaussian difference and Hessian filtering), noise reduction filters (Gaussian blurring) and membrane detectors (membrane projections). The model was trained with the Trainable Weka platform Arganda-Carreras et al. (2017). The first slice of scanned data at the top side of the sample was used for training with the corresponding result obtained from the active contouring method (cf. 2.3.3). On the trained model, diced data E1 was applied and its corresponding predictions were merged as explained in E3.

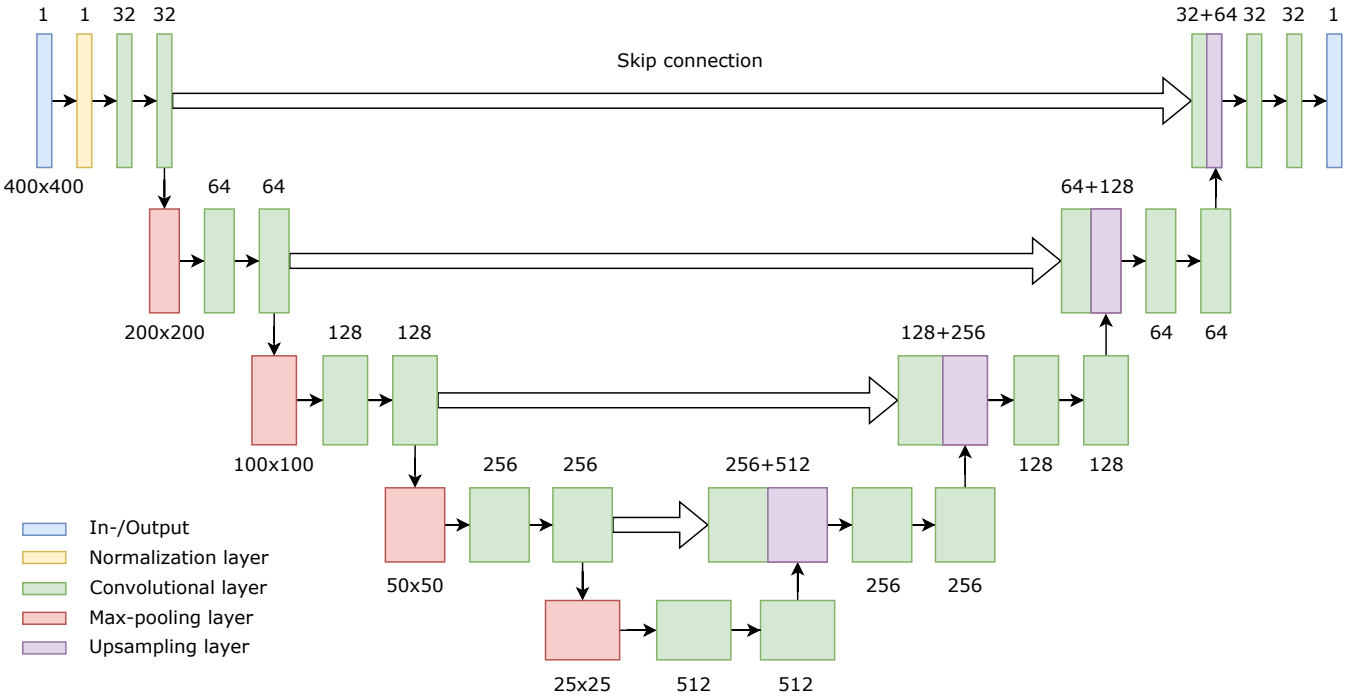

Figure 4. The schematic of the used 2D u-net model. The input image was down-/upsampled with the help of Max-pooling and Upsampling layers .The features were extracted by convolutional layers. Those extracted features during down-sampling were concatenated to corresponding up-sampled features (skip connection).

2.3.5 The U-net model

The U-net model, one of the Convolutional Neural Network (CNN) architectures, proposed by Ronneberger et al. (2015) was applied to segment the fractures. This model finds optimized predictions between input and target by supervising the model with a ground truth (GT). The GT was chosen from the segmentation results from the active contouring combined method 2.3.3 while the input data was raw images which were not treated with any noise reduction techniques.

The model makes use of repeating down-scaling of the input image with the help of max-pooling layers, and up-scaling with a de-convolutional layer. Additionally, before and after each of the up-/down-scaling layers, the convolutional layers which extract the feature maps are used with an activation function which introduces a non-linearity into the model. Each of the extracted and down-scaled features are concatenated to the same size of the up-scaled features, in order to force the output pixels to be located at reasonable locations (see Figure 4).

The images which were used for training were cropped into small tiles. This cropping had to be performed due to memory limitations (cf. E1). This made each tile to contain varying characteristics which also introduced a difficulty to train a model while taking these different characteristics into account. In order to cope with this problem, the normalization layer, which zero-centers the histogram of the data in the range of -1 to 1, was used previously to insert the data into the model. This was

essential to deal with the tiles with ring shape artefacts, which were induced during the reconstruction of the XRCT data. Without this layer the model might not had been able to identify the fractures in such regions due to the different profiles they bear in comparison to the other regions, since the artefacts affect the histogram and shape of the features.

For the sake of accuracy, the data augmentation technique was applied on the training data-set. This allowed us to enrich the training data-set by employing a modification to the data; thus, the model could be trained with sufficient data of different variations. This contributed to the prevention of overfitting, which made the model capable of dealing only with a specific case. In our application, we varied the brightness of the training data. Thus, the model was able to be trained by data with variation. This was necessary to get good predictions from all cropped tiles.

The implementation of the model was done by means of the keras library in Python Chollet et al. (2015) which includes useful functions for CNN such as convolutional, normalization layers, etc. With the given model details, the input data was separated into small pieces (*Splitting*), trained with GT data (*Training*) and finally it was merged back to the size of the original image (*Merging*). For more information on these processes, please refer to E.

3 Results and discussion

We successfully conducted the segmentation of the data acquired by the μ-XRCT scanning of a quenched Carrara marble sample, with all the above-mentioned segmentation schemes. By performing 2D segmentation on each slice of scanned data and stacking the segmentation results, the full 3D segmented fracture network was able to be acquired for the entire scanned domain.

Parameters like standard deviation, offset, smoothing factors for active contouring etc., of the methods based on conventional schemes, such as the Local threshold 2.3.1, Sato 2.3.2 and active contouring 2.3.3, were fine-tuned empirically. It was time consuming to decide these parameters, since we had to find the appropriate parameters which would give the best results along the entire data-stack, while some slices contained different histogram profiles, or artefacts. Since the conventional schemes showed a tendency to be very sensitive to artefacts, and the corresponding outputs varied significantly with the applied parameters, applying noise reduction (2.2) with carefully chosen parameters was a fundamental step before applying any conventional method, in order to minimize the error and obtain good results. In the case of machine learning-based segmentation, the applied methods had the ability to classify the fracture networks within the original data. Since these machine learning-based algorithms do not require any noise reduction, this was one of the biggest advantages which not only reduced the computation time, but also reduced the input parameters tuning time.

In Figure 5, the segmentation results of the top-viewed specimen (x-y-planar size of (2940 $\times$ 2940 pixels) are shown. We were able to acquire sufficiently good segmented results for the whole x-y-plane despite the beam hardening effect at the edges, and the ring artefacts in the middle of the sample. We selected the 1000th slice of the scanned data on the z-axis, which is located at the middle of the sample. Note that this slice was never used as training data for the machine learning-based methods. In Figure 6, the segmentation results from the side-viewed sample (x-z-planar size of 2940 $\times$ 2139 pixels) are demonstrated. These images were obtained by stacking the 2D segmentation results from x-y-planar images along the z-axis

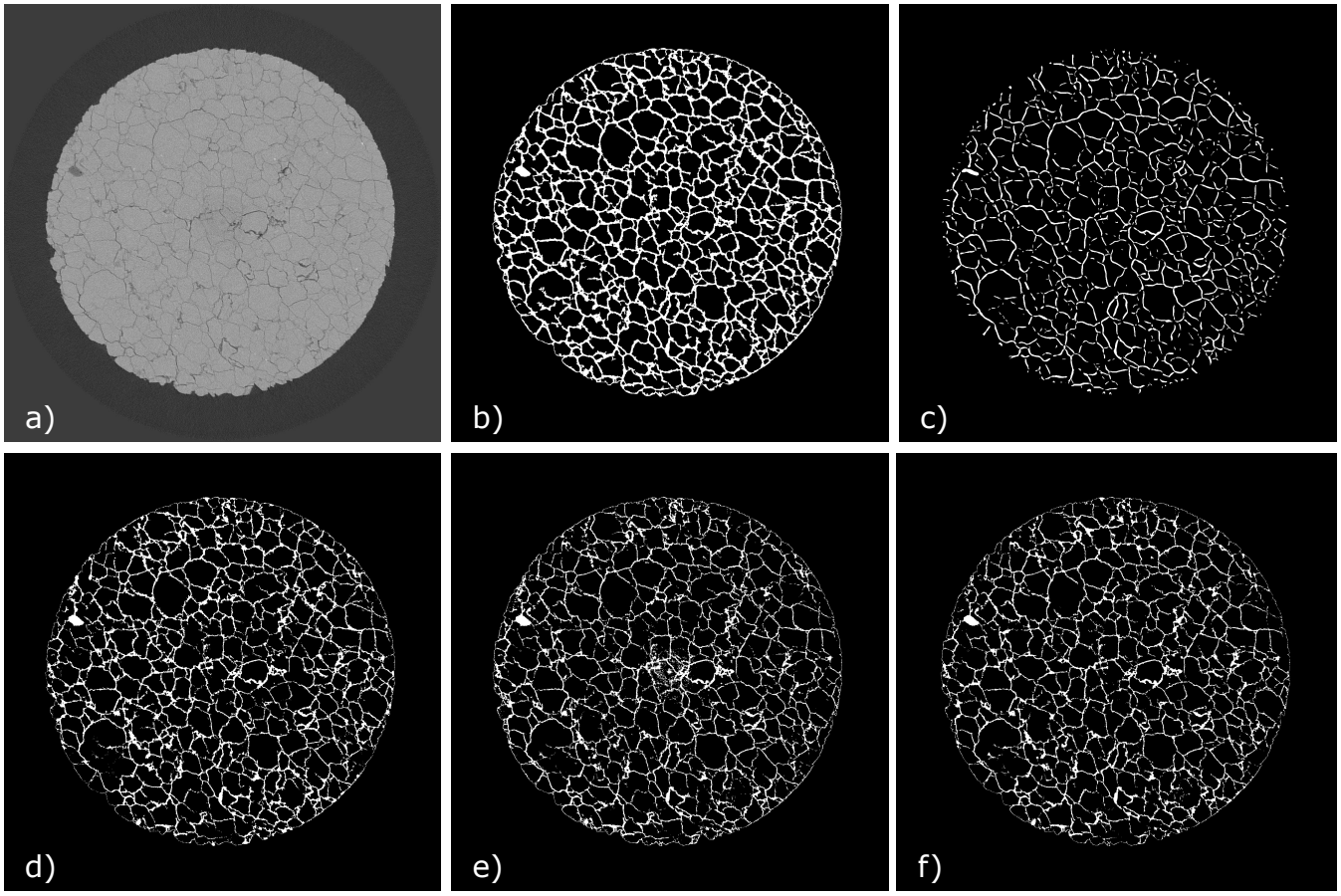

Figure 5. x-y-plane data extracted from the 1000th slice along the z-axis (total 2139). The segmentation results from the a) original data, b) LT, c) Sato, d) Active contouring, e) Random Forest and f) U-net are shown.

and extracting the 1400th slice on the y-axis. By showing the results from side and top, we demonstrate that there was no observable discontinuity of the segmented fractures which could be caused by performing 2D segmentation and splitting.

Based on the 2D results in both Figures 5 and 6, we were able to identify some advantages and limitations for each segmentation method. With the local threshold method we were able to obtain the segmented results which included most of the features within the data-set. The method definitely has the benefit of easy implementation and short computation time. Additionally, it was consistent providing a continuous shape for the fracture network. However, the thickness of the segmented fracture was unrealistically large. Also, due to the small contrast of the features, the method faced challenges distinguishing the features precisely when the fractures were located close to each other. In the case of the Sato method there was no significant benefit observed. The method was able to capture some of the fractures which had an elevated contrast however, it was not able to identify the faint fractures which had a lower contrast. In addition, the method also showed a limitation to classify the fractures at the points where multiple fractures were bifurcating. This was expected as a characteristic feature of a Hessian matrix-based

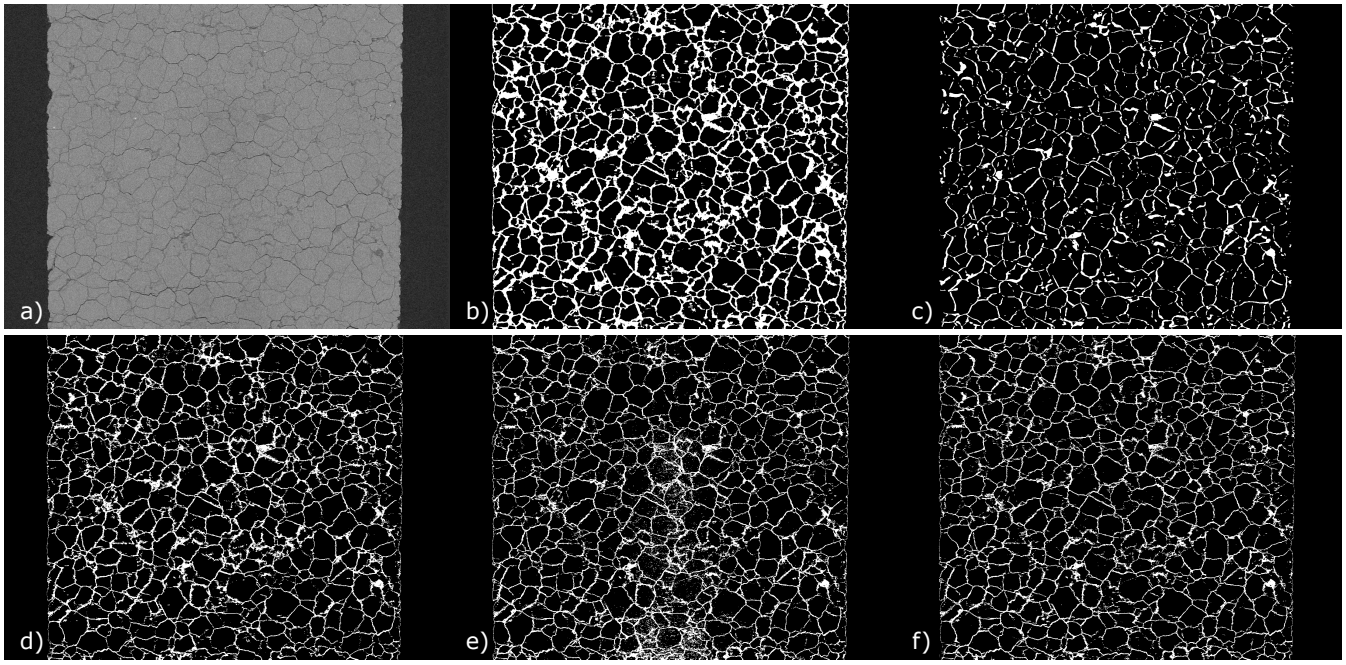

Figure 6. x-z-plane data extracted from the 1400th slice along the y-axis (total 2940). The segmentation results from the a) original data, b) **LT**, c) Sato, d) Active contouring, e) Random Forest and f) U-net are shown.

method. Additionally, the pores detected by this method tended to appear narrower than their real size. Please note that the outer edges between the exterior and inner part of the specimen were excluded with this method.

With the help of the active contouring method we were able to obtain better results than the other conventional segmentation methods. The fractures were detected with thinner profiles, and pores were also identified accordingly, despite of the ring artefacts at the central area of the image (as shown in Figure 5). However, the fractures close to the outer rim still appeared thicker. This was caused by the beam hardening effect, which induces spatially-varying brightness. Additionally, some parts which contained faint fractures were recognized with a weak connection, which meant that the segmented fractures were not well identified. This was induced due to a variation of contrast of the fractures within the region of interest. Although we cropped the images into small tiles before applying the method, the fractures with low contrast tended to be ignored when the fractures of larger contrast were mostly taken into account. Despite the observed drawbacks, the output of the method gave us sufficiently well segmented fractures thus, we adopted this result as a "truth" image for the machine learning-based models.

In the case of the Random Forest method, where training happens with the use of a single slice of output of the active contouring method, the trained model was able to detect the fractures with finer profiles, better than any other conventional method. More specifically, it was able to detect well the fractures when two fractures were located close to each other, which were identified as a merged one with the conventional methods. However, as shown in Figures 5 and 6, it seems that the method

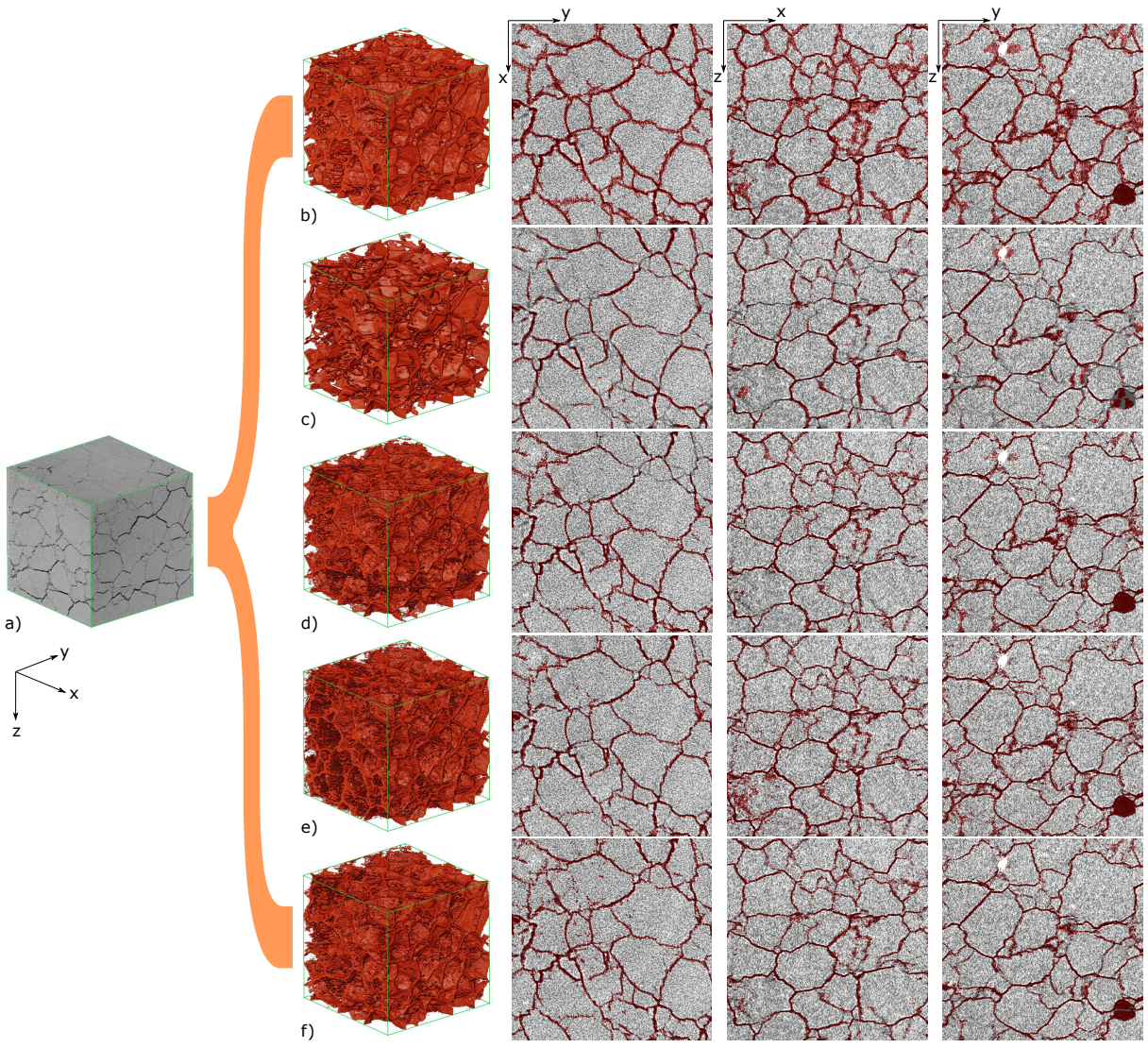

Figure 7. An extracted sub-volume from (1400, 1400, 1000) with a size of 600^3 voxels is shown in a). The segmentation results from b) Local threshold, c) sato, d) Active contouring, e) Random Forest and f) U-net are shown. The 300th slice in x-y-plane is also shown, as well as the 300th slices in x-z/y-z-plane. Each result is overlayed on the corresponding histogram-enhanced original image for a clear comparison.

is sensitive to ring artefacts based on a significant number of fuzzy voxels classified as fractures at the central part of the image, while this was not the case.

The segmentation results of the U-net method outperformed any other method employed in this comparison. With the help of the trained model, we were able to obtain clear segmentation results despite beam hardening and ring artefacts. In Figure 5, all the fractures which were of high and low contrast were efficiently classified while maintaining their aperture scale. In addition, with the help of the overlapping scheme, we obtained continuous fractures even though we trained the model and computed the predictions with cropped images.

For the sake of detailed visualization, we demonstrate the segmentation results in a 3D sub-volume (600^3 voxels) in Figure 7 with overlaid results on the contrast-enhanced original image on x-y- and x-z-planes at the middle (300th) of the sub-volume. The contrast of images was improved by using the histogram equalization method which adjusted the intensities of the image by redistributing those within a certain range Acharya and Ray (2005). In the 3D visualized structure with each of the adopted methods in Figure 7, we were able to obtain an overview of their unique responses. The segmented fractures from each method are marked in red. In the case of the local threshold method, as mentioned earlier, the method was able to recognize all the features however, the fractures were identified with a broader size compared to those in the original data. In addition, the outlines of the results were captured with rough surface profiles. Although the segmentation result with the method showed sufficiently well-matched shapes in general, from the middle of the x-z-plane overlaid image we could conclude that the method was not so appropriate to detect sophisticates profiles, especially when the fractures were next to each other since these features were identified as a cavity. In the case of the Sato method, as we showed in Figure 5 and Figure 6, we were able to conclude from the 3D visualized data that the method was not so efficient to detect the fractures. The roughness of the surface was not able to be accounted for, and an unrealistic disconnection between the fractures was observed due to unidentified fractures with this method.

In the 3D visualized result with the active contouring method in Figure 7, the fractures were generally well classified and visualized. However, as it is shown in the overlaid images in the right top corner, some fractures were not detected due to their low contrast. In addition, the outlines of the fractures with sophisticated shapes were not able to be well-classified.

Although we show in Figures 5 and 6 that the result with the Random Forest method had challenges dealing with ring artefacts, the result in general showed accurate fracture profiles as it is demonstrated in Figure 7 with an overlaid x-y-plane image. The segmentation results showed that the method was able to detect fractures with both high and low contrast. However, in the overlaid x-z-plane image, we can observe that the left bottom side contained a lot of of defects which were also found in the 3D visualization from the rough surface in the left corner. Thus, we were able to conclude that the segmentation was not well performed at some locations.

In the case of the U-net model, the segmented fractures were well matched with the fractures in the original data. Consequently, a clear outlook of fractures was also obtained in the 3D visualized data.

Based on mentioned environment in A, the noise reduction took 8 hours for the entire data-set. The total computation time to obtain the segmentation results were 1.3 hours with local threshold, 35.6 hours with Sato, 118.2 hours with active contouring, 433.7 hours with Random Forest and 4.1 hours with the U-net method. Note that the merging and cropping steps which

Table 1. Summary of the estimated porosity for all adopted segmentation methods in comparison to experimentally determined value

Methods	Estimated porosity (%)	Processing time (hours)
Local Threshold	25.00	1.3
Sato	11.42	35.6
Chan-vese	15.41	118.2
Random Forest	14.67	433.7
U-net	14.16	4.1
Experimental measurement	3	-

were necessary in order to employ the active contouring, Random Forest and U-net method are included in this time estimate (approximately 40 min). The training times for the machine learning based methods took 3.5 hours for the Random Forest and 21 hours for the U-net method.

By counting the number of fracture-classified voxels and dividing with the total number of voxels, we were able to compute an approximated porosity out of the images. The summary of the estimated porosity from all segmentation methods is shown in Table 1. In this estimation, the outer rim was excluded in the counting by applying a mask which was obtained by the combination of the Otsu method for binarization, and the fill and erosion method Vincent and Dougherty (1994). With the help of this evaluation, we acquired the porosity of the local threshold method: 25.00 %, Sato method: 11.42 %, active contouring method: 15.41 %, Random Forest method: 14.67 % and U-net method: 14.16 %. These values were significantly larger than the porosity obtained from measurements of the sample which was around 3 %. The porosity of the thermally treated sample was obtained from measurements of volume changes before/after quenching.

This mismatch between the numerically estimated values and the measurement for porosity was expected. The resolution of the scanned data was not high enough to accurately estimate the size of a single fracture aperture, although we adopted the highest resolution we could achieve Ruf and Steeb (2020a), which was this of 2 microns/voxel, since this resolution is comparable to the aperture of a fracture. In addition, our intention was to obtain geometrical information of fracture networks from a wide field of view. Nevertheless, we used these results to speculate about the comparative accuracy of the segmentation techniques, without emphasising on the accurate prediction of porosity, at this stage.

As we showed in Figure 7, the apertures in the result of the local threshold method were significantly overestimated. In the case of the Random Forest and active contouring method, although the methods were able to recognize fractures with thin profiles, the porosity was exaggerated due to poorly classified pixels. For the porosity estimation based on the Sato method, which showed the lowest value, we demonstrated that there were many undetected features. By considering these facts, the result from the U-net method again gave us the best result among the compared methods.

4 Conclusions

In this work, we demonstrate the segmentation workflows for three different conventional segmentation methods, and two machine learning methods for μ-XRCT data from a dry fracture network which was induced by thermal quenching in Carrara marble. Despite the low contrast of the fractures in the available data, due to their aperture being very close to the resolution of the μ-XRCT, we were able to successfully segment 3D fracture networks with the proposed segmentation workflows.

With the acquired segmentation results we conducted a quality and efficiency comparison and showed that the results with the U-net method were the most efficient and accurate ones. Among the applied conventional segmentation schemes we were able to obtain the best quality of results with the active contouring method, and the best time efficiency with the local threshold method. For the adopted machine learning schemes, we trained the models with results from the active contouring method. This was advantageous compared to manual labelling, especially since manual labelling was far from being the best option given that the fractures were of low contrast. Manual labelling work would be too arduous to obtain sufficient and accurate training data. In addition, we also showed that the defects which were detected with the conventional methods were improved with the trained machine learning methods.

In order to perform the segmentation with conventional methods, the application of the adaptive manifold non-local mean filter, which is one of the noise reduction techniques, was a fundamental step due to the inherent noise of the scanned data. However, with the machine learning methods this step was not necessary. We were able to obtain fully segmented data by providing less filtered data to the training model. This was quite advantageous compared to the conventional approaches since the computation time and resources requirements were significantly reduced. Additionally, the fine-tuning of the input parameters, which were mandatory for the conventional methods, did not play a significant role in the attempt to get good segmentation results.

We showed that obtaining a full 3D structure of segmented data can be performed efficiently by employing the proposed segmentation workflows in 2D images. Thus, the big volume of data could be dealt with a conventional workstation without requiring any advanced properties of a specific image-analysis workstation. Although, we showed that the resolution of the data to estimate the exact size of apertures of fracture networks was not sufficient, by comparing with the actual measurement of porosity, the geometrical information was acquired without demanding any additional procedure. For future work, we intend to correlate the acquired geometrical information regarding fractures with measurement results obtained from (mercury) porosimetry. In addition, characterization of fractures in terms of lengths and angle will be further investigated.

Code availability. The first version of the *fracture network segmentation* used for classification of sophisticated shapes of fracture network from quenched carrara marble data Ruf and Steeb (2020b) is preserved at https://doi.org/10.18419/darus-1847, available via no registration and create commons attribution conditions, and developed openly at python (https://www.python.org/), matlab (https://www.mathworks.com) and beanshell (https://beanshell.github.io/).

Data availability. The tomographic data of quenched carrara marble Ruf and Steeb (2020b) used for validation efficiency and testing effectiveness of adopted segmentation methods of in the study are available at DARUS, *micro-XRCT data set of Carrara marble with artificially created crack network: fast cooling down from 600°C* via doi:10.18419/darus-682 with no registration and creative commons attribution conditions.

Appendix A: Technical specifications

An open micro-focus tube *FineTec FORE 180.01C TT* with a tungsten transmission target from Finetec Technologies GmbH, Germany, in combination with a *Shad-o-Box 6K HS* detector with a CsI scintillator option from Teledyne DALSA Inc., Waterloo, Ontario, Canada with a resolution of $2940\,\mathrm{x}\,2304$ pixels and a pixel pitch of $49.5\,\mu\mathrm{m}$ and 14-bit depth resolution was employed. For more details about the system please refer to Ruf and Steeb (2020a). The highest achievable spatial resolution with this system is about 50 line-pairs/mm at $10\,\%$ of the Modulation Transfer Function (MTF), which is equal to a resolvable feature size of about $10\,\mu\mathrm{m}$. To achieve this, we are using a geometric magnification of 24.76 which results in a final voxel size of $2.0\,\mu\mathrm{m}$. With these settings the Field Of View (FOV) is about $5.88\,\mathrm{mm}$ in the horizontal direction and $4.61\,\mathrm{mm}$ in the vertical one. Consequently, the sample could be scanned over the entire diameter of $5\,\mathrm{mm}$. The overall physical size of the FOV and consequently the 3D volume is required for the definition of an appropriate Representative Elementary Volume (REV) for subsequent Digital Rock Physics (DRP), which is at the same time the challenge between a sufficient large FOV and a high spatial resolution to resolve the fractures. For further details about the image acquisition settings cf. Ruf and Steeb (2020b) and Ruf and Steeb (2020a). In order to reduce the beam hardening effect, the 0.5 mm thickness of the aluminium plate was used as a filter during the acquisition. The reconstruction was performed using the Filtered Back Projection (FBP) method implemented in the software *Octopus Reconstruction (Version 8.9.4-64 bit)*. With the same software, the simple beam hardening correction and ring artefact removal were conducted after employing FBP. The reconstructed 3D volume consists of $2940\,\mathrm{x}2940\mathrm{x}2139$ voxels, saved as an image stack of 2139 16-bit *.tif image files of $2940\,\mathrm{x}2940$ resolution. Noise reduction was deliberately omitted in the reconstruction process in order to provide an interested reader with the possibility to use their own adequate noise filtering methods. The 3D data-set with an extraction to better show the homogeneous fracture network is illustrated in Figure 1. The bright gray area represents the calcite phase whereas the dark gray area in between the calcite phase shows the initiated micro-fractures which were generated by the thermal treatment. As a consequence of the thermal treatment, a bulk volume increase of about $3.0\,\%$ under ambient conditions can be recorded if a perfect cylindrical sample shape is assumed. The noise reduction and segmentation workflows were performed with the hardware specification of intel(R) Core(TM) i7-8750H CPU @2.2GHz, 64GB of RAM and Nvidia Quadro P1000 (4GB).

Appendix B: Noise reduction schemes

The AMNLM filter operates in two parts: 1.) Creation of matrices (manifolds) which are filled with weighted intensities by coefficients. 2.) Iterative processing on created matrices. In the first part, in order to reduce the complexity and increase the efficiency of the filter, Gastal and Oliveiray (2012) applied the Principal Component Analysis (PCA) Pearson (1901) scheme

on the created matrices. In the second part, sequential steps were presented to adjust intensities of pixels based on components of created and PCA-treated matrices. These procedures are called *Splatting*, *Blurring* and *Slicing*. The *Splatting* and *Blurring* are iterative steps on components of created matrices with numerical modifications (adaptive manifolds). After these iterative procedures, *Slicing* is performed which is merely summing the outputs from the iteration and normalizing it.

From the creation of manifolds the coefficients to weight intensities were designed to have a shape of a Gaussian distribution in the kernel. A typical two-dimensional isotropic Gaussian kernel can be derived as follows,

$$G((x,y);\sigma_f) = A\exp\left(-\frac{x^2+y^2}{2\sigma_f^2}\right) \tag{B.1}$$

where A is an arbitrary chosen amplitude of the Gaussian profile, x and y are input vectors which have a mean value of zero, and σ_f is the standard deviation of Gaussian distribution.

By using the above profile and input vectors, a finite sized Gaussian kernel G_k could be defined. With the created Gaussian kernel, the manifolds of weighted intensities f_A could be created as follows,

$$f_A = vec(G_k)^T \cdot vec(I) \tag{B.2}$$

where $vec(\cdot)$ is a vectorization operator which transforms a matrix into a column vector, and I is the 2D image matrix which contains intensities.

Each dimension of f_A contains weighted intensities from the image. The dimensionality of the PCA scheme was reduced in order to boost the computation efficiency while minimizing the loss of information Jollife and Cadima (2016). This could be achieved by finding the principal components which would maximize the variance of data in dimensions of f_A. The principal components could be obtained by computing eigenvalues and theirs corresponding eigenvectors of covariance matrix C. The covariance matrix C of f_A is acquired as follows,

$$C = (f_A - \mathcal{E}(f_A))^T(f_A - \mathcal{E}(f_A)) \tag{B.3}$$

where $\mathcal{E}$ operator is to compute means of its arguments.

Among the eigenvalues of the obtained C, the bigger ones indicate higher variances of the data in the corresponding eigenvector space Jollife and Cadima (2016). Thus, after sorting the corresponding eigenvectors in an order of magnitudes of eigenvalues, by performing multiplication with these eigenvectors and f_A, the manifolds f_A could be ordered in order of significance. Based on this, the user defined number of dimensionality s is adopted and the manifold f_A is recast into f_S by extracting s number of principal components.

B0.1 Splatting

With the given extracted manifolds f_S, the Gaussian distance-weighted projection is performed as follows,

$$\Psi_0 = \exp\left(-\frac{\sum_{i=1}^{S}|f_i - \eta_i|^2}{2\sigma_r^2}\right), \tag{B.4}$$

with Ψ_p being the coefficients to weight the original image (projection), σ_r being a chosen filtering parameter which states the standard deviation of filter range, and η_S denoting the *Adaptive manifolds* which are noise reduced responses by a low-pass filter(eq. B.5) at the first iteration. These adaptive manifolds will be further updated via an iteration process. Consequently, the Euclidean distances between each of f_S and η_S are computed and integrated into Ψ_0 by considering the standard deviation σ_r.

In order to obtain *Adaptive manifolds*, the numerical description of the low-pass filter for a 1-D signal in the work of Gastal and Oliveiray (2012) is as follows,

$$S_{out}[i] = S_{in}[i] + \exp\left(-\frac{\sqrt{2}}{\sigma_s}\right)(S_{in}[i-1] - S_{in}[i]), \tag{B.5}$$

where S_{out} is the response of the filter, S_{in} is the signal input, and σ_s is a filtering parameter which states the spatial standard deviation of the filter. i denotes the location of a pixel in the image. Note that this low-pass filter has to be applied to each direction in an image due to its non-symmetric response corresponding to the applied direction. With this relation, the collected response has a smoother profile than the original signal.

By performing element-wise multiplication on the acquired Ψ_p and the original image I, the weighted image Ψ could be obtained such as,

$$\Psi = \Psi_0 \circ I. \tag{B.6}$$

where $\circ$ denotes the Schur product Davis (1962).

B0.2 Blurring

In the computed weighted image Ψ and the *adaptive manifolds* η_S, Recursive Filtering (RF) Gastal and Oliveira (2011) is applied so that the values of each manifold in η_S can be blurred accordingly in Ψ and Ψ_0. Gastal and Oliveira (2011) proposed to conduct this procedure in a down-scaled domain so that further smoothing could be expected after the following up-scaling interpolation. The down-scaling factor d_f is calculated as such (e.g., $d_f = 2$ states a half-scaled domain),

$$d_f = max(1, 2\lfloor \log_2(min(\sigma_s/4, 256\sigma_r))\rfloor) \tag{B.7}$$

by taking into account the adopted spatial/intensity filter range of standard deviations σ_s and σ_r.

B0.3 Slicing

As mentioned above, the *Splatting* and *Blurring* steps are performed in an iterative manner and, at each iteration step, a weight projection matrix Ψ_0 and a weighted image Ψ are created. By defining the *Blurring* responses of these with RF as $\Psi^{blur}(k)$ and ${\Psi_0}^{blur}(k)$ where k represents iteration number, the final normalized result of the filter I_f is as follows,

$$I_f = \frac{\sum_{k=1}^{K} \Psi^{blur}(k) \circ \Psi_0(k)}{\sum_{k=1}^{K} {\Psi_0}^{blur}(k) \circ \Psi_0(k)} \tag{B.8}$$

where K is the total numbers of created adaptive manifolds. Note that up-scaling is performed by bilinear interpolation on every Ψ^{blur} and ${\Psi_0}^{blur}$ to recover the domain size from the down-scaled one due to *Blurring* step.

B0.4 Adaptive manifolds

In the first iteration, the η_S are obtained from the low-pass filter response of f_S with the use of eq. B.5. The responses of the low-pass filter locally represent the mean value of the local intensities, which reflects well the majority of intensities within the sub-domain while it lacks variation. By taking into account this drawback, Gastal and Oliveira (2011) introduced a hierarchical structure to include better representative information from additionally created dimensions. These additional dimensions are created by clustering the pixels of f_S. By taking each of η_S as default points, the pixels which had intensities above η_S and below are classified accordingly.

This type of clustering, i.e. finding which pixels are located above and below η_S, could be acquired by the eigenvalues of the covariance matrix of $(f_S - \eta_S)$ with eq. B.3. From the dot product of the obtained eigenvalues and $(f_S - \eta_S)$, the clustered pixels are,

$$\begin{cases} p_i \in C_-, & \text{if J < 0,} \\ p_i \in C_+, & \text{otherwise,} \end{cases} \tag{B.9}$$

where J is the dot product, p_i is pixel i, C_- and C_+ denote the clusters with pixels below and above the manifold, respectively.

Based on the computed clusters C_- and C_+, *Adaptive manifolds* are calculated as follows,

$$\eta_{S-} = \mathcal{L}_p(C_- \circ (1 - \Psi_0) \circ f_S, \sigma_s) \oslash \mathcal{L}_p(C_- \circ (1 - \Psi_0), \sigma_s), \tag{B.10}$$

and

$$\eta_{S+} = \mathcal{L}_p(C_+ \circ (1 - \Psi_0) \circ f_S, \sigma_s) \oslash \mathcal{L}_p(C_+ \circ (1 - \Psi_0), \sigma_s), \tag{B.11}$$

where $\mathcal{L}_p$ indicates the low-pass filter in eq. B.5 and $\oslash$ is an element-wise division operator Wetzstein et al. (2012). Note that the low-pass filter is performed on down-scaled domain such as the step in *Blurring*. Thus, in order to match the size of matrix, η_S is up-scaled before it is applied to *Splatting* step. These steps are performed iteratively (compare eq. B.8). Thus, $K = 2^h - 1$ numbers of manifold are created while h is defined as follows,

$$h = 2 + max(2, \lceil (\lfloor \log_2 \sigma_s \rfloor - 1)(1 - \sigma_r) \rceil). \tag{B.12}$$

Appendix C: Sato filtering

The filter consists of two fundamental parts: a Gaussian filter and a Hessian matrix. In the Gaussian filtering part, with the Gaussian shape profile in eq. B.1, a finite-sized kernel w is designed by defining the input vectors $(x, y) = [-l_w, l_w]$ with a mean value of zero, and l_w being a positive integer defined by the standard deviation σ_f and truncate factor t_f e.g. ($l_w = t_f * \sigma_f + 0.5$). For the sake of stability of the kernel behavior, the kernel is normalized by dividing with the sum of all of its elements. Finally, the Gaussian filter can be expressed by means of the kernel with a convolution operator $\circledast$ such as,

$$I_f(\mathbf{X}) = G(w(x, y); \sigma_f) \circledast I(\mathbf{X}), \tag{C.1}$$

where I is an intensity matrix (input image). in the spatial domain $\mathbf{X}$ and I_f being Gaussian-filtered image.

The Hessian matrix is a matrix of second order partial derivatives of intensities in the x and y directions, and is often used to trace local curvatures with its eigenvalues. By bringing this concept to an image we can deduce a local shape at selected pixels, while observing the changes of intensity gradients. The numerical description of a Hessian matrix with a Gaussian filtered image at selected pixel positions z is as follows,

$$\nabla^2 I_f(z) = \begin{bmatrix} \frac{\partial^2}{\partial x^2} I_f(z) & \frac{\partial^2}{\partial x \partial y} I_f(z) \\ \frac{\partial^2}{\partial y \partial x} I_f(z) & \frac{\partial^2}{\partial y^2} I_f(z) \end{bmatrix} \tag{C.2}$$

With eq. C.1 we can re-write the elements of the matrix as the convolution of the input image and the second partial derivatives of the Gaussian shape function. For example, the first element of the Hessian matrix is,

$$\frac{\partial^2}{\partial x^2} I_f(z) = \frac{\partial^2}{\partial x^2} G(w(x,y); \sigma_f) \circledast I(z_w) \tag{C.3}$$

where z_w is the kernel which has its center at z and the same size as G. In this way we obtain a scalar element by convolution. Note that the second derivatives of the Gaussian shape function are conventionally used for reducing the noise while reinforcing the response of the signal with a specific standard deviation (in this case σ_f) Voorn et al. (2013). Thus, the elements in eq. C.3 represent an enhanced intensity response of the aperture of the fracture in the original input image.

Defining the local structure at pixel z with the help of second order partial derivatives could be done by calculating the eigenvalues of the obtained Hessian matrix. In this case, since $\frac{\partial^2}{\partial x \partial y} I_f$ and $\frac{\partial^2}{\partial y \partial x} I_f$ are identical, the matrix is symmetric with real numbers thus, the eigenvalues of the matrix can be assured to be real numbers and its eigenvectors have to be orthogonal Golub and Van Loan (1996).

By defining the eigenvalues as λ_1 and λ_2, the line-like shape curvature have a relation of $|\lambda_1| \approx 0, |\lambda_2| \gg 0$ or vice versa. This is because each eigenvalue states an amount of gradient change at the selected pixel to each eigenvector direction (orthogonal when $\lambda_1 \neq \lambda_2$). Then, in the case of line-like shape (long and thin), one must have a very small value of $|\lambda_1|$ while the other has relatively bigger value of $|\lambda_2|$. The author proposed to replace the intensity value at pixel z to $|\lambda_2|$ under the above mentioned line-like shape condition and to zero otherwise. Finally, the maximum responses from the output images were accumulated in a range of different σ_f to maximize the effects for various fracture apertures.

Based on this scheme, the multi-scaled Sato filtering method was applied on the noise reduced data with the `sato` function from python *skimage* library. Since the output of this method is an enhanced response of a string-like shape while minimizing the responses from the other shapes of structures, a further binarization method has to be applied to the output of the Sato method in order to obtain logical type of results. Here, we adopted the local threshold method (cf. 2.3.1) which was able to effectively distinguish the enhanced responses out of the output. As we mentioned in 2.3.1, before applying the binarization method, the outer part of the scanned data was eliminated since the contrast between the outer and inner part is bigger than the contrast of the fractures in the data.

After applying the binarization scheme, the remaining artifacts were eliminated with the use of morphological schemes such as `erosion` and `remove_small_object` which are also supported in the Python *skimage* library.

Appendix D: Active contouring

By employing the method which was introduced in the work of Chan and Vese (2001), we were able to obtain finer profiles from the segmented images. The method requires an initial mask which is a roughly segmented binary image and preferably contains most of the features of interest. Thus, we adopted the result of LT (cf. 2.3.1) as the initial mask. From the given initial mask and the corresponding original image, we proceeded with the method to obtain a segmented image. This could be done by minimizing the differences between the mean intensities of in-/out areas, defined by the boundaries of the initial mask C, and the intensities from the original images $I(x,y)$ based on a modified version of the *Mumford-Shah* functional Mumford and Shah (1989),

$$F(c_1,c_2,C) = \mu L(C) + \nu A(in(C)) + \lambda_1 \int\limits_{in(C)} |I(x,y)-c_1|^2 \,\mathrm{d}x\mathrm{d}y + \lambda_2 \int\limits_{out(C)} |I(x,y)-c_2|^2 \,\mathrm{d}x\mathrm{d}y, \tag{D.1}$$

where F is the functional of c_1 and c_2 which are the average intensities within the region of in-/outside of C, and L is the length and A is the inside area of C. The μ, ν, λ_1 and λ_2 are input parameters.

In order to derive the numerical description within the common domain $(x,y) \in \Omega$, the level set scheme, which is able to describe the boundaries of a feature of interest and its area by introducing higher dimensional manifold Osher and Tsai (2003), was combined in this relation. By following this scheme, the boundaries of the initial mask C and inner-/outer areas of it can be defined such as,

$$\begin{cases} C : \Phi(x,y) = 0, \\ in(C) : \Phi(x,y) > 0, \\ out(C) : \Phi(x,y) < 0, \end{cases} \tag{D.2}$$

where $\Phi(x,y)$ is a Lipschitz continuous function to assure that it has a unique solution for a given case.

With the help of the scheme and a Heavyside step function H, eq. D.1 could be rewritten within the common domain Ω as follows,

$$\begin{aligned} F(c_1,c_2,\Phi) =& \mu \int\limits_{\Omega} |\nabla H(\Phi(x,y))| \,\mathrm{d}x\mathrm{d}y + \nu \int\limits_{\Omega} H(\Phi(x,y)) \,\mathrm{d}x\mathrm{d}y \\ &+ \lambda_1 \int\limits_{\Omega} |I(x,y)-c_1|^2 H(\Phi(x,y)) \,\mathrm{d}x\mathrm{d}y + \lambda_2 \int\limits_{\Omega} |I(x,y)-c_2|^2 (1-H(\Phi(x,y))) \,\mathrm{d}x\mathrm{d}y. \end{aligned} \tag{D.3}$$

For fixed Φ, the intensity averages of interior c_1 and exterior c_2 of the contour could be obtained such as,

$$c_1(\Phi) = \frac{\int_\Omega I(x,y) H(\Phi(x,y)) \,\mathrm{d}x\mathrm{d}y}{\int_\Omega H(\Phi(x,y)) \,\mathrm{d}x\mathrm{d}y}, \tag{D.4}$$

and

$$c_2(\Phi) = \frac{\int_\Omega I(x,y)(1-H(\Phi(x,y))) \,\mathrm{d}x\mathrm{d}y}{\int_\Omega (1-H(\Phi(x,y))) \,\mathrm{d}x\mathrm{d}y}. \tag{D.5}$$

Based on these relations, in order to minimize F, the authors used the following scheme by relating it with an Euler-Lagrange equation. By applying the artificial time $t \geq 0$, the descending direction of Φ is such as,

$$\frac{\partial \Phi}{\partial t} = \delta(\Phi)\left[\mu \,\mathrm{div}\left(\frac{\mathrm{grad}(\Phi)}{|\mathrm{grad}(\Phi)|}\right) - \nu - \lambda_1(I(x,y)-c_1)^2 + \lambda_2(I(x,y)-c_2)^2\right], \tag{D.6}$$

where δ is the dirac delta function which is the first derivative of the Heavyside function H in one-dimensional form thus, only the zero level-set part could be considered. From the initial contour $\Phi(x, y, t = 0)$, a new contour was able to be defined by using the above relation till it reached to stationary state via iteration of the artificial time t.

Based on the theory, the method was applied on the original data which had not been dealt with the noise reduction technique. Additionally, the same masking criteria which were applied before the local threshold method 2.3.1 were employed before using the method. This is because the method is not able to cope with the image data which contain a large contrast between the inner and outer part of the sample, as mentioned before. Thus, by covering the exterior part with a manually extracted mean intensity of the non-fracture matrix, an effective segmentation of the fracture was able to be performed. Furthermore, in order to capture faint fractures which had a lower contrast than the other fractures, the method was applied on the cropped images with a size of (400 $\times$ 400 voxels). These segmented small tiles were merged later into the original size of the image.(See Figure 5)

Appendix E: Splitting, Training, Merging for the U-net model

The schematic of the used U-net model is shown in Figure 4. We adopted a 2×2 kernel for max-pooling and the deconvolutional layers. The kernel size of the convolutional layers was decided to be 3×3. The adopted activation function for the convolutional layer was the Rectifier Linear Unit (relu) which has a range of output from 0 to ∞. It returns 0 for negative inputs and a linear increment up to infinity for positives. On the final output layer, the sigmoid activation function was chosen, which has a non-linear shape and its output range is from 0 to 1. Consequently, the obtained output was a probability map where a number close to 1 was considered as the fracture and the rest was accounted for as non-fracture. The workflow of data processing is described in Figure E1, and the same principle holds for the Random Forest scheme.

E1 Splitting

Due to the in-house computer's memory limitations, the big size of the complete data-set could not be dealt with the adopted machine learning model. This was because of the data which would be used for training and the model itself had to be allocated in the random-access memory RAM thus, a training data-set which exceeded its memory limitation could not be applied. With the given information, the size of the applicable data for training was decided by means of non-/trainable parameters, induced by the number and size of the layers within the model. The size of the images and the batch size of the training data indicated the number of training data per one iteration of training (See E2). In our application moreover, since we wanted to take a benefit from the GPU, which could boost the training and processing time, the limitation of usable memory was even more restricted within the memory size of a GPU, which is normally smaller than the RAM and is not easily able to be expanded.

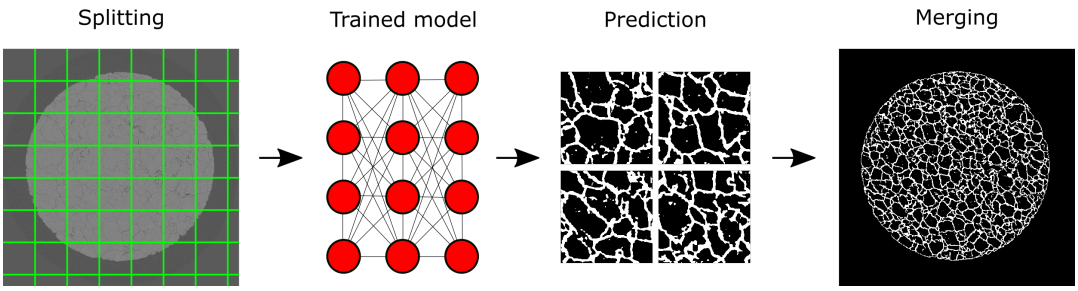

Figure E1. The workflow of the machine learning-based model. The green grid indicates the edges of the diced tiles (400 × 400 voxels) from the original image (2940 × 2940 voxels). After inputting each fragment into the trained model, we obtained the corresponding predictions. By applying the threshold (≈ 0.5 in the most cases) obtained by the Otsu method the tiles were binarized and merged to shape the original size of the image.

Consequently, due to these facts, the original size of 2D image data was split into small tiles. The original image size of 2940 × 2940 pixels was cropped into multiple 400 × 400 smaller images. We already knew that this could potentially induce some disconnection of the segmented fractures along the tiles, since the segmentation accuracy of the U-net model is low at the edges of given data due to the inevitable information loss during up-/down-scaling. In order to overcome and compensate this drawback, the overlapping scheme was applied, splitting the tiles to have 72 pixels of overlapping region at their borders. In this manner, 81 tiles were produced per 1 slice of a 2D original image. In addition, the format of the images which was "unsigned int 16" originally, was normalized into "unsigned int 8" after cutting the evaluated mean minimum and maximum intensities. This helped to reduce the memory demand caused by training data while preventing an over-simplification of the inherent histogram profiles.

E2 Training

Training was performed with the training data-set mapping the cropped original image and GT. Only few data (60 slices) of scanned data (where the data was located at the top part of the sample) was used as training data-set. In case of the training of the U-net model, this number of slices corresponded to 4860 cropped tiles. Among the data-set, 80 percent of the data (3888 tiles) were applied for updating the internal coefficients of the model, and the rest of the data (972 tiles) were used to evaluate the performance of the trained model by comparison between the predictions and GT.

The batch size in machine learning defines how many of training data to be used for updating the inner parameters of the model during training. This implies that the batch size would directly affect the memory usage since the defined numbers of training data would be held on the memory. Therefore, in this case, this number was selected carefully as 5 so as not to reach our GPU memory limitation. The steps per epoch was set at 777, estimated based on the following relation: $\approx \frac{\text{training data:}3888}{\text{batch size:}5}$ in a way that all the provided trainable data could be applied on each epoch. Finally, the model was trained with 100 epochs so that the segmentation accuracy could be enhanced along the iterations.

We adopted "binary cross entropy" loss function in order to deal with the binarization of image (classification of non-/fractures). The loss function L is defined such as,

$$L = -g_t \log(p) - (1 - g_t) \log(1 - p), \tag{E.1}$$

where g_t is the given truth, and p is the predicted probability of the model. Thus, the model was updated in a way that the difference between truths and predictions could be narrowed down. The "Adam" optimizer Kingma and Ba (2017) was used for this optimization. The adopted fixed learning rate was 1e-4.

E3 Merging

After training the model by following the procedure mentioned earlier E2, the model was able to predict the non-/fractures within the given tiles. In order to obtain the same size of output with the original image, we performed this merging procedure which was basically placing every prediction tile at their corresponding locations, after collecting the predictions from the entire data-set from the model. Since we applied the overlapping scheme E1 in order to prevent disconnections between tiles, 36 pixels at the edges were dropped for each tile which were considered as inaccurate predictions before merging. After the merging, the predictions which were in the range of 0 to 1 due to the output layer of the training model were binarized with the Otsu method Nobuyuki Otsu (1979) for the final segmentation output. The pixels which were classified as zeros in the predictions were excluded from this binarization by assuming that these pixels were certainly non-fracture.

Author contributions. DL: implementation and performance of image analyse/segmentation, data analysis, original draft preparation. NK: original draft preparation, designed research. MR: data curation, XRCT scanning, original draft preparation. HS: original draft preparation, designed research.

Competing interests. The authors declare that they have no conflict of interest

Acknowledgements. This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project Number 327154368 – SFB 1313. Holger Steeb is thanking the Deutsche Forschungsgemeinschaft (DFG) for supporting this work by funding EXC 2075-390740016 under Germany's Excellence Strategy.

References

Acharya, T. and Ray, A. K.: Image Processing: Principles and Applications, https://doi.org/10.1002/0471745790, 2005.

Ahamed, B. B., Yuvaraj, D., and Priya, S. S.: Image Denoising with Linear and Non-linear Filters, Proceedings of 2019 International Conference on Computational Intelligence and Knowledge Economy, ICCIKE 2019, 10, 806–810, https://doi.org/10.1109/ICCIKE47802.2019.9004429, 2019.

Al-amri, S. S., Kalyankar, N. V., and S.D, K.: Image segmentation by using edge detection, International Journal on Computer Science and Engineering, 2, 804–807, 2010.

Albawi, S., Mohammed, T. A., and Al-Zawi, S.: Understanding of a convolutional neural network, in: Proceedings of 2017 International Conference on Engineering and Technology, ICET 2017, vol. 2018-Janua, pp. 1–6, IEEE, https://doi.org/10.1109/ICEngTechnol.2017.8308186, 2018.

Alqahtani, N., Alzubaidi, F., Armstrong, R. T., Swietojanski, P., and Mostaghimi, P.: Machine learning for predicting properties of porous media from 2d X-ray images, Journal of Petroleum Science and Engineering, 184, 106 514, https://doi.org/10.1016/j.petrol.2019.106514, 2020.

Alzubaidi, F., Makuluni, P., Clark, S. R., Lie, J. E., Mostaghimi, P., and Armstrong, R. T.: Automatic fracture detection and characterization from unwrapped drill-core images using mask R–CNN, Journal of Petroleum Science and Engineering, 208, 109 471, https://doi.org/https://doi.org/10.1016/j.petrol.2021.109471, 2022.

Amit, Y. and Geman, D.: Shape Quantization and Recognition with Randomized Trees, Neural Computation, 9, 1545–1588, https://doi.org/10.1162/neco.1997.9.7.1545, 1997.

Arena, A., Delle Piane, C., and Sarout, J.: A new computational approach to cracks quantification from 2D image analysis: Application to micro-cracks description in rocks, Computers & Geosciences, 66, 106–120, https://doi.org/https://doi.org/10.1016/j.cageo.2014.01.007, 2014.

Arganda-Carreras, I., Kaynig, V., Rueden, C., Eliceiri, K. W., Schindelin, J., Cardona, A., and Seung, H. S.: Trainable Weka Segmentation: A machine learning tool for microscopy pixel classification, Bioinformatics, 33, 2424–2426, https://doi.org/10.1093/bioinformatics/btx180, 2017.

Behrenbruch, C. P., Petroudi, S., Bond, S., Declerck, J. D., Leong, F. J., and Brady, J. M.: Image filtering techniques for medical image post-processing: An overview, British Journal of Radiology, 77, https://doi.org/10.1259/bjr/17464219, 2004.

Berkowitz, B.: Analysis of Fracture Network Connectivity Using Percolation Theory, Mathematical Geology, 27, 467–483, https://doi.org/10.1007/BF02084422, 1995.

Beucher, S. and Meyer, F.: The Morphological Approach to Segmentation: The Watershed Transformation, in: Mathematical morphology in image processing, vol. 18, p. 49, iSBN: 1351830503, 9781351830508, 1993.

Bharodiya, A. K. and Gonsai, A. M.: An improved edge detection algorithm for X-Ray images based on the statistical range, Heliyon, 5, e02 743, https://doi.org/10.1016/j.heliyon.2019.e02743, 2019.

Buades, A., Coll, B., and Morel, J.-M.: Non-Local Means Denoising, Image Processing On Line, 1, 208–212, https://doi.org/10.5201/ipol.2011.bcm_nlm, 2011.

Canny, J.: A Computational Approach to Edge Detection, IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-8, 679–698, https://doi.org/10.1109/TPAMI.1986.4767851, 1986.

Caselles, V., Catté, F., Coll, T., and Dibos, F.: A geometric model for active contours in image processing, Numerische Mathematik, 66, 1–31, https://doi.org/10.1007/BF01385685, 1993.

Chan, T. F. and Vese, L. A.: Active contours without edges, IEEE Transactions on Image Processing, 10, 266–277, https://doi.org/10.1109/83.902291, 2001.

Chauhan, S., Rühaak, W., Khan, F., Enzmann, F., Mielke, P., Kersten, M., and Sass, I.: Processing of rock core microtomography images: Using seven different machine learning algorithms, Computers & Geosciences, 86, 120–128, https://doi.org/10.1016/j.cageo.2015.10.013, 2016.

Chollet, F. et al.: Keras, https://github.com/fchollet/keras, 2015.

Christe, P., Bernasconi, M., Vontobel, P., Turberg, P., and Parriaux, A.: Three-dimensional petrographical investigations on borehole rock samples: A comparison between X-ray computed- and neutron tomography, Acta Geotechnica, 2, 269–279, https://doi.org/10.1007/s11440-007-0045-9, 2007.

Chung, S. Y., Kim, J. S., Stephan, D., and Han, T. S.: Overview of the use of micro-computed tomography (micro-CT) to investigate the relation between the material characteristics and properties of cement-based materials, Construction and Building Materials, 229, 116 843, https://doi.org/10.1016/j.conbuildmat.2019.116843, 2019.

Coady, J., O'Riordan, A., Dooly, G., Newe, T., and Toal, D.: An overview of popular digital image processing filtering operations, Proceedings of the International Conference on Sensing Technology, ICST, 2019-Decem, https://doi.org/10.1109/ICST46873.2019.9047683, 2019.

Cracknell, M. J. and Reading, A. M.: Geological mapping using remote sensing data: A comparison of five machine learning algorithms, their response to variations in the spatial distribution of training data and the use of explicit spatial information, Computers & Geosciences, 63, 22–33, https://doi.org/https://doi.org/10.1016/j.cageo.2013.10.008, 2014.

Crawford, B. R., Tsenn, M. C., Homburg, J. M., Stehle, R. C., Freysteinson, J. A., and Reese, W. C.: Incorporating Scale-Dependent Fracture Stiffness for Improved Reservoir Performance Prediction, Rock Mechanics and Rock Engineering, 50, 3349–3359, https://doi.org/10.1007/s00603-017-1314-z, 2017.

Davis, C.: The norm of the Schur product operation, Numerische Mathematik, 4, 343–344, https://doi.org/10.1007/BF01386329, 1962.

De Kock, T., Boone, M. A., De Schryver, T., Van Stappen, J., Derluyn, H., Masschaele, B., De Schutter, G., and Cnudde, V.: A pore-scale study of fracture dynamics in rock using X-ray micro-CT under ambient freeze-thaw cycling, Environmental Science and Technology, 49, 2867–2874, https://doi.org/10.1021/es505738d, 2015.

Delle Piane, C., Arena, A., Sarout, J., Esteban, L., and Cazes, E.: Micro-crack enhanced permeability in tight rocks: An experimental and microstructural study, Tectonophysics, 665, 149–156, https://doi.org/10.1016/j.tecto.2015.10.001, 2015.

Deng, H., Fitts, J. P., and Peters, C. A.: Quantifying fracture geometry with X-ray tomography: Technique of Iterative Local Thresholding (TILT) for 3D image segmentation, Computational Geosciences, 20, 231–244, https://doi.org/10.1007/s10596-016-9560-9, 2016.

Dhanachandra, N., Manglem, K., and Chanu, Y. J.: Image Segmentation Using K-means Clustering Algorithm and Subtractive Clustering Algorithm, Procedia Computer Science, 54, 764–771, https://doi.org/10.1016/j.procs.2015.06.090, 2015.

Dong, Y., Li, P., Tian, W., Xian, Y., and Lu, D.: Journal of Natural Gas Science and Engineering An equivalent method to assess the production performance of horizontal wells with complicated hydraulic fracture network in shale oil reservoirs, Journal of Natural Gas Science and Engineering, 71, 102 975, https://doi.org/10.1016/j.jngse.2019.102975, 2019.

Drechsler, K. and Oyarzun Laura, C.: Comparison of vesselness functions for multiscale analysis of the liver vasculature, in: Proceedings of the 10th IEEE International Conference on Information Technology and Applications in Biomedicine, pp. 1–5, https://doi.org/10.1109/ITAB.2010.5687627, 2010.

Dura, R. and Hart, P.: Pattern Classification and Scene Analysis, John Wiley and Sons, first edn., iSBN: 0471223611, 9780471223610, 1973.

Erdt, M., Raspe, M., and Suehling, M.: Automatic Hepatic Vessel Segmentation Using Graphics Hardware, in: Medical Imaging and Augmented Reality, edited by Dohi, T., Sakuma, I., and Liao, H., pp. 403–412, Springer Berlin Heidelberg, Berlin, Heidelberg, iSBN: 978-3-540-79982-5, 2008.

Fei, Y., Wang, K. C. P., Zhang, A., Chen, C., Li, J. Q., Liu, Y., Yang, G., and Li, B.: Pixel-Level Cracking Detection on 3D Asphalt Pavement Images Through Deep-Learning- Based CrackNet-V, IEEE Transactions on Intelligent Transportation Systems, 21, 273–284, https://doi.org/10.1109/TITS.2019.2891167, 2020.

Frangakis, A. S. and Hegerl, R.: Noise reduction in electron tomographic reconstructions using nonlinear anisotropic diffusion, Journal of Structural Biology, 135, 239–250, https://doi.org/10.1006/jsbi.2001.4406, 2001.

Frangi, A. F., Niessen, W. J., Vincken, K. L., and Viergever, M. A.: Multiscale vessel enhancement filtering, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 1496, 130–137, https://doi.org/10.1007/bfb0056195, 1998.

Fredrich, J. T. and Wong, T.-f.: Micromechanics of thermally induced cracking in three crustal rocks, Journal of Geophysical Research: Solid Earth, 91, 12 743–12 764, https://doi.org/https://doi.org/10.1029/JB091iB12p12743, 1986.

Furat, O., Wang, M., Neumann, M., Petrich, L., Weber, M., Krill, C. E., and Schmidt, V.: Machine learning techniques for the segmentation of tomographic image data of functional materials, Frontiers in Materials, 6, https://doi.org/10.3389/fmats.2019.00145, 2019.

Gastal, E. S. and Oliveira, M. M.: Domain transform for edge-aware image and video processing, ACM Transactions on Graphics, 30, https://doi.org/10.1145/1964921.1964964, 2011.

Gastal, E. S. and Oliveiray, M. M.: Adaptive manifolds for real-time high-dimensional filtering, ACM Transactions on Graphics, 31, https://doi.org/10.1145/2185520.2185529, 2012.

Golub, G. H. and Van Loan, C. F.: Matrix Computations, The Johns Hopkins University Press, third edn., iSBN: 0801830109, 9780801830105, 1996.

Griffiths, L., Heap, M., Baud, P., and Schmittbuhl, J.: Quantification of microcrack characteristics and implications for stiffness and strength of granite, International Journal of Rock Mechanics and Mining Sciences, 100, 138–150, https://doi.org/https://doi.org/10.1016/j.ijrmms.2017.10.013, 2017.

Halisch, M., Steeb, H., Henkel, S., and Krawczyk, C. M.: Pore-scale tomography and imaging: applications, techniques and recommended practice, Solid Earth, 7, 1141–1143, https://doi.org/10.5194/se-7-1141-2016, 2016.

Healy, D., Rizzo, R. E., Cornwell, D. G., Farrell, N. J., Watkins, H., Timms, N. E., Gomez-Rivas, E., and Smith, M.: FracPaQ: A MATLAB™ toolbox for the quantification of fracture patterns, Journal of Structural Geology, 95, 1–16, https://doi.org/https://doi.org/10.1016/j.jsg.2016.12.003, 2017.

Huang, N., Liu, R., Jiang, Y., Cheng, Y., and Li, B.: Shear-flow coupling characteristics of a three-dimensional discrete fracture network-fault model considering stress-induced aperture variations, Journal of Hydrology, 571, 416–424, https://doi.org/10.1016/j.jhydrol.2019.01.068, 2019.

Jiang, T., Zhang, J., and Wu, H.: Experimental and numerical study on hydraulic fracture propagation in coalbed methane reservoir, Journal of Natural Gas Science and Engineering, 35, 455–467, https://doi.org/10.1016/j.jngse.2016.08.077, 2016.

Jing, Y., Armstrong, R. T., and Mostaghimi, P.: Rough-walled discrete fracture network modelling for coal characterisation, Fuel, 191, 442–453, https://doi.org/10.1016/j.fuel.2016.11.094, 2017.

Jollife, I. T. and Cadima, J.: Principal component analysis: A review and recent developments, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 374, https://doi.org/10.1098/rsta.2015.0202, 2016.

Karimpouli, S., Tahmasebi, P., Ramandi, H. L., Mostaghimi, P., and Saadatfar, M.: Stochastic modeling of coal fracture network by direct use of micro-computed tomography images, International Journal of Coal Geology, 179, 153–163, https://doi.org/10.1016/j.coal.2017.06.002, 2017.

Karimpouli, S., Tahmasebi, P., and Saenger, E. H.: Coal Cleat/Fracture Segmentation Using Convolutional Neural Networks, Natural Resources Research, 29, 1675–1685, https://doi.org/10.1007/s11053-019-09536-y, 2019.

Ketcham, R. A. and Hanna, R. D.: Beam hardening correction for X-ray computed tomography of heterogeneous natural materials, Computers & Geosciences, 67, 49–61, https://doi.org/10.1016/j.cageo.2014.03.003, 2014.

Khryashchev, V., Ivanovsky, L., Pavlov, V., Ostrovskaya, A., and Rubtsov, A.: Comparison of Different Convolutional Neural Network Architectures for Satellite Image Segmentation, in: Conference of Open Innovation Association, FRUCT, vol. 2018-Novem, pp. 172–179, https://doi.org/10.23919/FRUCT.2018.8588071, 2018.

Kingma, D. P. and Ba, J.: Adam: A Method for Stochastic Optimization, 2017.

Kodym, O. and Španěl, M.: Semi-automatic ct image segmentation using random forests learned from partial annotations, in: BIOIMAGING 2018 - 5th International Conference on Bioimaging, Proceedings; Part of 11th International Joint Conference on Biomedical Engineering Systems and Technologies, BIOSTEC 2018, vol. 2, pp. 124–131, https://doi.org/10.5220/0006588801240131, 2018.

Kumari, W. G., Ranjith, P. G., Perera, M. S., and Chen, B. K.: Experimental investigation of quenching effect on mechanical, microstructural and flow characteristics of reservoir rocks: Thermal stimulation method for geothermal energy extraction, Journal of Petroleum Science and Engineering, 162, 419–433, https://doi.org/10.1016/j.petrol.2017.12.033, 2018.

Lai, J., Wang, G., Fan, Z., Chen, J., Qin, Z., Xiao, C., Wang, S., and Fan, X.: Three-dimensional quantitative fracture analysis of tight gas sandstones using industrial computed tomography, Scientific Reports, 7, 1–12, https://doi.org/10.1038/s41598-017-01996-7, 2017.

Lei, Q. and Gao, K.: Correlation Between Fracture Network Properties and Stress Variability in Geological Media, Geophysical Research Letters, 45, 3994–4006, https://doi.org/10.1002/2018GL077548, 2018.

Li, P., Xia, H., Zhou, B., Yan, F., and Guo, R.: A Method to Improve the Accuracy of Pavement Crack Identification by Combining a Semantic Segmentation and Edge Detection Model, Applied Sciences, 12, https://doi.org/10.3390/app12094714, 2022.

Li, S., Song, W., Fang, L., Chen, Y., Ghamisi, P., and Benediktsson, J. A.: Deep learning for hyperspectral image classification: An overview, IEEE Transactions on Geoscience and Remote Sensing, 57, 6690–6709, https://doi.org/10.1109/TGRS.2019.2907932, 2019.

Li, S. Z. and Jain, A., eds.: Local Adaptive Thresholding, pp. 939–939, Springer US, Boston, MA, https://doi.org/10.1007/978-0-387-73003-5_506, 2009.

Lissa, S., Ruf, M., Steeb, H., and Quintal, B.: Effects of crack roughness on attenuation caused by squirt flow in Carrara marble, in: SEG Technical Program Expanded Abstracts 2020, Society of Exploration Geophysicists, https://doi.org/10.1190/segam2020-3427789.1, 2020.

Lissa, S., Ruf, M., Steeb, H., and Quintal, B.: Digital rock physics applied to squirt flow, Geophysics, (in press), 2021.

Long, J., Shelhamer, E., and Darrell, T.: Fully Convolutional Networks for Semantic Segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence, 39, 640–651, https://doi.org/10.1109/TPAMI.2016.2572683, 2017.

MacQueen, J.: Some methods for classification and analysis of multivariate observations, in: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics, pp. 281–297, University of California Press, Berkeley, Calif., https://projecteuclid.org/euclid.bsmsp/1200512992, 1967.

Marr, D. and Hildreth, E.: Theory of edge detection, Proceedings of the Royal Society of London - Biological Sciences, 207, 187–217, https://doi.org/10.1098/rspb.1980.0020, 1980.

Maurer, C. R., Qi, R., and Raghavan, V.: A linear time algorithm for computing exact Euclidean distance transforms of binary images in arbitrary dimensions, IEEE Transactions on Pattern Analysis and Machine Intelligence, 25, 265–270, https://doi.org/10.1109/TPAMI.2003.1177156, 2003.

Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N., and Terzopoulos, D.: Image Segmentation Using Deep Learning: A Survey, pp. 1–23, http://arxiv.org/abs/2001.05566, 2020.

Mumford, D. and Shah, J.: Optimal approximations by piecewise smooth functions and associated variational problems, Communications on Pure and Applied Mathematics, 42, 577–685, https://doi.org/10.1002/cpa.3160420503, 1989.

Nguyen, T. S., Avila, M., and Begot, S.: Automatic detection and classification of defect on road pavement using anisotropy measure, in: 2009 17th European Signal Processing Conference, pp. 617–621, 2009.

Nobuyuki Otsu: A Threshold Selection Method from Gray-Level Histograms, IEEE Trans. Syst. Man Cybern, 9, 62–66, 1979.

Osher, S. and Tsai, R.: Level Set Methods and Their Applications in Image Science, Communications in Mathematical Sciences, 1, 1–20, https://doi.org/10.4310/cms.2003.v1.n4.a1, 2003.

Palafox, L. F., Hamilton, C. W., Scheidt, S. P., and Alvarez, A. M.: Automated detection of geological landforms on Mars using Convolutional Neural Networks, Computers & Geosciences, 101, 48–56, https://doi.org/https://doi.org/10.1016/j.cageo.2016.12.015, 2017.

Peacock, S., McCann, C., Sothcott, J., and Astin, T.: Seismic velocities in fractured rocks: an experimental verification of Hudson's theory, Geophysical Prospecting, 42, 27–80, https://doi.org/10.1111/j.1365-2478.1994.tb00193.x, 1994.

Pearson, K.: LIII. On lines and planes of closest fit to systems of points in space, The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 2, 559–572, 1901.

Perona, P. and Malik, J.: Scale-Space and Edge Detection Using Anisotropic Diffusion, IEEE Transactions on Pattern Analysis and Machine Intelligence, 12, 629–639, https://doi.org/10.1109/34.56205, 1990.

Pieri, M., Burlini, L., Kunze, K., Stretton, I., and Olgaard, D. L.: Rheological and microstructural evolution of Carrara marble with high shear strain: results from high temperature torsion experiments, Journal of Structural Geology, 23, 1393 – 1413, https://doi.org/10.1016/S0191-8141(01)00006-2, 2001.

Pimienta, L., Orellana, L. F., and Violay, M.: Variations in Elastic and Electrical Properties of Crustal Rocks With Varying Degree of Microfracturation, Journal of Geophysical Research: Solid Earth, 124, 6376–6396, https://doi.org/10.1029/2019jb017339, 2019.

Poulose, M.: Literature Survey on Image Deblurring Techniques, International Journal of Computer Applications Technology and Research, 2, 286–288, https://doi.org/10.7753/ijcatr0203.1014, 2013.

Ramandi, H. L., Mostaghimi, P., and Armstrong, R. T.: Digital rock analysis for accurate prediction of fractured media permeability, Journal of Hydrology, 554, 817–826, https://doi.org/10.1016/j.jhydrol.2016.08.029, 2017.

Rezaie, A., Achanta, R., Godio, M., and Beyer, K.: Comparison of crack segmentation using digital image correlation measurements and deep learning, Construction and Building Materials, 261, 120 474, https://doi.org/https://doi.org/10.1016/j.conbuildmat.2020.120474, 2020.

Roberts, G., Haile, S. Y., Sainju, R., Edwards, D. J., Hutchinson, B., and Zhu, Y.: Deep Learning for Semantic Segmentation of Defects in Advanced STEM Images of Steels, Scientific Reports, 9, 1–12, https://doi.org/10.1038/s41598-019-49105-0, 2019.

Ronneberger, O., Fischer, P., and Brox, T.: U-net: Convolutional networks for biomedical image segmentation, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9351, 234–241, https://doi.org/10.1007/978-3-319-24574-4_28, 2015.

Ruf, M. and Steeb, H.: An open, modular, and flexible micro X-ray computed tomography system for research, Review of Scientific Instruments, 91, 113 102, https://doi.org/10.1063/5.0019541, 2020a.

Ruf, M. and Steeb, H.: micro-XRCT data set of Carrara marble with artificially created crack network: fast cooling down from 600°C, https://doi.org/10.18419/DARUS-682, 2020b.

Saenger, E. H., Vialle, S., Lebedev, M., Uribe, D., Osorno, M., Duda, M., and Steeb, H.: Digital carbonate rock physics, Solid Earth, 7, 1185–1197, https://doi.org/10.5194/se-7-1185-2016, 2016.

Salman, N.: Image Segmentation Based on Watershed and Edge Detection Techniques, The International Arab Journal of Information Technology, 3, 104–110, 2006.

Sarout, J., Cazes, E., Delle Piane, C., Arena, A., and Esteban, L.: Stress-dependent permeability and wave dispersion in tight cracked rocks: Experimental validation of simple effective medium models, Journal of Geophysical Research: Solid Earth, 122, 6180–6201, https://doi.org/10.1002/2017jb014147, 2017.

Sato, Y., Nakajima, S., Atsumi, H., Roller, T., Gerig, G., Yoshida, S., and Kikinis, R.: 3D Multi-Scale Line Filter for Segmentation and Visualization of Curvilinear Structures in Medical Images, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 1205, 213–222, https://doi.org/10.1007/bfb0029240, 1997.

Sheppard, A. P., Sok, R. M., and Averdunk, H.: Techniques for image enhancement and segmentation of tomographic images of porous materials, Physica A: Statistical Mechanics and its Applications, 339, 145–151, https://doi.org/10.1016/j.physa.2004.03.057, 2004.

Shorten, C. and Khoshgoftaar, T. M.: A survey on Image Data Augmentation for Deep Learning, Journal of Big Data, 6, https://doi.org/10.1186/s40537-019-0197-0, 2019.

Simonyan, K. and Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition, https://doi.org/10.48550/ARXIV.1409.1556, 2014.

Singh, S. P., Wang, L., Gupta, S., Goli, H., Padmanabhan, P., and Gulyás, B.: 3D Deep Learning on Medical Images: A Review, http://arxiv.org/abs/2004.00218, 2020.

Su, T. C., Yang, M. D., Wu, T. C., and Lin, J. Y.: Morphological segmentation based on edge detection for sewer pipe defects on CCTV images, Expert Systems with Applications, 38, 13 094–13 114, https://doi.org/10.1016/j.eswa.2011.04.116, 2011.

Suzuki, A., Miyazawa, M., Okamoto, A., Shimizu, H., Obayashi, I., Hiraoka, Y., Tsuji, T., Kang, P., and Ito, T.: Inferring fracture forming processes by characterizing fracture network patterns with persistent homology, Computers & Geosciences, 143, 104 550, https://doi.org/https://doi.org/10.1016/j.cageo.2020.104550, 2020.

Taylor, H. F., O'Sullivan, C., and Sim, W. W.: A new method to identify void constrictions in micro-CT images of sand, Computers & Geotechnics, 69, 279–290, https://doi.org/10.1016/j.compgeo.2015.05.012, 2015.

van Santvoort, J. and Golombok, M.: Improved recovery from fractured oil reservoirs, Journal of Petroleum Science and Engineering, 167, 28–36, https://doi.org/https://doi.org/10.1016/j.petrol.2018.04.002, 2018.

Vincent, L. and Dougherty, E. R.: Morphological Segmentation for Textures and Particles, Digital Image Processing Methods, pp. 43–102, https://doi.org/10.1201/9781003067054-2, 1994.

Voorn, M., Exner, U., and Rath, A.: Multiscale Hessian fracture filtering for the enhancement and segmentation of narrow fractures in 3D image data, Computers & Geosciences, 57, 44–53, https://doi.org/10.1016/j.cageo.2013.03.006, 2013.

Weerakone, W. M. and Wong, R. C.: Characterization of Variable Aperture Rock Fractures Using X-ray Computer Tomography, in: Advances in X-ray Tomography for Geomaterials, pp. 229–235, https://doi.org/10.1002/9780470612187.ch21, 2010.

Wetzstein, G., Lanman, D., Hirsch, M., and Raskar, R.: Supplementary Material: Tensor Displays: Compressive Light Field Synthesis using Multilayer Displays with Directional Backlighting A Additional Details on Nonnegative Matrix and Tensor Factorization, pp. 1–20, https://doi.org/10.1145/2185520.2185576, 2012.

Xing, C., Huang, J., Xu, Y., Shu, J., and Zhao, C.: Research on crack extraction based on the improved tensor voting algorithm, Arabian Journal of Geosciences, 11, 342, https://doi.org/10.1007/s12517-018-3676-2, 2018.

Yamaguchi, T. and Hashimoto, S.: Fast crack detection method for large-size concrete surface images using percolation-based image processing, Machine Vision and Applications, 21, 797–809, https://doi.org/10.1007/s00138-009-0189-8, 2010.

Zhang, G., Ranjith, P. G., Perera, M. S., Haque, A., Choi, X., and Sampath, K. S.: Characterization of coal porosity and permeability evolution by demineralisation using image processing techniques: A micro-computed tomography study, Journal of Natural Gas Science and Engineering, 56, 384–396, https://doi.org/10.1016/j.jngse.2018.06.020, 2018.

Zhou, Z., Siddiquee, M. M. R., Tajbakhsh, N., and Liang, J.: UNet++: Redesigning Skip Connections to Exploit Multiscale Features in Image Segmentation, IEEE Transactions on Medical Imaging, pp. 1–1, https://doi.org/10.1109/tmi.2019.2959609, 2019.