

U-Net Crack Detection in Photos of Sandstones and Volcanic Rocks

Shi Yongxiang, Marco Ballesio, Dan Trentman, Ron Bruhn,
Yunsong Huang, Kasper Johansen,
Mathew McCabe and Gerard Schuster

October 20, 2020

Abstract

Cracks are a key feature used for assessing the integrity of rocks. To accurately detect cracks in drone images of large rock walls in the Middle East, we show that the U-Net convolutional neural network can be used to accurately detect cracks in thousands of drone photos by semantic segmentation. In our examples each photo was sub-divided into many sub-photos for labeling and training. Compared with ridge detection by the shearlet transform, the proposed approach yields fewer false positives and higher accuracy. The accuracy in identifying cracks in images is 97.5% in comparison to human identification, which is sufficient for assessing the general crack properties of the rock faces for some engineering or geology projects. After training on sub-photos from 127 photos, our trained network was able to successfully detect cracks in 23,845 photographs in less than 22 hours using two NVidia V100 GPUs. Training the network required 20 hours on the same computer system but the bottleneck was the more than 100 hours for the manual labeling of photos.

Then we applied the trained network to photos of a volcanic outcrop more than 8000 miles away in Wyoming. The network was able to detect more than 80% of the observable cracks without additional training. With a modest amount of extra labeling on photos of the volcanic outcrop and transfer training, we found that the accuracy of the labeled data significantly improved. We also demonstrated that the sandstone U-Net was able to detect many linear features in photos of the Mars surface taken by a Mars orbiting satellite.

The surprising outcome of this research is that the U-Net crack detector trained on photos of sedimentary rocks can also be effective on photos of volcanic rock faces. Better accuracy could be achieved with transfer training and less than an hour of effort in labeling the cracks in the volcanic rock images. This can be important for real-time assessment of geological hazards and lithology information for dam inspection, active volcanoes and planetary exploration by autonomous vehicles.

1 Introduction

All solids weaken over time and develop a reduction in their mechanical strength. A sign of this weakness is the development of cracks, a quasi-linear physical separation of material on the surface and in the interior of a solid. For siting buildings on a rock foundation, assessing dam hazards, avoiding drilling hazards or for mining excavation, it is critical to assess the density and distribution of fractures in the rock mass. To aid in this task, thousands of aerial photos of a rock area can easily be obtained by cameras mounted on Unmanned Aerial Vehicles (UAVs). As introduced in [?], UAVs can be programmed to photograph an area of interest to a centimeter accuracy, no matter how large the rock mass.

In the past, surface cracks were labeled by experts and their distribution was cataloged ([?]). This gave engineers an estimate of the integrity and stability of the rock mass ([?]). However, if there are thousands of images then the manual interpretation of cracks is both time consuming and error prone because of variable lighting, shadows, non-crack erosional features, rock spall, and complex rock surfaces. In addition, the accuracy of manual crack interpretation depends on the expertise of the interpreter, and the interpretation criteria ([?],[?]). Therefore, there is a growing demand to develop tools that can automatically detect and catalogue cracks and rock shapes in a more efficient and accurate way. As an example, Figure 1b shows a picture in which the red curves delineate the cracks. Once the cracks are labeled, their orientations and density distributions can easily be calculated to give the Rose diagram in Figure 1b, where most of the cracks are nearly horizontal.

Because of the needle-like shape of cracks, many crack detection methods are based on edge-detection algorithms. For example, [?] successfully applied edge-detection algorithms to UAV photographs to detect cracks in building structures. Their algorithm applied Sobel ([?]) and particle filters ([?]) for detecting cracks in building facades. For fractured outcrop

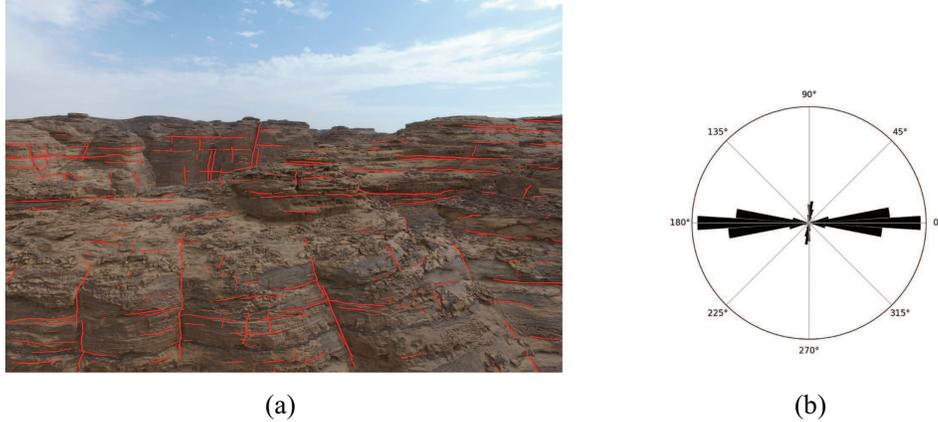


Figure 1: (a) UAV photo of a rock face in the survey area, where red curves delineate cracks. (b) Distribution of crack orientations in the photo.

images, [?] used the complex shearlet transform to automatically extract fracture ridge realizations from images. After getting the features of ridges, post-processing image analysis algorithms were then used to vectorize the fracture traces in an automated manner. The resulting fracture detection maps are generally consistent with the human-labeled fractures in photos taken over a geological outcrop in Parmelan, France.

One problem with an image processing approach is that it cannot easily distinguish abrupt changes in a photo’s intensity caused by, for example, a defects in the photo or some other physical lineament ([?]). This type of noise is easily distinguished by an interpreter, but manual detection is labor intensive when thousands of photos must be analyzed. To incorporate human intelligence into the automated decision process, convolutional neural network (CNN) architectures were introduced to detect cracks. In the earliest applications, a variety of CNN networks including AlexNet ([?]) and VGG16 ([?]) were used to detect cracks in concrete structures. Their limitation is that they are inefficient for high-resolution semantic segmentation, which is the task of classifying all pixels in the input image. Such CNN methods cannot localize cracks at the pixel level. In the work of [?], a method based on CNN can only locate cracks in a box which has the same size as the CNN’s input samples. Sufficiently large samples are required by this CNN to ensure its accuracy, which limits its resolution.

The problem with the previous CNN approaches is that they are not directly applicable to photos of large rock faces, such as the facade in Figure

2. Here, the photos are complicated by uneven lighting conditions, different view angles, erosional lineaments, long linear shadows, and different distances between the aerial camera and rock face. All of these imperfections can degrade the accuracy of crack detection by a standard CNN method. To mitigate these problems, we develop a novel architecture of the U-Net CNN designed for semantic segmentation.

The standard U-Net CNN ([?]) mitigates the main flaw of previous CNN architectures by classifying each pixel in the input image with a very high precision. It has been widely used by medical personnel for analyzing MRI, CT, and ultrasound scans ([?], [?], [?], [?], [?]). The U-Net architecture has been successfully applied for crack detection in concrete structures in [?], [?], [?]. In our work, we innovate the U-Net architecture by adding special convolutional modules that are tuned to efficiently detect the crack orientations that are primarily along the horizontal or vertical axes. These orientations dominate the sandstone crack distributions in more than 23,000 drone photos in our study area (see Figure 4).

Most engineering geologists will often use the terms *joints* or *fractures* instead of cracks in rock faces ([?]). We have not checked for the mode of origin so we will use the term *crack* to indicate any discontinuity on the surface of the rock face, whether it is a bedding surface, a shearing surface, a fracture or a joint where the rock face separates due to cooling.

1.1 Crack Detection in the Sandstone Massif



Figure 2: Some photos taken in the survey area.

Sandstones in the study area are cut by a NW-trending, roughly vertical set of long cracks that dissect the rocks, forming blocky massifs and elongated rock fins that are several tens to hundreds of meters wide, up to a

kilometer or more in length and several tens of meters in height. NE-trending vertical cracks form a second but less prominent set of vertical cracks in the study area. Cliff faces or facades are marked by traces of both vertical and sub-horizontal cracks, with the latter aligned parallel to bedding. Discontinuous vertical cracks located within the rocks terminate at or near bedding surfaces suggesting that mechanical changes across bedding locally inhibited crack growth [?]. Bedding-parallel cracks are relatively short, discontinuous features caused mostly by failure along bedding surfaces exposed in vertical facades. Consequently, failure along bedding surfaces should be anticipated in the design of underground excavations. This type of failure is common in crack (joint) bounded rock fins, where failure and collapse along bedding surfaces leads to the development of natural arches.

Large cracks, especially those longer than 1 meter and visible at the scale of the photos, pose a problem for engineers who must drill into portions of the sandstones, as shown in Figure 2. Prior to drilling, a strict safety assessment must be carried out to access the rock integrity. Mechanical integrity is related to the density and distribution of large cracks, where drilling into the massif with a high density of cracks must be avoided or extra precautions should be taken.

To quantitatively estimate the crack density, more than 23,000 drone photos were taken of the study area. The cracks in the raw photos were labeled by human interpreters that required more than 100 hours of laborious labeling. A semantic segmentation algorithm is then used to detect and label large cracks resulting in a 97.5% accuracy in terms of matching the cracks identified by a human interpreter. This type of accuracy is sufficient to assess rock integrity prior to drilling into the massif.

This paper describes the steps for detecting the cracks in the drone photos. It is organized into four sections. Following the introduction, we present the methodology of our CNN procedure for crack detection. It is an optimized U-Net that includes special convolutional modules that are tuned to the mostly near-vertical and near-horizontal orientations of the cracks, as well as their large length/width ratios. The workflow for training and testing of the CNN is shown in Figure 3. After this, we describe the training of the network and present results from the validation set. The next section presents the numerical results of applying the trained U-Net to more than 23,000 unlabeled drone images of our survey area. To test the generalizability of the trained U-Net, we use it to detect cracks in photos of volcanic rocks near the Teton dam site in Wyoming. We draw conclusions in the last section.

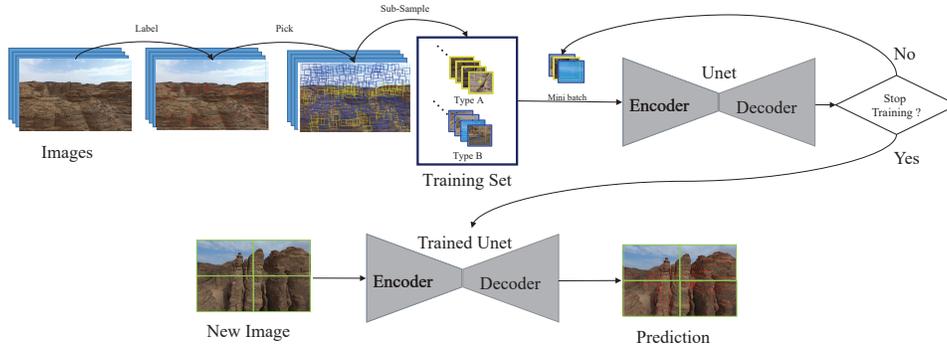


Figure 3: Workflow for training and labeling of cracks in drone photos.

2 Methodology

We first describe the U-Net architecture for detecting cracks in drone photos. Some of the convolution modules are tuned to the dominant dip orientations of the cracks and the large length/thickness ratios of cracks, and so this makes our U-Net more efficient. We then describe the procedures for labeling, training and validation of sub-photos from 127 photos taken by a high-resolution camera mounted on a drone. This is only a fraction of the more than 23,000 drone photos taken over a sandstone massif in the Middle East.

2.1 U-Net Architecture

The U-Net architecture is diagrammed in Figure 4. It consists of a series of contracting encoders followed by expanding decoders weighted with features from the contraction path. In addition to the cross-connected paths, each convolution block implements an internal encoder/decoder with three convolution branches to enhance the detection of specific orientations of cracks. The convolution block forms the residual function to which an identity operation is added ([?]). The dimensions of the final U-Net output are the same as the input images but there is only one output channel of binary values to represent whether a pixel is labeled as background or a crack.

The U-Net design was chosen for the following reasons:

- **Additional convolution branches:** This idea comes from the Inception Network ([?]) because most of the cracks in our drone images are approximately horizontal or vertical. The two additional convolution branches with filter sizes 1×9 and 9×1 enable the efficient extraction

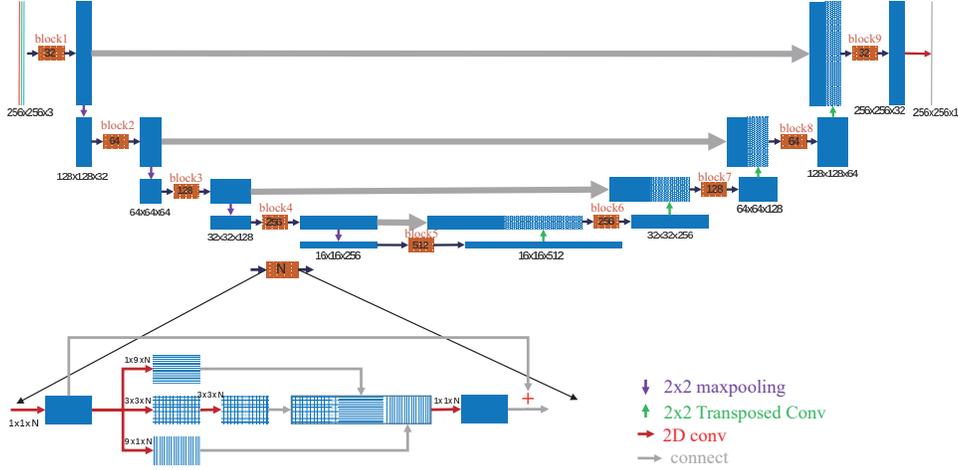


Figure 4: U-Net architecture used for detecting cracks in drone images. In a) and b) each arrowed tier represents a layer; the blue boxes represent features after the calculations from that layer. The input is a 256×256 RGB image. After five convolutional blocks (brown block) and down-sampling (violet arrow), the feature map has the size 16×16 . This is followed by up-sampling of features with transposed layers (green arrow), concatenation (gray arrow) with the former output of the same size, and a final convolution. The final output size matches the 256×256 input to allow individual pixel classification of the input. Each convolutional block contains 6 separate convolutional layers (red arrow) with a final input identity operation (+) summed with the residual function block to form the desired output.

of crack features by focusing on a specified dimension without the need for a square filter. A square filter with many parameters can detect a wide variation of crack orientations, but this is computationally inefficient if the cracks are confined to just a few orientations. This is a data specific enhancement.

- **Residual function connection:** In deep learning models, the convergence rate and accuracy can become degraded with an increase in the number of layers ([?]). Including a residual operation to each convolution block helps to improve accuracy in deep models [?].

Our U-Net has 9 convolutional blocks, including 36 convolution layers, 4 maxpooling layers, and 4 transposed layers. For each convolution layer, we include batch normalization ([?]) and ReLU activation ([?]). Each con-

volution layer is followed by a dropout layer with a rate of 0.5 (a random choice, half are dropped) for additional regularization. The final layer uses a sigmoid function to constrain the output label to binary values of either 1 or 0.

The labeling targets are cracks that tend to form less than 1% of the entire set of images. Therefore, there is an imbalanced set of equations due to most labels having a label value of 0 (background). This typically leads to poor convergence and large errors in inference labeling. To overcome the imbalance problem, we test the performance of two weighted loss functions:

1. Weighted cross entropy (WCE):

$$WCE(y, p) = - \sum_i [\lambda y^{(i)} \log p^{(i)} + (1 - \lambda)(1 - y^{(i)}) \log(1 - p^{(i)})],$$

where $y^{(i)}$ is our assigned label for the i^{th} example, $p^{(i)}$ is the U-Net prediction, and λ is a scalar chosen during training to improve convergence.

2. Focal loss (FL) [?]:

$$FL(y, p) = - \sum_i [\lambda y^{(i)} \times (1 - p^{(i)})^\gamma \log p^{(i)} + (1 - \lambda)(1 - y^{(i)}) \times (p^{(i)})^\gamma \log(1 - p^{(i)})],$$

where we use the constant $\gamma = 2$.

WCE is a typical loss function used in classification problems where λ provides for a rebalancing of the possibly underrepresented positive pixels in the loss calculation. FL is used to address class imbalance by down-weighting the contribution of correctly identified pixels to focus on the loss due to misidentified pixels. We used a fixed value of $\gamma = 2$, varying only λ for all comparisons of each loss function. WCE reduces to be an FL where $\gamma = 0$. Therefore, we effectively compare FL with two values for γ . Our U-Net architecture is implemented using Keras 2.2.3 and Tensorflow-GPU 1.14.

2.2 Labeling, Training and Validation

Although the facade and top images have different crack features, we only describe U-Net training of the facade images because the workflow is the same for the top images.

Labeling and Subimages Picking

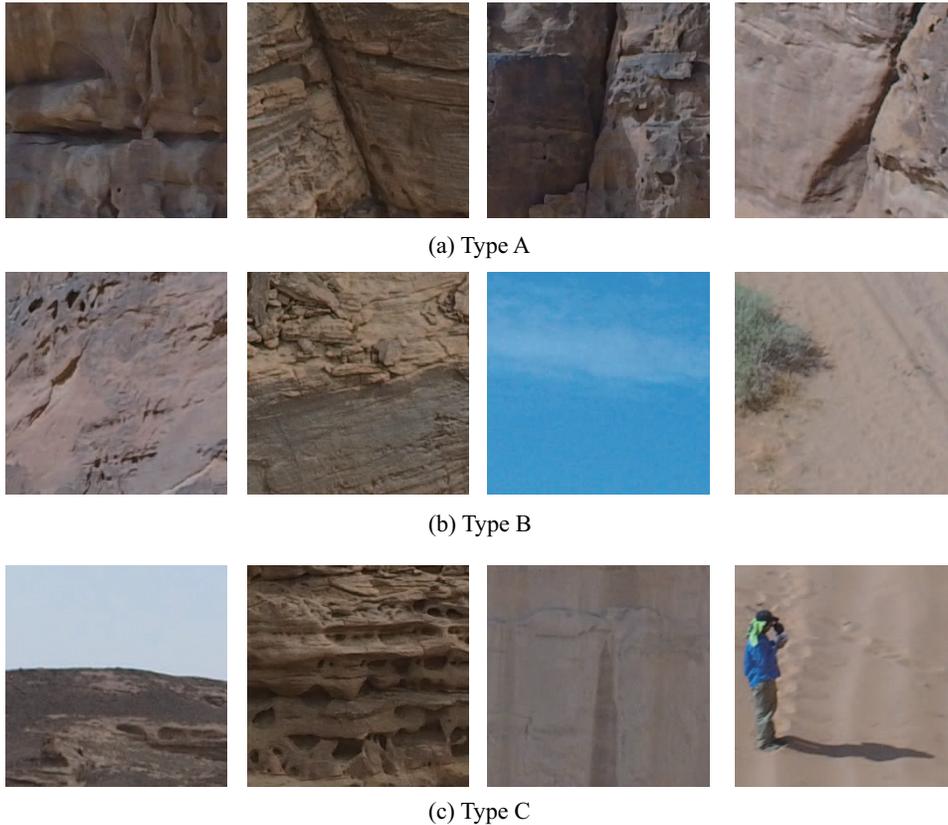


Figure 5: Different pictures from the drone dataset with more than 23,000 photos.

We select 57 photos from the facade views to be manually labeled for training and validation. Fifty-four photos are used for training and 3 were used for validation. The public-domain editing software GIMP is used to label the crack lines, each approximately 6-pixels wide to closely match the size of the cracks of interest. Image labeling is a time intensive task necessary for the training of the CNN. Labeling requires approximately 30-60 minutes per photo, each is comprised of 3000×4000 pixels so that more than 100 hours are needed to manually label the photos used for training and validation.

The biggest problem in labeling is defining what constitutes a crack of

interest. In the drone photos, there is a large variation in the size of cracks and not all cracks are of interest. For these data, geologists and building-site engineers are only interested in identifying cracks longer than 1 meter, which are long enough to indicate possible instability in the building project. There are two examples of crack labels in Figure 7a. We only label those that we are quite confident about. Soil and small-rock sections covering parts of cracks are not labeled, resulting in discontinuous crack labels. Another problem is the mixture of horizontal cracks and some special bedding contacts. Some bedding contacts are strongly eroded so that there are some troughs along bedding surfaces. It is challenging to discriminate them from cracks. So we could include some troughs into the "horizontal cracks" label.

There are three types of subimages selected from the labeled photos: subimages of type A contain labeled cracks (Figure 5a); type B subimages contain only background d rock, sky, sand, and/or bare rocks without any complex shapes, and are devoid of labeled cracks (Figure 5b); type C is also devoid of labeled cracks but includes items which complicate network training, including line-like shapes of rain traces, shadow edges, and trees (Figure 5c). Although the CNN output is a binary value classification, identifying three types for inclusion in training allows for selecting a good balance of cracks and non-cracks. Each type is not equally represented in each subimage. Varying the number of B and C type objects used for training shows no improvement in the accuracy of predicting the class type. This suggests that selective inclusion in the training is not necessary and it is only necessary to balance the combined types B and C, hereafter named B, with type A to improve the CNN output.

For samples of type A, labels were indexed with small random shifts in the choice of index positions along the cracks to include the ends of cracks and avoid choosing only the centers when selecting training samples. Type B subimages are randomly sampled. To reduce the bias of samples being too close to each other, a minimal spacing distance was set for center points of types A and B as 70 and 150 pixels, respectively. We select a count of 100 A types and 200 B types in each photo for balanced training.

2.2.1 Network Training

Subsampling the 57 photos resulted in a training dataset consisting of 16,200 subimages and a validation dataset consisting of 900 subimages. The CNN networks are trained using two Tesla v100 GPU cards with a batch size of 20 sub-photos and an initial learning rate of 0.001. The data are augmented in each batch of processed training samples by adding copies

with horizontal, vertical, or 0° to 45° rotation using a reflected sample to fill in the boundary space created by rotation. The data augmentation is performed within Keras.

The maximum number of epochs is set to 100 and training is stopped when either the maximum is reached or when loss in the validation set does not decrease for 30 epochs (Figure 6). Loss values in both the training and validation sets do not decrease very much after around epoch number 50, so training is terminated at epoch 80.

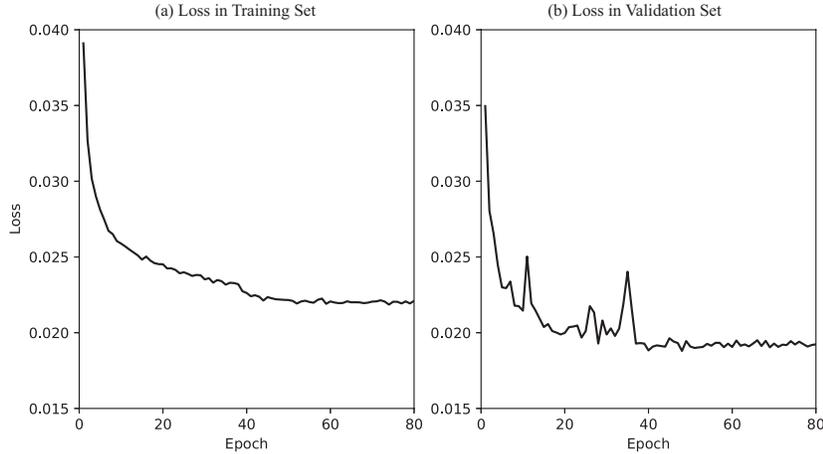


Figure 6: Learning curve in training. The loss value is defined using WCE and $\lambda = 0.9$.

2.2.2 Validation of Valid Images

		Actual class	
		Crack	Non-crack
Prediction	Positive	True positive (TP)	False positive (FP)
	Negative	False negative (FN)	True negative (TN)

Table 1: Definition of a confusion matrix.

We used two loss functions and λ weights to train the U-Net. The performance of the U-Net using different parameters is determined by comparison of the resultant confusion matrix (defined in Table 1). The prediction accuracy is calculated from the values used in the confusion matrices as the

	Crack	Non-crack		Crack	Non-crack
Positive	244	3,560	Positive	410	792
Negative	291	32,000	Negative	125	34,768
(a) AFD			(b) U-Net with FL and $\lambda = 0.9$		

	Crack	Non-crack		Crack	Non-crack
Positive	213	255	Positive	408	775
Negative	254	35,348	Negative	127	34,785
(c) U-Net with WCE and $\lambda = 0.7$			(d) U-Net with WCE and $\lambda = 0.9$		

Table 2: Confusion matrices with different U-Net parameters. Units are 1000s of pixels.

value of $(TP+TN)/(TP+FN+FP+TN)$. After trial-and-error tests, an output value threshold of 0.5 is used for all pixels in the U-Net for classification as either a crack or background. Cracks in the output are marked as red in the figures. For comparison with U-Net, the automatic fracture detection (AFD) code in [?] is used as a ridge-detection method for the validation images. We used 108 shearlet systems with a threshold of 0.52 for AFD.

Figure 7 displays results from the U-Net and AFD codes. The results from AFD show many false positives caused by sharp changes in intensity associated with non-crack features and it missed some labeled cracks. The statistics in Table 2a shows that AFD produces more false positives and false negatives than the U-Net method for the parameter values used. Compared to AFD, The U-Net results in Figure 7d correlate well with human labeling, showing a significant reduction of FP in the background compared to AFD.

In Table 2b and 2d, the accuracy of the U-Net is 97.5%. U-Net shows a similar performance when using the different loss functions WCE and FL with the same λ weighting, as shown in Figure 7d and 7e. In Table 2d and 2e, with the same λ , FL shows improvement in the number of FNs but degrades in the number of FPs relative to WCE which indicates a similar performance. However, when we show the raw output of FL and WCE for the same image in Figure 8, the output of WCE more clearly highlights large cracks of interest. Given the limited values used for comparison, WCE is a better choice for our work.

Two examples from using different values of λ are displayed in Figure 7c and 7d, with the corresponding confusion matrices in Table 2c and 2b. U-

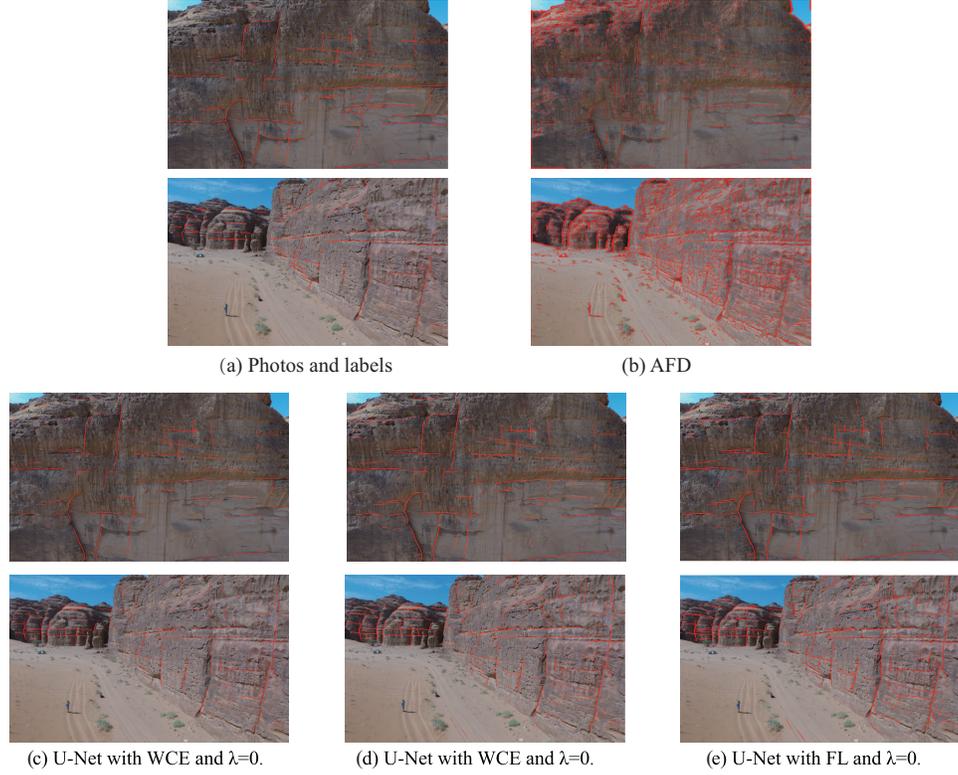


Figure 7: Results of different algorithms applied to two validation images. Photos are 4000×2250 pixels each. Cracks marked in red color overlain on raw photos. Human labeled images are shown in a) ; b) is calculated by automatic fracture detection (AFD)[?]; c) and d) are trained with weighted cross entropy (WCE) using different values of λ ; e) focal loss (FL) shows output from one value of λ .

Net shows more labeled cracks when using a greater value of λ , as expected due to preferential weighting of loss contributions from positive matches but it also leads to higher FP counts. Considering that underestimation of crack density could lead to safety problems, high FP counts are more acceptable than missing cracks of interest. To reduce the high number of FP counts associated with large values of λ , $\lambda = 0.9$ is used in the following work.

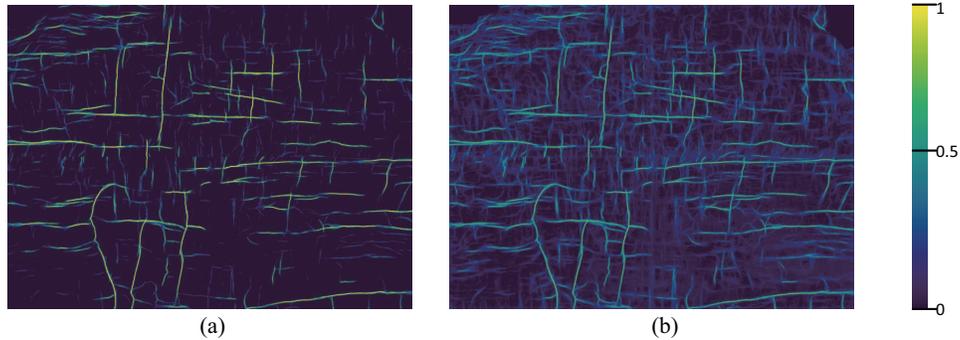


Figure 8: U-Net predictions of the first image in Figure 7a. Here, a) is the raw output from the U-Net with WCE as the objective function and b) is the raw output from the U-Net with FL as the objective function. The two kinds of objective functions share the same value of $\lambda = 0.9$. We use the sigmoid as the activation function in the last layer so when a value in the image is closer to 1, it is more likely to be a crack.

3 U-Net Crack Detection in Unlabeled Photos

The trained U-Net is now used to label cracks in the study area with the large sandstone massifs in Figure 2. Over 23,000 unlabeled photos of the sandstone massif are taken by a drone with a high-resolution camera. In addition, the trained U-Net is used to label cracks in photos of volcanic rocks in Idaho as well as those in Martian orbiter pictures. These last two examples were used to show that the U-Net trained on photos of sandstone cracks can also be used to label the cracks in photos of rocks with a different geological genesis. We also show how the accuracy of the U-Net crack detection can be improved by transfer learning.

3.1 Labeling of Cracks in Photos of Sandstone Massifs

A drone was flown over the sandstone massifs and recorded 23,845 photos of the tops and facades (sides) of the target rock masses. Variations in camera-to-target distances created pixels approximately 0.8 to 10 centimeter wide, with most having a spatial resolution of several centimeters. Photo sizes are 2250×4000 and 3000×4000 pixels with about 80% overlap of areas.

We were limited by the amount of GPU memory, so we partitioned each photo into 4 small sub-photos, each with half the width and height of the original. Partitioning is shown by the green lines in Figure 3. The U-Net

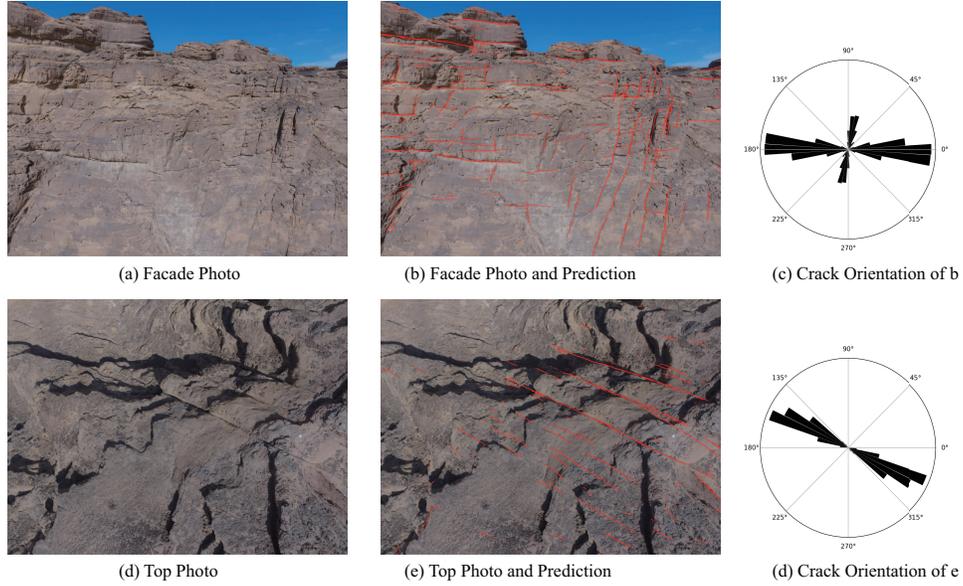


Figure 9: Photos labeled by U-Net from unlabeled photos. We picked two example images from the out-of-the training set. a) and b) depict a facade view; d) and e) are a top view. c) and f) are their distributions of crack orientations and densities.

labeling of each photo takes about 3 seconds per GPU card.

Then the U-Net model trained with WCE and $\lambda = 0.9$ was applied to the unlabeled facade images. As shown in Figure 9a and 9b, the U-Net results for unlabeled images are judged to be of acceptable accuracy. The orientation distribution in Figure 9c shows that most cracks in this photo are horizontal but there are some along 70° . We then applied the same method and trained a new U-Net for the top images, where we labeled 70 images which created a dataset with 16,800 subimages. Figure 9d and 9e shows a processed top image, showing results as accurate as those produced for the facade images. Figure 9f shows all cracks have the same orientation angle of about 150° , which is consistent with the cracks seen in Figure 9d. The U-Net labeling of the remaining 23,000 images (facade and top) required approximately 22 GPU hours to finish. The labeled images were then used to assess the crack densities and orientations in the areas of engineering interest.

3.2 Labeling of Cracks in Photos of Volcanic Rocks

After achieving accurate detection of cracks in the sandstone photos, we apply the sandstone U-Net¹ to photos of volcanic rock faces at the Teton Dam site in Idaho, USA (https://en.wikipedia.org/wiki/Teton_Dam). The Teton Dam in eastern Idaho failed catastrophically on the morning of June 5, 1976, causing a large flood. The dam is built on silica-rich volcanics (welded tuff) that were derived by the eruption of the Yellowstone super-volcano. Investigations suggest the failure of the dam was related to movement of water through large fractures (cracks) in the bedrock foundation.

Figure 10a depicts a 4000×2000 picture from a larger Gigapan image of cliffs at the Teton Dam site provided by R.L. Bruhn. Mapping the cracks in the Teton Dam photo tests the ability of the sandstone U-Net to accurately delineate cracks in photos of volcanic rocks. The nearly vertical cracks are cooling joints in welded volcanic tuff. The nearly vertical cracks mapped by our sandstone U-Net are labeled by the red lines in Figure 10b, where many cracks are correctly labeled. However, some cracks are not labeled even though they are visible to the eye and there are some false positives. This demonstrates that the sandstone U-Net algorithm is capable of mapping the trends in crack orientations of rocks that are different from those from which it was trained.

We can improve the accuracy of the sandstone U-Net model by using transfer learning. Transfer learning combines the CNN weights learned from the sandstone images as well as those learned from a small number of labeled cracks from the Teton dam photos. It only requires a small number of labeled images from the photos of volcanic rock faces because it reuses common crack patterns from the sandstone training. This can result in significantly less labeling and computation time compared to standard CNN training. For relabeling, we used another 4000×2000 photo of the Gigapan image, broke it up into 120 256×256 sub-photos, and manually labeled the cracks. This required less than 60 minutes of labor compared to an estimate of more than 20 hours of training required by a new U-Net model.

For the transfer training, we freeze the weights in blocks from 3 to 7 and allow four symmetric blocks (block 1,2 and 8,9 in Figure 4) to be trained on the newly labeled photos. This decreases the trainable parameters from 20,000,000 to 730,000. An Adam optimizer is used and we set the learning rate to be 10^{-4} in order to fine-tune the trainable layers. The batch size is 5 and the total number of 256×256 sub-photos is 120. The transfer training

¹The sandstone U-Net model is exclusively trained from the sandstone pictures.



Figure 10: Crack detection of volcanic rocks at the Teton Dam Site, Idaho. a): near-vertical cooling cracks in volcanic rocks just downstream from the dam site; b): cracks labeled in red by U-Net. The red markings correspond to a probability between 0.7 and 1.0 in the softmax output, while the white-bluish markings correspond to a probability between 0.3 and slightly less than 0.7.

is halted at epoch 30 when the accuracy flattens out at 97%, which takes no more than 20 minutes of computation time on a workstation. We denote this U-Net as the hybrid U-Net model because the original weights were trained on images of sandstones and adjusted to those for volcanic rocks.

After transfer learning, the sandstone U-Net and hybrid U-Net were applied to the top row of raw 4000×2000 images in Figure 11a to give the labeled cracks in Figure 11b and Figure 11c, respectively. Figure 11c. show a much denser and more accurate collection of labeled cracks than the ones in Figure 11b. The important accomplishment is that it required no more than an hour of manual labeling to achieve this goal.

3.3 Labeling of Photos of Martian Alluvial Fans

Photos from a satellite orbiting Mars are labeled by the sandstone U-Net model. Such labels do not necessarily delineate faults but detect sharp changes in intensity patterns in the photos. These sharp changes depict patterns in landscape that were likely generated by ancient water flows from the nearby mountains.

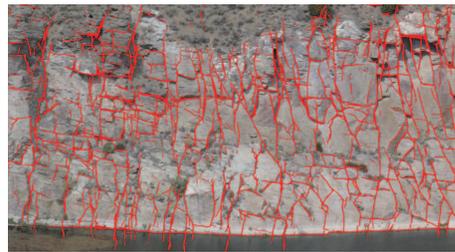
The raw and U-Net labeled orbiter photos are shown in Figure 12, where the red labels clearly delineate some of the alluvial fan patterns that likely formed from rivers flowing from the adjacent mountains. Regional rose diagrams of these features can be computed to show trends in the regional



a) Photos of the Dam Site



b) U-Net from the Survey Area



c) U-Net after Transfer Learning

Figure 11: a) Raw photos from the Teton dam site, and crack labels computed by the b) sandstone U-Net and the c) hybrid U-Net models. Red labels have a softmax probability between 0.7 and 1.0.

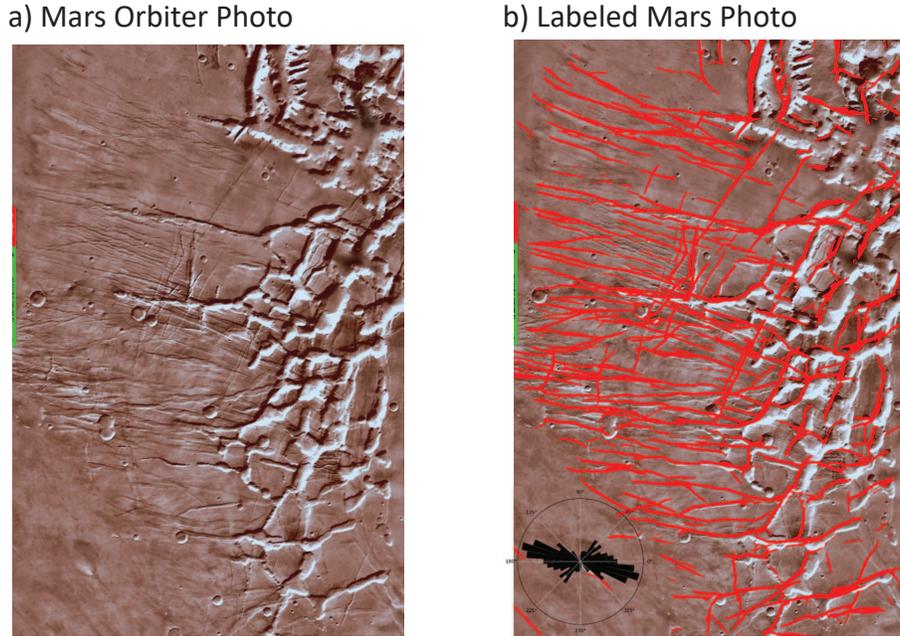


Figure 12: a) Photo taken by a satellite orbiting Mars and b) the crack-labeled photo. The *crack* distribution is depicted in the inset Rose diagram of b), and the labels are computed by the sandstone U-Net. In this example, a *crack* is defined as a sharp linear change in the photo’s intensity value.

changes in geology and morphology.

4 Conclusions and Discussion

This work presents a successful use of an improved U-Net CNN to label cracks in rock faces. We compare U-Net with a ridge-detection method and find that U-Net provides a more accurate detection of cracks, assessed by TP and TN counts, compared to AFD after being trained with just a few images. We believe our results suggest that the U-Net approach provides a viable alternative to the conventional AFD method for detecting cracks in rock massifs.

We tested several loss functions in our U-Net. Results show that using the same weight in the loss function and the threshold value of 0.5, WCE and FL show similar performance metrics. However, WCE produced a preferred

response highlighting large cracks. In comparison, FL included many small cracks and other edge-features considered to be noise for this application. Using different loss weights produces significantly different results and higher values, which causes the U-Net to label more cracks and generate higher TP counts but it also increases FP counts.

The disadvantage of U-Net labeling is that it requires a large amount of manual labeling to train the network. However, U-Net has transferability capabilities so that a well-trained U-Net with transfer training can be used to detect cracks on other kinds of rocks without an extensive effort in re-labeling. We proved this to be true by using transfer training to significantly improve the accuracy of labeling cracks in the Teton Dam photos. Less than 60 minutes were required for manual labeling of cracks in the Teton Dam photos.

There are some notable areas where the U-Net CNN can be improved. One complication is that close-up photos of the rock reduces U-Net’s ability to distinguish erosional features from small cracks. The consequence is that it produces false positives. We believe this problem can be mitigated by introducing distance information to the input of U-Net. Another problem is that manual labeling of cracks is tedious, error prone and time consuming, especially when rocks are broken or cracks are parallel to the bedding surfaces. Therefore, a more accurate and less labor intensive way of labeling cracks is desirable. In this case we suggest the use of transfer training as an effective means for labeling cracks in photos of any type of geology. This can be important for real-time assessment of geological hazards and lithology information for tasks such as dam inspection and planetary exploration by autonomous vehicles. For this reason we are making the weights and architecture of our sandstone and hybrid U-Nets to be available to the public.

Acknowledgments

The research reported in this publication was supported by the King Abdullah University of Science and Technology (KAUST) in Thuwal, Saudi Arabia. For computer time, this research used the resources of the Supercomputing Laboratory at KAUST and the IT Research Computing Group. We thank them for providing the computational resources required for carrying out this work.

References

- Aydan, O., Ulusay, R., and Tokashiki, N. (2014). A new rock mass quality rating system: rock mass quality rating (RMQR) and its application to the estimation of geomechanical characteristics of rock masses. *Rock mechanics and rock engineering*, 47(4):1255–1276
- Behboodi and Rivaz, 2019 Behboodi, B. and Rivaz, H. (2019). Ultrasound segmentation using U-Net: learning from simulated data and testing on real data. In 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pages 6628–6631. IEEE.
- Bemis et al., 2014 Bemis, S. P., Micklethwaite, S., Turner, D., James, M. R., Akciz, S., Thiele, S. T., and Bangash, H. A. (2014). Ground-based and UAV-based photogrammetry: A multi-scale, high-resolution mapping tool for structural geology and paleoseismology. *Journal of Structural Geology*, 69:163–178.
- Cha et al., 2017 Cha, Y.-J., Choi, W., and Bykztrk, O. (2017). Deep learning-based crack damage detection using convolutional neural networks. *Computer-Aided Civil and Infrastructure Engineering*, 32(5):361–378.
- Cheng et al., 2018 Cheng, J., Xiong, W., Chen, W., Gu, Y., and Li, Y. (2018). Pixel-level crack detection using U-Net. In TENCON 2018-2018 IEEE Region 10 Conference, pages 0462–0466. IEEE.
- Dorafshan et al., 2018 Dorafshan, S., Thomas, R. J., and Maguire, M. (2018). Comparison of deep convolutional neural networks and edge detectors for image-based crack detection in concrete. *Construction and Building Materials*, 186:1031–1045.
- Eckle and Schmidt-Hieber, 2019 Eckle, K. and Schmidt-Hieber, J. (2019). A comparison of deep networks with relu activation function and linear spline-type methods. *Neural Networks*, 110:232–242.
- Han and Ye, 2018 Han, Y. and Ye, J. C. (2018). Framing U-Net via deep convolutional framelets: Application to sparse-view ct. *IEEE Transactions on Medical Imaging*, 37(6):1418–1429.
- He and Sun, 2015 He, K. and Sun, J. (2015). Convolutional neural networks at constrained time cost. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 5353–5360.

- He et al., 2016 He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Hillier et al., 2015 Hillier, J., Smith, M., Armugam, R., Barr, I., Boston, C., Clark, C., Ely, J., Frankl, A., Greenwood, S., Gosselin, L., Httestrand, C., Hogan, K., Hughes, A., Livingstone, S., Lovell, H., McHenry, M., Munoz, Y., Pellicer, X., Pellitero, R., Robb, C., Roberson, S., Ruther, D., Spagnolo, M., Standell, M., Stokes, C., Storrar, R., Tate, N., and Wooldridge, K. (2015). Manual mapping of drumlins in synthetic landscapes to assess operator effectiveness. *Journal of Maps*, 11(5):719–729.
- Ioffe and Szegedy, 2015 Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502–03167*.
- Li et al., 2020 Li, G., Ma l network with alternately updated clique. *Sensors*, 20(3):717.
- Lin et al., 2017 Lin, T., Goyal, P., Girshick, R., He, K., and Dollr, P. (2017). Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2999–3007.
- Liu et al., 2019 Liu, Z., Cao, Y., Wang, Y., and Wang, W. (2019). Computer vision-based concrete crack detection using U-net fully convolutional networks. *Automation in Construction*, 104:129–139.
- Mohan and Poobal, 2018 Mohan, A. and Poobal, S. (2018). Crack detection using image processing: A critical review and analysis. *Alexandria Engineering Journal*, 57(2):787–798.
- Pereira and Pereira, 2015 Pereira, F. C. and Pereira, C. E. (2015). Embedded image processing systems for automatic recognition of cracks using uavs. *IFAC-PapersOnLine*, 48(10):16–21. 2nd IFAC Conference on Embedded Systems, Computer Intelligence and Telematics CESCIT 2015.
- Prabhakaran et al., 2019 Prabhakaran, R., Bruna, P.-O., Bertotti, G., and Smeulders, D. (2019). An automated fracture trace detection technique using the complex shearlet transform. *Solid Earth*, 10(6):2137–2166.
- Ronneberger et al., 2015 Ronneberger, O., Fischer, P., and Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer.

- Sander et al., 1997 Sander, P., Minor, T. B., and Chesley, M. M. (1997). Ground-water exploration based on lineament analysis and reproducibility tests. *Ground-water*, 35(5):888–894.
- Sanderson et al., 2019 Sanderson, D. J., Peacock, D. C., Nixon, C. W., and Rotevatn, A. (2019). Graph theory and the analysis of fracture networks. *Journal of Structural Geology*, 125:155–165. Back to the future.
- Silva and Lucena, 2018 Silva, W. R. L. d. and Lucena, D. S. d. (2018). Concrete cracks detection based on deep learning image classification. In *Multidisciplinary Digital Publishing Institute Proceedings*, volume 2, page 489.
- Sobel, 1990 Sobel, I. (1990). Machine vision for three-dimensional scenes. Chapter An Isotropic, 3(3):376–379.
- Szegedy et al., 2017 Szegedy, C., Ioffe, S., Vanhoucke, V., and Alemi, A. A. (2017). Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-First AAAI Conference on Artificial Intelligence*.
- Thrun et al., 2005 Thrun, S., Burgard, W., and Fox, D. (2005). *Probabilistic Robotics*. MIT Press.
- Yang et al., 2019 Yang, J., Faraji, M., and Basu, A. (2019). Robust segmentation of arterial walls in intravascular ultrasound images using dual path U-Net. *Ultrasonics*, 96:24–33.
- Yap et al., 2017 Yap, M. H., Pons, G., Marti, J., Ganau, S., Sentis, M., Zwiggelaar, R., Davison, A. K., and Marti, R. (2017). Automated breast ultrasound lesions detection using convolutional neural networks. *IEEE journal of biomedical and health informatics*, 22(4):1218–1226.
- Zhou et al., 2018 Zhou, Z., Rahman Siddiquee, M. M., Tajbakhsh, N., and Liang, J. (2018). UNet++: A nested U-Net architecture for medical image segmentation. In Stoyanov, D., Taylor, Z., Carneiro, G., Syeda-Mahmood, T., Martel, A., Maier-Hein, L., Tavares, J. M. R., Bradley, A., Papa, J. P., Belagiannis, V., Nascimento, J. C., Lu, Z., Conjeti, S., Moradi, M., Greenspan, H., and Madabhushi, A., editors, *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pages 3–11, Cham. Springer International Publishing.
- Zhuang et al., 2017 Zhuang, X., D. Tannant and G. Ma, 2017, Characterisation of fractures in rock: from theory to practice (ROCKFRAC): *Engineering Geology*: **225**, 1–114.